Smart Innovation, Systems and Technologies 246

Anupam Biswas
Ripon Patgiri
Bhaskar Biswas *Editors*

Principles of Social Networking

The New Horizon and Emerging Challenges

KES International

Springer

Smart Innovation, Systems and Technologies

Volume 246

The Smart Innovation, Systems and Technologies book series encompasses the topics of knowledge, intelligence, innovation and sustainability. The aim of the series is to make available a platform for the publication of books on all aspects of single and multi-disciplinary research on these themes in order to make the latest results available in a readily-accessible form. Volumes on interdisciplinary research combining two or more of these areas is particularly sought.

The series covers systems and paradigms that employ knowledge and intelligence in a broad sense. Its scope is systems having embedded knowledge and intelligence, which may be applied to the solution of world problems in industry, the environment and the community. It also focusses on the knowledge-transfer methodologies and innovation strategies employed to make this happen effectively. The combination of intelligent systems tools and a broad range of applications introduces a need for a synergy of disciplines from science, technology, business and the humanities. The series will include conference proceedings, edited collections, monographs, handbooks, reference books, and other relevant types of book in areas of science and technology where smart systems and technologies can offer innovative solutions.

High quality content is an essential feature for all book proposals accepted for the series. It is expected that editors of all accepted volumes will ensure that contributions are subjected to an appropriate level of reviewing process and adhere to KES quality principles.

Indexed by SCOPUS, EI Compendex, INSPEC, WTI Frankfurt eG, zbMATH, Japanese Science and Technology Agency (JST), SCImago, DBLP.

All books published in the series are submitted for consideration in Web of Science.

More information about this series at http://www.springer.com/series/8767

Anupam Biswas · Ripon Patgiri · Bhaskar Biswas
Editors

Principles of Social Networking

The New Horizon and Emerging Challenges

Springer

Editors
Anupam Biswas
Department of Computer Science
and Engineering
National Institute of Technology Silchar
Silchar, Assam, India

Ripon Patgiri
Department of Computer Science
and Engineering
National Institute of Technology Silchar
Silchar, Assam, India

Bhaskar Biswas
Department of Computer Science
and Engineering
Indian Institute of Technology (BHU)
Varanasi, India

ISSN 2190-3018 ISSN 2190-3026 (electronic)
Smart Innovation, Systems and Technologies
ISBN 978-981-16-3397-3 ISBN 978-981-16-3398-0 (eBook)
https://doi.org/10.1007/978-981-16-3398-0

This Springer imprint is published by the registered company Springer Nature Singapore Pte Ltd.
The registered company address is: 152 Beach Road, #21-01/04 Gateway East, Singapore 189721, Singapore

Preface

With the rapid growth of social networking platforms, the study of social networks is gaining immense popularity for both theory and practice within the discipline of Data Science. The analysis of social networks is mainly driven by the modeling of social networks as a directed or undirected graph of nodes and edges. At this epoch, every person is involved with social networking either directly or indirectly. Since today's social networks are very large, to analyze such large-scale networks modeled as graphs and extract useful information (like communities in the networks, information diffusion, important nodes in the network, etc.), networks involving heterogeneous social media and sources are modeled as complex, multidimensional, and multimodal networks. A plethora of information is shared over the social networking platforms which includes textual, image, and audio-visual data. Consequently, the massive amount of data is generated, which contains text, image, audio as well as emoji. These large-scale multimodal data pose new challenges for the researcher community in processing, managing, and discovering hidden knowledge.

This book provides new and innovative current discoveries in social networking which will contribute enough knowledge to the research community. The book includes the chapters presenting research advances in social network analysis and the issues emerged with diverse social media data. The book presents the applications of the theoretical algorithms and network models to analyze real-world social networks and the data emanating from them as well as characterize the topology and behavior of these networks. Moreover, it also provides in-depth insight on various issues, for instance, rumor detection, sentiment analysis through deep learning, and various machine learning techniques. Furthermore, the book will cover the new emerging challenges in the field of social networking, for example, multirelational, multidimensional, multi-aspect, multilayer networks, etc., to be stitched together to provide rich insight into the social networks. Lastly, the book also covers privacy and security of social media data as well as collection and management of social media data.

Silchar, India — Anupam Biswas
Silchar, India — Ripon Patgiri
Varanasi, India — Bhaskar Biswas
February 2021

Contents

Editors and Contributors

About the Editors

Anupam Biswas received his Ph.D. degree in computer science and engineering from Indian Institute of Technology (BHU), Varanasi, India. He has received his M. Tech. and B. E. Degree in computer science and engineering from Nehru National Institute of Technology Allahabad, Prayagraj, India and Jorhat Engineering College, Jorhat, Assam, India, respectively. He is currently working as an Assistant Professor in the Department of Computer Science & Engineering, National Institute of Technology Silchar, Assam. He has published several research papers in reputed international journals, conference, and book chapters. His research interests include Social network analysis, Computational music, Machine learning, Fuzzy systems, Information retrieval, and Evolutionary computation. He has served as Program Chair of International Conference on Big Data, Machine Learning and Applications (BigDML 2019). He is serving as General Chair of 25th International Symposium Frontiers of Research in Speech and Music (FRSM 2020). He is also an Editor of multi-authored book, titled "Health Informatics: A Computational Perspective in Healthcare," in the book series of Studies in Computational Intelligence, Springer.

Ripon Patgiri has received his Bachelor Degree from Institution of Electronics and Telecommunication Engineers, New Delhi in 2009. He has received his M.Tech. degree from Indian Institute of Technology Guwahati in 2012. He has received his Doctor of Philosophy from National Institute of Technology Silchar in 2019. After M.Tech. degree, he has joined as Assistant Professor at the Department of Computer Science & Engineering, National Institute of Technology Silchar in 2013. He has published numerous papers in reputed journals, conferences, and books. His research interests include distributed systems, file systems, Hadoop and MapReduce, big data, bloom filter, storage systems, and data-intensive computing. He is a senior member of IEEE. He is a member of ACM and EAI. He is a lifetime member of ACCS, India. Also, he is an associate member of IETE. He was General Chair of 6th International Conference on Advanced Computing, Networking, and Informatics (ICACNI

2018) and International Conference on Big Data, Machine Learning and Applications (BigDML 2019). He is Organizing Chair of 25th International Symposium on Frontiers of Research in Speech and Music (FRSM 2020 and International Conference on Modeling, Simulations and Applications (CoMSO 2020). He is convenor, Organizing Chair, and Program Chair of 26th annual International Conference on Advanced Computing and Communications (ADCOM 2020). He is guest editor in special issue "Big Data: Exascale computation and beyond" of EAI Endorsed Transactions on Scalable Information Systems. He is also an Editor in a multi-authored book, titled "Health Informatics: A Computational Perspective in Healthcare," in the book series of Studies in Computational Intelligence, Springer.

Bhaskar Biswas received Ph.D. in Computer Science and Engineering from Indian Institute of Technology (BHU), Varanasi. He received the B.Tech. degree in Computer Science and Engineering from Birla Institute of Technology, Mesra. He is working as Associate Professor at Indian Institute of Technology (BHU), Varanasi in the Computer Science and Engineering department. His research interests include Data Mining, Text Analysis, Machine Learning, and Social Network Analysis. He has published several research papers in reputed international journals, conference, and book chapters.

Contributors

Sara Ahajjam Data and Systems Engineering, National School of Applied Sciences of Tangier, Abdelmalek Essaadi University, Tangier, Morocco

Naveen Arora Jagran Institute of Management, Kanpur, Uttar Pradesh, India

Hassan Badir Data and Systems Engineering, National School of Applied Sciences of Tangier, Abdelmalek Essaadi University, Tangier, Morocco

Álvaro Bernabeu-Bautista Building Sciences and Urbanism Department, University of Alicante, San Vicente del Raspeig, Alicante, Spain

Anupam Biswas Department of Computer Science and Engineering, National Institute of Technology, Silchar, Assam, India

Bhaskar Biswas Department of Computer Science & Engineering, Indian Institute of Technology (BHU), Varanasi, India

Tanmoy Chakraborty IIIT-Delhi, New Delhi, India

Soumita Das Department of Computer Science and Engineering, National Institute of Technology, Silchar, Assam, India

Mahipal Jadeja Malaviya National Institute of Technology, Jaipur, India

Rucha Bhalchandra Joshi National Institute of Science Education and Research, Bhubaneswar, India

Chitra Krishnan Amity International Business School, Amity University, Noida, India

Ajay Kumar Department of Computer Science & Engineering, Indian Institute of Technology (BHU), Varanasi, India

Ponnurangam Kumaraguru IIIT-Delhi, New Delhi, India

Muralidhar Kurni Department of CSE, Anantha Lakshmi Institute of Technology and Sciences, Ananthapuram, India

Mubashir Majid Baba Department of Management Studies, University of Kashmir, Srinagar, Jammu and Kashmir, India

Jasmine Mariappan Department of Business Studies, University of Technology and Applied Sciences, Ibra, Sultanate of Oman

Pablo Martí Building Sciences and Urbanism Department, University of Alicante, San Vicente del Raspeig, Alicante, Spain

Shivansh Mishra Department of Computer Science & Engineering, Indian Institute of Technology (BHU), Varanasi, India

Subhankar Mishra Homi Bhabha National Institute, Anushaktinagar, Mumbai, India

M. Mrunalini Department of Computer Applications, MSRIT, Bangalore, India

Arindam Pal Data61, CSIRO, Sydney, Australia;
Cyber Security CRC, Canberra, Australia;
University of New South Wales, Sydney, Australia

Viraj Parimi Carnegie Mellon University, Pittsburgh, USA

Harita Reddy Surat, Gujarat, India

Sushmita Ruj Data61, CSIRO, Sydney, Australia;
Indian Statistical Institute, Kolkata, India

K. Saritha Sri Venkateswara Degree and PG College, Ananthapuram, India

Akrati Saxena Department of Mathematics and Computer Science, Eindhoven University of Technology, Eindhoven, The Netherlands

Pratishtha Saxena G. L. Bajaj Institute of Technology and Management, Greater Noida, India

Rahul Saxena Malaviya National Institute of Technology, Jaipur, India;
Manipal University Jaipur, Jaipur, India

Leticia Serrano-Estrada Building Sciences and Urbanism Department, University of Alicante, San Vicente del Raspeig, Alicante, Spain

Geocey Shejy VES Institute of Technology, Chembur, Mumbai, Maharashtra, India

Gurinder Singh Amity International Business School, Amity University, Noida, India

Rishi Ranjan Singh Department of Electrical Engineering and Computer Science, Indian Institute of Technology, Bhilai, Chhattisgarh, India

Shashank Sheshar Singh Department of Computer Science and Engineering, Bennett University, Greater Noida, India

Atul Kumar Verma Malaviya National Institute of Technology, Jaipur, India

Chapter 1
Centrality Measures: A Tool to Identify Key Actors in Social Networks

Rishi Ranjan Singh

Abstract Experts from several disciplines have been widely using centrality measures for analyzing large as well as complex networks. These measures rank nodes/edges in networks by quantifying a notion of the importance of nodes/edges. Ranking aids in identifying important and crucial actors in networks. In this chapter, we summarize some of the centrality measures that are extensively applied for mining social network data. We also discuss various directions of research related to these measures.

1.1 Introduction

Social networks are an abstraction of real-world social systems where people are represented as nodes and social relationships among them are portrayed as links between nodes. The number of links and nodes in a network is referred as the *size* and *order* of that network, respectively. The order of social networks varies a lot. It may be as small as in two digits, for example, Zachary's karate club [1]. It may be as large as in millions. Orkut, Flickr, LiveJournal [2], Facebook [3], Twitter, Instagram, etc., are examples of popular online social networks of that order. The number of active Facebook users has been reported in few billions by https://www.statista.com/ in August 2020. These networks are dynamic and are continuously changing at a fast pace. Every hour, several users are joining or leaving online social network platforms, forming new connections or blocking/deleting older relationships, giving rise to the addition/deletion of nodes and links in the corresponding networks.

Social network analysis is a sub-area within Network Science and Analysis where researchers attempt mining social network data for various applications. The books by Wasserman et al. [4], Carrington et al. [5], Scott and Carrington [6], and

R. R. Singh (✉)
Department of Electrical Engineering and Computer Science, Indian Institute of Technology, Bhilai, Chhattisgarh, India
e-mail: rishi@iitbhilai.ac.in

A. Biswas et al. (eds.), *Principles of Social Networking*,
Smart Innovation, Systems and Technologies 246,
https://doi.org/10.1007/978-981-16-3398-0_1

Knoke and Yang [7] may be referred for basic and detailed understanding of social network analysis. A book written in popular science style by Freeman [8] discusses the development of social network analysis area.

There are several research problems related to the analysis of complex networks which are also studied for social network analysis. For example, identifying important nodes and edges in a given network by defining and applying centrality measures; partitioning networks into densely connected sub-networks which are sparsely connected with each other by detecting community structures; understanding spreading patterns of ideas, memes, and information by studying information diffusion models; guessing which non-adjacent nodes have high probability of becoming adjacent in future by predicting links; etc.

This chapter aims to summarize some of the centrality measures that are extensively applied for mining social network data and identifying key actors. Experts from several disciplines have been widely using centrality measures for analyzing large as well as complex networks. These measures rank nodes/edges in networks by quantifying a notion of the importance of nodes/edges based on a given application. Therefore, the definition of importance is application-specific and it changes from one application to another. Ranking aids in identifying important and crucial actors in networks. In the last two decades, several interdisciplinary studies evolved just around the use of these measures to mine and analyze underlying network data. Identifying the best-suited measure from the pool of existing centrality measures for applications has been one of the most popular directions of research. Several other measures have been defined by either generalizing or extending the classical centrality measures. *Group-centrality measures* [9] are those measures where the goal is to rank subsets of nodes by computing collective centrality measures of subsets. *Hybrid-centrality measures* are those measures that are defined by combining different simple centrality measures for better performance.

This chapter starts with the basic notion of centrality measures. Then, we cover the definition of the traditional and few other popular centrality measures for social network analysis. We briefly mention algorithms to compute and estimate these measures. Next, we discuss various directions of research related to centrality measures. Afterward, we summarize a handful of applications of various centrality measures for analyzing real-world social networks. Finally, we conclude the chapter with a discussion on some future directions and open problems.

1.2 Centrality Measures

Centrality measures are network analysis tools to identify the most powerful, central, or important people/relationship in social networks. In this section, we discuss the traditional centrality measures which are not only popular in social networks but across all types of networks. Further, we also summarize few other centrality measures related to social networks. We use the following notations throughout the chapter. Let $G = (V, E)$ be a social network, where V denotes the set of nodes representing people and E denotes the set of links representing relationships between people. For simplicity, we discuss every centrality measure in this section in the context of undirected and unweighted social networks. It is trivial to extend these for weighted as well as directed social networks. Let n be the number of nodes (order), i.e., $|V| = n$ and m be the number of links (size), i.e., $|E| = m$. Let A be the $n \times n$ adjacency matrix of G, where the relationship between node i and node j is denoted by a_{ij}, an entry in A.

1.2.1 Traditional Centrality Measures

In this section, we discuss four traditional centrality measures: degree, closeness, betweenness, and eigenvector.

Degree Centrality: This centrality measure quantifies direct friendship support available to a node in social networks. As per this notion of power, a node's importance is assumed to be proportional to its degree [10]. The degree centrality of a node i, $DC(i)$ is defined as

$$DC(i) = \sum_{j \in V \setminus \{i\}} a_{ij},$$

where a_{ij} denotes the adjacency relationship between node i and j. The normalization factor is $n - 1$, i.e., the normalization of these values can be done by dividing the degree of nodes with $n - 1$, where n denotes the order of networks.

It is a notion of popularity in social networks. Nodes with a large number of relationships are powerful and central according to this measure and exhibit higher following, strength, and emotional support available. Such nodes are also highly exposed to flowing information or spreading disease in networks. Nodes with a small number of degree are not very popular and represent introvert personalities. The limitation of this measure is its local view of the network topology due to which it uses only limited local knowledge to decide the importance.

Closeness Centrality: This measure has been known as *status* of a node since 1959 [11]. Freeman [10] in 1979 termed it as closeness centrality. According to him, power of a person in a social network in terms of closeness centrality is inversely proportional to the sum of its distance to all the other persons in that social network. The closeness centrality of a node i is computed as

$$CC(i) = \frac{1}{\sum_{j \in V \setminus \{i\}} d_{ij}},$$

where d_{ij} denotes the shortest path length from node i to node j. $d_i j$ is also known as geodesic distance from node i to j. The normalization factor is $\frac{1}{n-1}$. Recall, n denotes the order of social networks.

Closeness centrality doesn't work in disconnected networks. Therefore, *harmonic centrality* [12, 13] may be used in its place which is a highly correlated measure with closeness centrality. Harmonic centrality measure assumes importance to be proportional to the sum of inverse of distances. It is defined as

$$HC(i) = \sum_{j \in V \setminus \{u\}} \frac{1}{d_{ij}}.$$

The closeness centrality of a node quantifies the average distance to all other nodes in the network from that node. This notion is useful to identify those nodes which receive any information that originated anywhere in the network in the least expected time. It is due to a smaller expected length from the originating node. Vice versa, any information originating at high closeness central nodes takes a small amount of the expected time to reach all other nodes. As the information reaches to closeness central nodes quickly, therefore, it is of high fidelity, i.e., with low noise in information. On the negative aspect, these nodes are prone to get infected from a spreading disease in the network faster than other nodes due to the expected shorter distance from the seed nodes for diseases and vice versa.

Betweenness Centrality: Bavelas [14] defined the notion of importance of a point in communication networks proportional to the number of shortest paths between other points that are passing through that point. It was termed as *point centrality*. Later, Anthonisse [15] and Freeman [16] introduced independently the definition of betweenness centrality. The betweenness centrality version of power and importance of a node is assumed to be proportional to the fraction of shortest paths between all possible pairs of nodes that are passing through that node [10]. The betweenness centrality of a node i is

$$BC(i) = \sum_{i,j,k \epsilon V:\{i,j,k\}=3} \frac{\sigma_{jk}(i)}{\sigma_{jk}},$$

where $\sigma_{jk}(i)$ denotes the number of shortest paths from node j to k which are passing through node i and σ_{jk} denotes the total number of shortest paths from node j to k. The normalization factor is $\binom{n-1}{2} = \frac{(n-1)(n-2)}{2}$ i.e.

$$BC(i) = \frac{2}{(n-1)(n-2)} \sum_{i,j,k \epsilon V:\{i,j,k\}=3} \frac{\sigma_{jk}(i)}{\sigma_{jk}}.$$

The definition of these measures is based on an assumption that transportation and communication happen through the shortest paths. In social network, this measure represents the brokerage power of a person. It is also a good indicator of the expected amount of communication load a node has to handle. A person with high betweenness centrality has higher control over the information flowing across the network. At the same time, that person is heavily loaded due to the reason that a major fraction of the information is traveling through him/her. In some types of flow networks (e.g., Communication networks, Gas Line networks, Power Grid networks) heavy load may also attract frequent demand for maintenance and such nodes are relatively more prone to fail resulting in a major breakdown in the network system. Several studies tried to replicate the phenomena of cascading failure[17–20] and observed that faults at highly loaded nodes may result in a cascading failure and may cause breakdown of the system.

Eigenvector Centrality: Eigenvalues and eigenvectors are the most popular analytical tools to understand the behavior of a square matrix and its linear transformations. Bonacich's [21] proposed that the eigenvector corresponding to the dominant eigenvalue may also be considered for ranking nodes. This measure assigns the importance of a node proportional to the sum of the importance of neighbors of that node. The eigenvector centrality of a node i is defined as

$$EC(i) = \sum_{j \in V \setminus \{i\}} (a_{ij} \cdot EC(j)),$$

where recall that a_{ij} denotes the adjacency relationship between node i and j. Eigenvector centrality resolves the local view-based limitation of degree centrality. This measure assumes that if a person's friends are powerful in the network, then that person will also be powerful. Nodes with higher eigenvector centrality scores denote that such nodes have connection to other powerful nodes in networks. A person with lower eigenvector centrality in a social network denotes that the friends of that person are not important and powerful. The major limitation of this measure is that it does not work well in directed acyclic networks. This measure gave basis to define one of the most popular and extensively used ranking measure PageRank which is used in Google to rank pages before giving search results. Few other centrality measures have been developed on a similar principle to eigenvector centrality which has been proven to be extremely usable for network analysis.

Several studies have analyzed and compared the above-mentioned traditional centrality measures [22–25]. It has been observed that, although the top central nodes as per these measures may differ on various networks, the ranking of all the nodes by these measures are positively correlated [26, 27]. Ranking due to degree centrality has been found to be highly correlated with the ranking due to betweenness and eigenvector centrality measures. In the next section, we summarize few other centrality measures that have been extensively used to analyze social and complex networks.

1.2.2 Other Popular Centrality Measures

This section mentions few other popular centrality measures other than the traditional ones which are used to analyze social and complex networks.

Katz Centrality: This measure can be used to estimate a person's influence in a social network. It considers the number of walks between node pairs to assign importance [28]. Mathematically, it is defined as

$$KC(i) = \sum_{k=1}^{\infty} \sum_{j \in V} \alpha^k (A^k)_{ij},$$

where $(A^k)_{ij}$ denotes the number of $k-$length walks from node i to j and α represents an attenuation factor that helps damping the effect of longer walks while computing the importance. The value of the attenuation factor is chosen such that $0 \leq \alpha \leq \frac{1}{|\lambda|}$ where λ denotes the principle eigenvalue of matrix A. Few variations and generalizations of this measure are given in [29–31].

PageRank Centrality: This centrality measure [32, 33] was introduced for the directed web page network to rank web pages for efficient searching. Google search engine came into light after this measure. Katz centrality faced an issue that if a high central node points to many other nodes, then all of those nodes also attain high centrality score. PageRank resolves this issue by diluting the contribution of the neighboring nodes using their out-degrees. The PageRank centrality of a node i is defined as

$$PRC(i) = \alpha \sum_{j \in V} \frac{a_{ij}}{D_j} PRC(j) + \beta,$$

where α and β are two constant quantities and D_j denotes the number of links outgoing from node j. Whenever there are no outgoing links from a node j, D_j is considered 1. α and β, similar as considered in [29–31], are the factors for consideration of dependency on the network topology and exogenous component respectively.

Decay Centrality: This centrality is similar to closeness and harmonic centrality and is also based on the shortest path lengths to all the nodes in a network. Harmonic centrality computes importance as the sum of the inverse of distances while this measure assigns importance proportional to the sum of an exponentially decreasing function over distance [34]. It is defined as

$$DKC(i) = \sum_{i \in V \setminus \{j\}} \delta^{d_{ij}},$$

where δ is a decay parameter such that $0 < \delta < 1$. This measure can be used in applications wherein the place of harmonic or closeness centrality, the geodesic distances have to be penalized exponentially.

Social Centrality: Recently, Saxena et al. [35] proposed a new centrality measure specific to social networks and called it *social centrality*. This measure assigns importance to a node proportional to its socializing capability to gain access to resources available on other nodes and its inter-community/intra-community ties which represent its bonding potential within its community and bridging potential to other communities. The centrality score of a node is computed by aggregating its sociability index with its bridging and bonding potential. A high central node as per this measure can easily manage access to resources available within the system due to its hierarchy and position within and across communities while a low central node may struggle for the resources.

Other Centrality Measures: Few other popular measures for analyzing social networks are mentioned next. *Information Centrality* [36] is based on all the possible paths between pairs of points and the information contained on these paths. Hage and Harary[37] introduced *eccentricity* as a centrality measure which gives larger importance to the node whose maximum geodesic distance to other nodes in the network is smaller. Brandes and Fleischer [38] defined variations of closeness and betweenness centrality called *current-flow closeness* and *current-flow betweenness* which assumes that the spread of information is like electricity, therefore, not only the shortest paths but also all the possible paths should be considered. They showed that current-flow closeness measure is same as information centrality [36]. *Diffusion centrality* is a measure to evaluate the influence of actors in a social network for spreading information [39]. It is a generalization of degree, eigenvector, and Katz centrality. *Coverage centrality* [40] is similar to betweenness centrality and it assigns importance to a node proportional to the number of pairs of nodes between which at least one shortest path passes through that node.

1.3 Directions of Research

In this section, we discuss various directions of research related to centrality measures in social and complex networks. We start with approaches for the exact computation of traditional measures. The computation of few kinds of centrality scores has been realized to be expensive in terms of time over large networks. Several studies have been conducted that focused on the fast estimation of those types of centrality scores to tackle the issue. We summarize a few of such literature on the estimation of tradition centrality measures. Next, we focus on the problem of computing and keeping centrality measures up to date in dynamic networks. These types of networks evolve over time. Algorithms for such kind of networks are called dynamic algorithms and we brief few related literature on centrality measures. Few of the recent studies on estimation algorithms over dynamic networks are mentioned next. Afterward, parallel and distributed algorithms for speeding up and scaling computation of traditional measures are mentioned. Although computing top-k central nodes as per a centrality measure is a widely studied problem, but recently, researchers started designing fast algorithms for ordering/ranking a set of arbitrary nodes in a large network based on some centrality measure. We note down few studies on both types of problems. Further, few generalizations of centrality measures considering weights either on the edges or on the nodes or on both have been discussed. Some applications require computing cumulative centrality scores of a set of nodes than computing individual scores. These measures are known as group centrality and few related studies are briefed next. Hybridization of centrality measures to analyze social and complex networks is another direction. A graph-editing-based problem on the improvement or maximization of centrality scores is discussed next. Finally, some applications of centrality measures in social networks are summarized.

1.3.1 Exact Computation

In this section, we discuss approaches to compute exact traditional centrality scores of nodes. The exact algorithm to compute degree centrality is very trivial which requires $O(n)$ time for computing the degree of one node and $O(m)$ time for computing the degree of all nodes. Recall that n denotes the order of a network and m denotes the size of a network. A simple exact algorithm to compute closeness centrality score of a node is based on Dijkstra's single-source shortest path (SSSP) computation algorithm [41] which takes $O(m + n \log n)$ and $O(m)$ time in weighted and unweighted networks, respectively. Closeness scores of all nodes can be computed using either SSSP computation from all nodes requiring $O(mn + n^2 \log n)$ and $O(mn)$ time in weighted and unweighted networks, respectively, or all pair shortest path (APSP) computation using Floyd-Warshall's algorithm [42, 43] which takes O(n^3)time. Sariyüce et al. [44] gave a framework to compute closeness centrality faster than the trivial approach. Their proposed framework modifies a network by compressing and splitting it into small sub-networks in which centrality scores can be computed independently. Their proposed algorithm empirically outperformed competitive algorithms by several folds.

Kintali [45] conjectured that the exact betweenness score computation of a node is as time-consuming as computing betweenness scores of all nodes. Similar to closeness centrality, all the algorithms to compute betweenness scores are either based on SSSP computation from all nodes or APSP computation. A modified version of the Floyd–Warshall's APSP computation algorithm [42, 43] is the most trivial algorithm to compute exact betweenness scores for one as well as all nodes. As stated above, this approach takes $O(n^3)$ time. Computation of betweenness scores takes $O(n^3)$ even when SSSP computation from all nodes is used. It is due to the reason that even when the number of the shortest path between all the pair of nodes is given, computation of betweenness formulation still takes $O(n^3)$ time. Brandes [46] reformulated the definition of betweenness centrality in terms of summing up *dependency* value. Dependency of a node i on node j denotes the contribution of the shortest paths originating at node i in the betweenness score of node j. His algorithm was based on a modification to Dijkstra's [41] algorithm. Due to the new formulation, it started computing exact betweenness score in the same asymptotic time whatever was required for running Dijkstra's [41] algorithm from all nodes. Although, several faster algorithms by Baglioni et al. [47], Puzis et al. [48], Sariyüce et al. [49], Erdos et al. [50], Chehreghani et al. [51], Bentert et al. [52], and Daniel et al. [53] have been proposed that attempted to reduce the time to compute betweenness score empirically or theoretically on some special type of networks, but so for no algorithm guarantees to perform asymptotically better than the algorithm by Brandes [46].

Eigenvector centrality scores can be computed using the power method [54]. The power method starts with a vector whose euclidean norm is 1. It is an initial approximation of the eigenvector with respect to the dominant eigenvalue. Each iteration of this method takes the resulting vector from the previous iteration as input and multiplies it with the adjacency matrix of the network under consideration to improve the approximation. The convergence of this method is certain if the adjacency matrix has a dominant eigenvalue. The time for convergence relies on the ratio between the absolute values of the dominant and the second dominant eigenvalues. The same method is also used to compute PageRank and some other variants of eigenvector centrality. A basic foundation for algorithms to compute traditional centrality scores is given in Chap. 4 in the book by Brandes and Erlebach [55].

1.3.2 Estimation

Even the best algorithms for computing exact centrality scores in large social networks might be time-consuming. To overcome this limitation, researchers developed several estimation (approximation) algorithms that take a relatively lesser amount of time than exact algorithms and compute approximate values of centrality scores. Most of the estimation approaches are sampling-based. In the sampling technique, in place of conducting computation based on every member from a set of entities, a subset of entities is chosen and then estimated values are computed based on that subset. Sampling may be uniform or nonuniform. We briefly discuss few such studies next. The exact computation of the degree centrality of a node as well as all the nodes is very efficient, therefore, there does not seems any requirement of an estimation algorithm. Though when one wants to know a node's rank using only the local information, a need for a rank estimation algorithm arises even for degree centrality. Details about rank estimation algorithms are given in Sect. 1.3.6.

Eppstein and Wang [56] proposed a node sampling-based algorithm to estimate closeness centrality scores of all the nodes and gave theoretical bounds on the error in estimating scores. The idea was to sample a few nodes from the set of all nodes and consider the single-source shortest path computation (SSSP) from only the chosen nodes (also called pivot) for centrality computation. Ohara et al. [57] proposed a similar algorithm to estimate closeness centrality scores as by Eppstein and Wang [56], but they gave a different theoretical analysis than [56]. Rattigan et al. [58] proposed to create a network structure index for efficient estimation of closeness centrality and betweenness centrality. Cohen et al. [59] gave a scalable algorithm for estimating closeness centrality scores on undirected as well as directed graphs. A group testing-based algorithm for identifying top closeness central nodes was given by Ufimtsev and Bhowmick [60]. Murai [61] gave pivot guided estimation-based algorithm to estimate closeness centrality scores in undirected networks and strongly connected directed networks. His algorithm outperformed the estimation algorithms in [56, 59] theoretically as well as empirically.

The time to compute a node's betweenness has been conjectured to asymptotically similar to the time to compute betweenness score of all the nodes. There are two classes of estimation algorithms for betweenness centrality. The first one focuses on estimating the scores of all the nodes together while the other one just estimates betweenness score of a particular node. Brandes and Pich [62] used a similar idea as used by Eppstein and Wang [56], for estimating betweenness centrality measure. An adaptive node sampling-based algorithm was proposed by Bader et al. [63] to estimate a node's betweenness score. A theoretical bound on the error was also provided. Geisberger et al. [64] proposed a generalization of the algorithm coined by Brandes and Pich [62] which achieved better results. Most of the studies discussed use randomization algorithms, but Gkorou et al. [65] and Ercsey-Ravasz et al. [66] proposed a deterministic estimation algorithm. They gave an estimation algorithm for computation of the betweenness scores in large networks by considering only the shortest paths of length k. A comparative analysis of Gkorou et al.'s algorithm [65] with Geisberger et al.'s [64] and Brandes and Pich's [62] algorithm is done in [67]. Ohara et al. [57] also studied estimation of betweenness centrality in addition to closeness centrality and did bound analysis for error in estimation. Riondato and Kornaropoulos [68] proposed two randomized algorithms based on the sampling of shortest paths to estimate betweenness scores. Chehreghani [69] used non-uniform node sampling to estimate a nodes' betweenness score. Agarwal et al. [70] analyzed random graphs and proposed another non-uniform node sampling-based estimation algorithm which performed better than [69] and other competitive algorithms to estimate betweenness centrality score of a node. Their estimation algorithm was further applied to solve betweenness-ordering problem [71]. Due to popularity and wide applicability, most of the estimation algorithms for centrality measures are for betweenness measure. Several recent studies approximately compute betweenness scores [72–77]. A review of approximation algorithms for computing betweenness centrality has been done by Matta et al. [78].

Wink et al. [79] presented an algorithm to estimate voxel-wise eigenvector centrality scores in fMRI data. Kumar et al. [80] gave an estimation algorithm for eigenvector centrality and PageRank based on neural networks. Charalambous et al. [81] proposed a distributed approach to efficiently estimate eigenvector centrality of nodes in directed networks. Ruggeri and De Bacco [82] gave an algorithm on incomplete graphs to estimate eigenvector centrality scores. Their estimation algorithm is based on a sampling idea derived from spectral approximation theory. Mitliagkas et al. [83] proposed a fast approximation algorithm to estimate PageRank.

1.3.3 Updating Centrality Scores

Real-world networks are large and dynamic. Therefore, to maintain updated centrality scores of nodes, applying exact algorithms after every or even a few number of updates in batches can be impractical when the exact algorithms are time-consuming on large networks. There can be a significant difference in the ranking of vertices before and after an update [84]. Updates can be insertion/deletion of edges or nodes or increase/decrease in edge weights. Algorithms designed to update values of some attributes on nodes/edges or some other network properties in case of updates in networks faster than re-computing scores using exact algorithms are called *dynamic algorithms*. Algorithms tackling different nature of updates are categorized as incremental, decremental, or fully dynamic algorithm. In this section, we briefly mention some dynamic algorithms for traditional centrality measures.

Kas et al. [85] proposed an incremental approach to update closeness scores in evolving social networks after addition/removal of links and nodes. Sarıyüce et al. [86] also gave an incremental approach for computing closeness scores after edge insertion/deletions. Yen et al. [87] also proposed a dynamic algorithm to update closeness centrality scores after edge insertion/deletion. The basic idea used in their algorithms is to efficiently identify nodes whose closeness centrality will change after a link update. Wei and Carley [88] proposed an online algorithm framework to update closeness and betweenness scores after link updates. Khopkar et al. [89] proposed an incremental algorithm for all pair shortest paths and used the idea to develop incremental algorithm for closeness and betweenness centrality. Sarıyüce et al. [90] also gave an incremental algorithm to compute closeness centrality scores in dynamic networks relying on a distributed memory framework. Santos et al. [91] proposed a scalable algorithm for updating closeness centrality scores after deletion of edges. A dynamic algorithm to compute a node's closeness centrality in social networks that are evolving with time is given by Ni et al. [92]. Most of the algorithms to compute closeness centrality are based on network topology and structures while their idea relies on temporal network features. Shao et al. [93] recently gave a dynamic algorithm to compute closeness. They proposed to calculate the exact closeness centrality scores by using bi-connected blocks and articulation vertices. Their approach is to detect all the shortest paths that are affected and then update the centrality value based on articulation vertices.

Vignesh et al. [94] considered a related problem that requires updating betweenness centrality scores after addition or deletion of nodes. Lee et al. [95] gave a dynamic algorithm to tackle link updates (addition/deletion). Their approach was based on the observation that whenever an update happens within a bi-connected component, the re-computation of centrality scores is required only for the nodes in that bi-connected block. Betweenness scores of nodes outside that block can be updated very efficiently without a need of re-computation. For node updates, they suggested to use their proposed algorithm for every link incident on nodes under consideration. Green et al. [96] proposed an incremental approach for betweenness centrality in case of a series of link addition over time. The idea used in their algorithm was to maintain and update breadth-first tree data structure rooted at every vertex. Kas et al. [97] also gave an incremental algorithm to tackle updates for nodes and edges in the form of change in edge weights. Their idea was based on Ramalingam and Reps' [98] incremental approach for updating all pair shortest paths (APSPs). Further, they extended their incremental approach for a variant of betweenness centrality where the shortest paths of at most k lengths are considered for computation of betweenness centrality scores [99]. An incremental algorithm similar to the one in [96] was given by Nasre et al. [100]. Their algorithm tackled node as well as edge updates and used breadth-first search-based directed cyclic graphs (BFS DAGs) as the data structure in place of BFS trees. Later, Nasre et al. [101] proposed a decremental approach for updating all pair all shortest paths (APASPs) extending the approach by Demetrescu and Italiano [102] for APSPs which founded the basis for a decremental approach to update betweenness scores. Kourtellis et al. [103] proposed a scalable online algorithm to update betweenness scores of nodes and edges in case of edge addition/deletion. Pontecorvi and Ramachandran [104] gave a fully dynamic algorithm for updating APASPs and extended the algorithm to develop a fully dynamic approach to update betweenness scores [105, 106]. A dynamic algorithm to update betweenness scores after node addition and deletion, similar to Lee et al.'s [95] approach, was given in [107, 108]. Hayashi et al. [109], Bergamini et al. [110], and Tsalouchidou et al. [111] also gave dynamic algorithms to update betweenness scores.

More work has been done to update PageRank Centrality and Katz Centrality in dynamic networks than traditional Eigenvector Centrality. Bahmani et al. [112], Rossi and Gleich [113], Rozenshtein et al. [114], and recently Zhan et al. [115] gave algorithms to update PageRank in dynamic networks. Nathan and Bader [116] gave an algorithm to update Katz centrality scores in a dynamic network. There have been studies to update various other centrality scores in dynamic networks. For example, Sarmento [117] gave an incremental algorithm to update laplacian centrality measures. Recent studies on updating centrality scores in dynamic networks show that the direction is still open for a more efficient algorithm for various centrality measures.

Although dynamic graphs and algorithm on dynamic graphs have been studied extensively in the last few decades, in the last decade, it has been explored in several studies under a new name called *temporal networks* [118, 119] or time-dependent graphs [120]. On such types of networks, the question of identifying important nodes and edges are done with the help of centrality measures for temporal networks [121]. For example, closeness centrality [122], betweenness centrality [111], eigenvector centrality [123], random-walk centrality [124], pagerank centrality [125], etc., have been studied over temporal networks.

1.3.4 Approximation Algorithms for Dynamic Graphs

Several literature on centrality measures studied either dynamic algorithms or approximation algorithms for the computation of centrality scores in the last two decades. Recently, the problem of updating estimated centrality scores in dynamic networks came to light. Bergamini et al. [126], Bergamini and Meyerhenke [127], Riondato et al. [128], and Chehreghani et al. [129] proposed approaches for updating approximated betweenness scores in dynamic networks. Zhang et al. [130] gave such kind of approach for personalized PageRank centrality, while Nathan and Bader [131] gave for personalized Katz centrality measure.

1.3.5 Parallel and Distributed Computation

Real-world networks are very large in size. The computation of centrality scores for closeness, betweenness, and similar measures require asymptotically quadratic or cubic time in the order of networks. These kinds of computations are time-consuming when implemented sequentially. A similar order of time is required to keep the scores up to date when network topology changes over time. Parallel computing has been proven as one of the best methods to reduce time for computation whenever algorithms support parallelism by utilizing super-computing resources. Distributed computing is a popular tool to perform large-scale computation. Distributed algorithms for centrality measures aim to compute centrality scores at each node using the information attained by those nodes based on the interactions with their neighbors. Due to this reason, distributed algorithms also face a challenge to exactly compute those centrality measures that require information of the whole network.

Several literature in the last two decades study parallel and distributed computation of centrality measures over static as well as dynamic networks. Bader et al. [132] studied parallel algorithms for computing most of the traditional centrality measures. A recent study on parallel computation of these measures is [133]. Santos et al. [91, 134] gave algorithms for computation of closeness centrality in a parallel setting. Shukla et al. [135] gave parallel approaches for computing closeness and betweenness centrality scores in dynamic networks. Wang and Tang [136, 137] have given distributed algorithms for tree and general networks. You et al. [138] have considered the problem of computing traditional centrality measure in distributed environment. Most of literature related to parallel and distributed compution of centrality measures are for betweenness centrality. It is due to the factor that the computation of a node's betweenness score is time consuming and the scalability of the computation is challenging. Following are literature on computation of exact and approximate betweenness scores in parallel [45, 53, 86, 139–149] and distributed [150–153] frameworks.

1.3.6 Centrality Ordering and Ranking

Most of the algorithms for centrality measures compute or estimate scores to rank nodes. Some applications may demand ranking of top k nodes with high centrality scores, while others may want to rank a set of arbitrarily picked nodes. The first problem is called *top-k central node* computation, while the later one is known as *centrality-ordering* problem [71]. A solution to the above problems may output exact or estimated ranks. Several studies on ranking all nodes, estimating a node's rank, finding top-k central nodes, or ordering k arbitrarily chosen nodes based on various centrality measures have been done.

Bian et al. [154] recently conducted a survey on the identification of the top k nodes based on degree centrality, closeness centrality, and influence for diffusion. Studies on ranking of the top k central node based on closeness centrality [155, 156], betweenness centrality [68, 157, 158], and Katz centrality [159] may be referred. Kumar et al. [80] gave rank estimation algorithm on the basis of eigenvector centrality and PageRank based on neural networks. Computation of degree centrality of a node is very efficient but identifying rank of a node based on degree centrality requires larger computation. Saxena et al. [160] proposed methods to estimate a node's degree rank. Computing closeness centrality of a node takes relatively a lot smaller time than closeness rank of that node. Saxena et al. [161] gave a heuristic to estimate a node's closeness rank. Kumar et al. [80] gave a neural network-based rank estimation algorithm on the basis of eigenvector centrality and PageRank.Singh et al. [71] introduced centrality-ordering problem and gave an efficient algorithm to estimate betweenness-ordering. They motivated for an open direction related to the study of ordering problem on other centrality measures.

1.3.7 Weighted Centrality Measures

The common practice of defining centrality measures is to first introduce them for unweighted network, i.e., every actor or entity represented by nodes are assumed to have same features and relationships between actors are also assumed to be uniform. These measures are called unweighted centrality measures. Definitions of the traditional centrality measures given in Sect. 1.2.1 are for unweighted networks. In some of the networks, weights on the edges are given and to better analyze the network, it becomes essential to use the weights. The definition of centrality measures that consider weights on the edges while computing the scores are called edge-weighted centrality measures. Although, the weights on the edges are considered for analysis, yet the weights on nodes are still assumed to be uniform. Most of the weighted version of the centrality measures are defined only considering the edge weights [13, 26, 162–164]. The edge-weighted degree centrality has been used in several applications in biological network to identify crucial nodes [165–167].

Similarly, some studies defined node-weighted centrality measures by considering node weights and uniform weights on the edges [168–173]. These studies suggested to combine the edge-weighted version of the definition of the centrality measures to get fully weighted centrality measures that can analyze networks while considering weights on the edges as well as on the nodes. The assumption of uniform weights on the nodes and the edges is to simplify the analysis of networks. It is highly unlikely that all the actors in a network possess the same characteristics and features. We are surrounded by fully weighted networks but edge weights are easily available in comparison to node weights. Due to privacy and security concerns, actor do not share details about their personal information which makes it difficult and complicated to map characteristics/attributes/features of actors in the form of node weights. Relationship data is relatively easily available and, therefore, it becomes relatively easy to consider edge weights. In other types of complex networks, similar constraints exist. Singh et al. [171, 172] proposed a way to overcome the difficulty in figuring out node weights. They suggested to apply appropriate measures to compute weights on nodes and then apply node-weighted or fully weighted centrality measures.

1.3.8 Group Centrality Measures

The idea of centrality measures for individual nodes/edges has been extended to compute collective centrality scores of a group of nodes/edges by Everett and Borgatti [9]. These type of centrality measures identify a set/group/class of nodes or edges which collectively dominate other sets/groups/classes on the basis of a quantitative notion of importance. An application-oriented study of this variant of degree, closeness, and betweenness centrality has been conducted by Ni et al. [174]. Zhao et al. [175] gave an efficient algorithm to compute group closeness centrality for disk-resident networks. Chen et al. [176] shown that the problem of finding a group of k nodes whose collective closeness centrality is maximum, is an NP-hard problem to solve. Group betweenness centrality has been used in multiple applications [177, 178] and to compute or estimate group betweenness centrality, several algorithms have been proposed [129, 179–181].

1.3.9 Hybrid Centrality Measures

Individual centrality measures might not appear as a fruitful tool for analyzing some complex systems which has a mixed notion of importance. For networks based on such systems, hybrid centrality measures are used. Hybrid centrality measures are defined by combining more than one measure to produce a better rank than individual ranks by each measure in the combination. In this section, we briefly mention few literature on hybridizing centrality measures. Few of these can be used in the analysis of social networks.

In recent studies by Singh et al. [171, 172], a hybridization methodology of centrality measures is proposed based on the formulation of node-weighted measures. Singh et al. [171, 172] proposed to generate weights on nodes based on a centrality measure, and then use the generated weights while computing node-weighted version of another centrality measure. They also applied these measures for two applications. One of the demonstrated applications of such hybridization was to find influential spreaders in a complex contagion scenario. Abbasi and Hossain [169] proposed a new set of hybrid centrality measures by hybridizing traditional centrality measures within the framework of degree centrality. They applied their hybrid measures on a co-authorship network and noted that the hybrid measures performed differently than the traditional centrality measures and further noticed that the newly proposed measures were significantly correlated to authors' performance. Abbasi [168] proposed hybrid measures on weighted collaboration networks based on h-index, a-index, and g-index. These indices (h-index, a-index, g-index) are considered as traditional collaborative performance measures to rank authors. The proposed measure gave results highly correlated with ranking based on citation-count and publication-count. The two quantities, citation-count and publication-count, are widely used and well-established performance measures for scholars. All of the three studies mentioned above proposed hybrid measures that have application for analyzing social networks of different nature.

Linear combination is a popular strategy for combining values across several disciplines. Qiu et al. [182] has defined a hybridization-based on this principle to mix cohesion centrality and degree centrality. This hybridization was further used by Li-Qing et al. [183] for detecting community structures. A hybridization of closeness centrality and betweenness centrality was proposed by Zhang et al. [184] rank nodes in satellite communication networks. In another study, a hybridization of degree centrality, a variation of traditional closeness centrality for networks with more than one connected component and betweenness centrality measures was proposed by Buechel and Buskens [185]. A hybrid page-ranking approach based on the traditional centrality measures was proposed by Qiao et al. [186]. Lee and Djauhari [95] also had proposed a linear combination-based hybridization of traditional centrality measures which was applied to identify highly significant and influential stocks. In an early study by Wang et al. [187], a hybridization of degree centrality, betweenness centrality, and degree of neighbors was proposed.

1.3.10 Centrality Improvement and Maximization

A graph-editing problem related to centrality measures is to improve or maximize centrality score of a node by adding links. Several literature in the last two decades study centrality improvement or maximization problems. Avrachenkov and Litvak [188] studied the change in pagerank scores due to link addition. Later page-rank maximization problem using addition of new outgoing links [189] or new incoming links [190] was considered. Maximization of eccentricity centrality [191, 192] was studied soon after. Further, the centrality improvement and maximization problem was considered for other centrality measures: Closeness and Harmonic centrality [193], Betweenness centrality [194, 195], Information Centrality [196], and Coverage centrality [197, 198]. This nature of graph editing problem has also been explored for maximization of group centrality measures [198, 199].

1.3.11 Application

Centrality measures have been widely used to analyze social and complex networks. In this section, we brief few applications on social networks. Girvan and Newman [200] have suggested to use edge betweenness centrality to detect community structures in social and complex networks. Yan and Ding [201] have applied degree, closeness, betweenness, and PageRank centrality in a co-authorship network for impact analysis. Ghosh and Lerman [202] have analyzed that a variation of Katz centrality turns out to be a good predictor of influence in online social networks. Ilyas and Radha [203] have used centrality measures to identify influential nodes in an online friendship network from Orkut and a gaming network from Facebook. Mehrotra et al. [204] have proposed to use centrality measures for the detection of fake followers on Twitter. Riquelme and González-Cantergiani [205] have conducted a survey on various measures including centrality to evaluate user's influence on Twitter.

Few of the recent applications of centrality measures are summarized next. Eigenvector centrality can be used to analyze fMRI data of the human brain to identify connectivity pattern [206]. In a study by Zinoviev [207], a social network is formed of Russian Kompromat has been analyzed using the traditional centrality measures which identified Vladimir Putin as the top central kompromat figure. Kim et al. [208] used normalized closeness and betweenness centrality measures on a word network derived from users' posts on Reddit to analyze the perspective of public toward renewable energy and identifying frequent issues related to renewable energy. Nurrokhman et al. [209] has recently used degree, closeness, and betweenness centrality to analyze the collaboration within students for sharing knowledge. Stelzhammer in his thesis [210] attempts to improve the detection of influential users in a recommender system using centrality measures. Trach and Bushuyev [211] have used degree, betweenness, eigenvector, and PageRank centrality measures to analyze a social network between project participants for the construction of a residential building located in Ukraine. Yuan [212] have used degree centrality and structural holes to analyze and forecast tourist arrivals in a tourism social network. Neuberger [213] has analyzed relationship between the actors and directors from the Soviet film industry. Nagdive et al. [214] have used centrality measures to identify key organizations, places, and persons in a terrorist network.

1.3.12 Defining New Centrality Measures

The above sections brief about various research directions for computing and applying centrality measures for analyzing social and complex networks. The last direction that existed since the beginning of the study on centrality measures is to define a new measure when other measures don't seem useful enough. This had led us to a point today when there is an abundance of centrality measures. A web page (*http://schochastics.net/sna/periodic.html*) contains a list of several centrality measures in an interesting representation. It remains open to define new centrality measures that perform better than the existing measures and provide a more insightful analysis of social and complex systems.

1.4 Conclusion

Centrality measures have been a popular tool to mine social network data. In this chapter, we have reviewed various directions of research related to computing centrality measures and applying these in the identification of key actors in social as well as complex networks. Most of the research directions are still evolving and comprise open problems to improve existing approaches, design new algorithms that outperform previous ones and solve new problems related to the computation of various centrality measures.

References

1. Zachary, W.W.: An information flow model for conflict and fission in small groups. J. Anthropol. Res. **33**(4), 452–473 (1977)
2. Mislove, A., Marcon, M., Gummadi, K.P., Druschel, P., Bhattacharjee, B.: Measurement and analysis of online social networks. In: Proceedings of the 7th ACM SIGCOMM Conference on Internet Measurement, pp. 29–42 (2007)
3. Traud, A.L., Mucha, P.J., Porter, M.A.: Social structure of Facebook networks. Phys. A **391**(16), 4165–4180 (2012)
4. Wasserman, S., Faust, K., et al.: Social Network Analysis: Methods and Applications. Cambridge University Press (1994)
5. Carrington, P.J., Scott, J., Wasserman, S.: Models and Methods in Social Network Analysis. Cambridge University Press (2005)
6. Scott, J., Carrington, P.J.: The SAGE Handbook of Social Network Analysis. SAGE Publications (2011)
7. Knoke, D., Yang, S.: Social Network Analysis. SAGE Publications (2019)
8. Freeman, L.: The Development of Social Network Analysis, vol. 1. Empirical Press (2004)
9. Everett, M.G., Borgatti, S.P.: The centrality of groups and classes. J. Math. Sociol. **23**(3), 181–201 (1999)
10. Freeman, L.C.: Centrality in social networks conceptual clarification. Soc. Netw. **1**(3), 215–239 (1979)
11. Harary, F.: Status and contrastatus. Sociometry, 23–43 (1959)
12. Boldi, P., Vigna, S.: Axioms for centrality. Internet Math. **10**(3–4), 222–262 (2014)
13. Opsahl, T., Agneessens, F., Skvoretz, J.: Node centrality in weighted networks: generalizing degree and shortest paths. Soc. Netw. **32**(3), 245–251 (2010)
14. Bavelas, A.: A mathematical model for group structures. Hum. Organ. **7**(3), 16–30 (1948)
15. Anthonisse, J.M.: The rush in a directed graph. Stichting Mathematisch Centrum. Mathematische Besliskunde (BN 9/71), 1–10 (1971)
16. Freeman, L.C.: A set of measures of centrality based on betweenness. Sociometry **40**(1), 35–41 (1977)

17. Buldyrev, S.V., Parshani, R., Paul, G., Stanley, H.E., Havlin, S.: Catastrophic cascade of failures in interdependent networks. Nature **464**(7291), 1025–1028 (2010)
18. Kinney, R., Crucitti, P., Albert, R., Latora, V.: Modeling cascading failures in the north american power grid. Eur. Phys. J. B-Condens. Matter Complex Syst. **46**(1), 101–107 (2005)
19. Lin, G., Di, Z., Fan, Y.: Cascading failures in complex networks with community structure. Int. J. Mod. Phys. C **25**(05), (2014)
20. Motter, A.E., Lai, Y.C.: Cascade-based attacks on complex networks. Phys. Rev. E **66**(6), 065102 (2002)
21. Bonacich, P.: Factoring and weighting approaches to status scores and clique identification. J. Math. Sociol. **2**(1), 113–120 (1972)
22. Borgatti, S.P.: Centrality and network flow. Soc. Netw. **27**(1), 55–71 (2005)
23. Borgatti, S.P., Everett, M.G.: A graph-theoretic perspective on centrality. Soc. Netw. **28**(4), 466–484 (2006)
24. Das, K., Samanta, S., Pal, M.: Study on centrality measures in social networks: a survey. Soc. Netw. Anal. Min. **8**(1), 13 (2018)
25. Landherr, A., Friedl, B., Heidemann, J.: A critical review of centrality measures in social networks. Bus. Inf. Syst. Eng. **2**(6), 371–385 (2010)
26. Lee, J.-Y.: Centrality measures for bibliometric network analysis. J. Korean Soc. Libr. Inf. Sci. **40**(3), 191–214 (2006)
27. Valente, T.W., Coronges, K., Lakon, C., Costenbader, E.: How correlated are network centrality measures? Connect. (Toronto, Ont.) **28**(1), 16 (2008)
28. Katz, L.: A new status index derived from sociometric analysis. Psychometrika **18**(1), 39–43 (1953)
29. Bonacich, P.: Power and centrality: a family of measures. Am. J. Sociol. **92**(5), 1170–1182 (1987)
30. Bonacich, P., Lloyd, P.: Eigenvector-like measures of centrality for asymmetric relations. Soc. Netw. **23**(3), 191–201 (2001)
31. Hubbell, C.H.: An Input-Output Approach to Clique Identification, pp. 377–399. Sociometry (1965)
32. Brin, S., Page, L.: The Anatomy of a Large-Scale Hypertextual Web Search Engine (1998)
33. Page, L., Brin, S., Motwani, R., Winograd, T.: The Pagerank Citation Ranking: Bringing Order to the Web. Technical report, Stanford InfoLab (1999)
34. Matthew, O.: Jackson. Social and Economic Networks. Princeton University Press, Princeton, NJ, USA (2008)
35. Saxena, R., Kaur, S., Bhatnagar, V.: Social centrality using network hierarchy and community structure. Data Min. Knowl. Disc. **32**(5), 1421–1443 (2018)
36. Stephenson, K., Zelen, M.: Rethinking centrality: methods and examples. Soc. Netw. **11**(1), 1–37 (1989)
37. Hage, P., Harary, F.: Eccentricity and centrality in networks. Soc. Netw. **17**(1), 57–63 (1995)
38. Brandes, U., Fleischer, D.: Centrality measures based on current flow. In: Annual Symposium on Theoretical Aspects of Computer Science, pp. 533–544. Springer (2005)
39. Banerjee, A., Chandrasekhar, A.G., Duflo, E., Jackson, M.O.: The diffusion of microfinance. Science **341**(6144) (2013)
40. Yoshida, Y.: Almost linear-time algorithms for adaptive betweenness centrality using hypergraph sketches. In: Proceedings of the 20th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, pp. 1416–1425 (2014)
41. Dijkstra, E.W.: A note on two problems in connexion with graphs. Numer. Math. **1**(1), 269–271 (1959)
42. Floyd, R.W.: Algorithm 97: shortest path. Commun. ACM **5**(6), 345 (1962)
43. Warshall, S.: A theorem on boolean matrices. J. ACM (JACM) **9**(1), 11–12 (1962)
44. Sariyüce, A.E., Kaya, K., Saule, E., Çatalyürek, Ü.V.: Graph manipulations for fast centrality computation. ACM Trans. Knowl. Discov. Data (TKDD) **11**(3), 1–25 (2017)
45. Kintali, S.: Betweenness centrality: algorithms and lower bounds. arXiv:0809.1906 (2008)

46. Brandes, U.: A faster algorithm for betweenness centrality. J. Math. Sociol. **25**(2), 163–177 (2001)
47. Baglioni, M., Geraci, F., Pellegrini, M., Lastres, E.: Fast exact and approximate computation of betweenness centrality in social networks. In: State of the Art Applications of Social Network Analysis, pp. 53–73. Springer (2014)
48. Puzis, R., Elovici, Y., Zilberman, P., Dolev, S., Brandes, U.: Topology manipulations for speeding betweenness centrality computation. J. Complex Netw. **3**(1), 84–112 (2015)
49. Sariyüce, A.E., Saule, E., Kaya, K., Çatalyürek, Ü.V.: Shattering and compressing networks for betweenness centrality. In: SIAM Data Mining Conference (SDM). SIAM (2013)
50. Erdos, D., Ishakian, V., Bestavros, A., Terzi, E.: A divide-and-conquer algorithm for betweenness centrality. arXiv:1406.4173 (2014)
51. Chehreghani, M.H., Bifet, A., Abdessalem, T.: Efficient exact and approximate algorithms for computing betweenness centrality in directed graphs. In: Pacific-Asia Conference on Knowledge Discovery and Data Mining, pp. 752–764. Springer (2018)
52. Bentert, M., Dittmann, A., Kellerhals, L., Nichterlein, A., Niedermeier, R.: An adaptive version of brandes' algorithm for betweenness centrality. arXiv:1802.06701 (2018)
53. Daniel, C., Furno, A., Zimeo, E.: Cluster-based computation of exact betweenness centrality in large undirected graphs. In: 2019 IEEE International Conference on Big Data (Big Data), pages 603–608. IEEE (2019)
54. Wilkinson, J.H.: The Algebraic Eigenvalue Problem, vol. 87. Clarendon press Oxford (1965)
55. Ulrik, B., Thomas, E.: Network Analysis: Methodological Foundations, vol. 3418. Springer (2005)
56. Eppstein, D., Wang, J.: Fast approximation of centrality. J. Graph Algorithms Appl. **8**, 39–45 (2004)
57. Ohara, K., Saito, K., Kimura, M., Motoda, H.: Resampling-based framework for estimating node centrality of large social network. In: Discovery Science, pp. 228–239. Springer (2014)
58. Rattigan, M.J., Maier, M., Jensen, D.: Using structure indices for efficient approximation of network properties. In: Proceedings of the 12th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, pp. 357–366 (2006)
59. Cohen, E., Delling, D., Pajor, T., Werneck., R.F.: Computing classic closeness centrality, at scale. In: Proceedings of the Second ACM Conference on Online Social Networks, pp. 37–50 (2014)
60. Ufimtsev, V., Bhowmick, S.: An extremely fast algorithm for identifying high closeness centrality vertices in large-scale networks. In: IA3@ SC, pp. 53–56 (2014)
61. Murai, S.: Theoretically and empirically high quality estimation of closeness centrality. In: 2017 IEEE International Conference on Data Mining (ICDM), pp. 985–990. IEEE (2017)
62. Brandes, U., Pich, C.: Centrality estimation in large networks. Int. J. Bifurc. Chaos **17**(07), 2303–2318 (2007)
63. Bader, D.A., Kintali, S., Madduri, K., Mihail, M.: Approximating betweenness centrality. In: Proceedings of the 5th International Conference on Algorithms and Models for the Web-Graph, WAW'07, pp. 124–137. Springer, Berlin (2007)
64. Geisberger, R., Sanders, P., Schultes, D.: Better Approximation of Betweenness Centrality, vol. 8, pp. 90–100 (2008)
65. Gkorou, D., Pouwelse, J., Epema, D., Kielmann, T., van Kreveld, M., Niessen, W.: Efficient approximate computation of betweenness centrality. In: 16th Annual Conference of the Advanced School for Computing and Imaging (ASCI 2010) (2010)
66. Ercsey-Ravasz, M., Lichtenwalter, R.N., Chawla, N.V., Toroczkai, Z.: Range-limited centrality measures in complex networks. Phys. Rev. E **85**(6), 066103 (2012)
67. Gkorou, D., Pouwelse, J., Epema, D.: Betweenness centrality approximations for an internet deployed p2p reputation system. In: Parallel and Distributed Processing Workshops and Phd Forum (IPDPSW), IEEE International Symposium on 2011, pp. 1627–1634. IEEE (2011)
68. Riondato, M., Kornaropoulos, E.M.: Fast approximation of betweenness centrality through sampling. In: Proceedings of the 7th ACM International Conference on Web Search and Data Mining, pp. 413–422. ACM (2014)

69. Chehreghani, M. H.: An efficient algorithm for approximate betweenness centrality computation. Comput. J. bxu003 (2014)
70. Agarwal, M., Singh, R.R., Chaudhary, S., Iyengar, S.R.S.: An efficient estimation of a node's betweenness. In: Complex Networks VI, pp. 111–121. Springer (2015)
71. Singh, R.R., Iyengar, S.R.S., Chaudhary, S., Agarwal, M.: An efficient heuristic for betweenness estimation and ordering. Soc. Netw. Anal. Min. **8**(1), 66 (2018)
72. Borassi, M., Natale, E.: Kadabra is an adaptive algorithm for betweenness via random approximation. J. Exp. Algorithmics (JEA) **24**(1), 1–35 (2019)
73. Chehreghani, M.H., Abdessalem, T., Bifet, A.: Metropolis-hastings algorithms for estimating betweenness centrality talel abdessalem. In: 22nd International Conference on Extending Database Technology EDBT 2019. Lisbon, Portugal (2019)
74. Furno, A., El Faouzi, N.E., Sharma, R., Zimeo, E.: Two-level clustering fast betweenness centrality computation for requirement-driven approximation. In: 2017 IEEE International Conference on Big Data (Big Data), pp. 1289–1294. IEEE (2017)
75. Furno, A., El Faouzi, N.E., Sharma, R., Zimeo, E.: Fast approximated betweenness centrality of directed and weighted graphs. In: International Conference on Complex Networks and their Applications, pp. 52–65. Springer (2018)
76. Haghir Chehreghani, M., Bifet, A., Abdessalem, T.: Adaptive algorithms for estimating betweenness and k-path centralities. In: Proceedings of the 28th ACM International Conference on Information and Knowledge Management, pp. 1231–1240 (2019)
77. Ostrowski, D.A.: An approximation of betweenness centrality for social networks. In: Proceedings of the 2015 IEEE 9th International Conference on Semantic Computing (IEEE ICSC 2015), pp. 489–492. IEEE (2015)
78. Matta, J., Ercal, G., Sinha, K.: Comparing the speed and accuracy of approaches to betweenness centrality approximation. Comput. Soc. Netw. **6**(1), 2 (2019)
79. Wink, A.M., de Munck, J.C., van der Werf, Y.D., van den Heuvel, O.A., Barkhof, F.: Fast eigenvector centrality mapping of voxel-wise connectivity in functional magnetic resonance imaging: implementation, validation, and interpretation. Brain Connect. **2**(5), 265–274 (2012)
80. Kumar, A., Mehrotra, K. G., Mohan, C. K.: Neural networks for fast estimation of social network centrality measures. In: Proceedings of the Fifth International Conference on Fuzzy and Neuro Computing (FANCCO-2015), pp. 175–184. Springer (2015)
81. Charalambous, T., Hadjicostis, C.N., Rabbat, M.G., Johansson, M.: Totally asynchronous distributed estimation of eigenvector centrality in digraphs with application to the pagerank problem. In: 2016 IEEE 55th Conference on Decision and Control (CDC), pp. 25–30. IEEE (2016)
82. Ruggeri, N., De Bacco, C.: Sampling on networks: estimating eigenvector centrality on incomplete networks. In: International Conference on Complex Networks and Their Applications, pp. 90–101. Springer (2019)
83. Mitliagkas, I., Borokhovich, M., Dimakis, A.G., Caramanis, C.: Frogwild! fast pagerank approximations on graph engines. Proc. VLDB Endow. **8**(8), 874–885 (2015). April
84. Yan, Y., Xiao, L., Xintian, Z.: Analyzing and identifying of cascading failure in supply chain networks. In: 2010 International Conference on Logistics Systems and Intelligent Management, vol. 3, pp. 1292–1295 (2010)
85. Kas, M., Carley, K.M., Carley, L.R.: Incremental closeness centrality for dynamically changing social networks. In: Proceedings of the 2013 IEEE/ACM International Conference on Advances in Social Networks Analysis and Mining, pp. 1250–1258 (2013)
86. Sariyüce, A.E., Kaya, K., Saule, E., Çatalyiirek, Ü.V.: Incremental algorithms for closeness centrality. In: 2013 IEEE International Conference on Big Data, pp. 487–492. IEEE (2013)
87. Yen, C.C., Yeh, M.Y., Chen, M.S.: An efficient approach to updating closeness centrality and average path length in dynamic networks. In: 2013 IEEE 13th International Conference on Data Mining, pp. 867–876. IEEE (2013)
88. Wei, W., Carley, K.: Real time closeness and betweenness centrality calculations on streaming network data (2014)

89. Khopkar, S.S., Nagi, R., Nikolaev, A.G., Bhembre, V.: Efficient algorithms for incremental all pairs shortest paths, closeness and betweenness in social network analysis. Soc. Netw. Anal. Min. **4**(1), 1–20 (2014)
90. Sarıyüce, A.E., Saule, E., Kaya, K., Çatalyürek, Ü.V.: Incremental closeness centrality in distributed memory. Parallel Comput. **47**, 3–18 (2015)
91. Santos, E.E., Korah, J., Murugappan, V., Subramanian, S.: Efficient anytime anywhere algorithms for closeness centrality in large and dynamic graphs. In: 2016 IEEE International Parallel and Distributed Processing Symposium Workshops (IPDPSW), pp. 1821–1830. IEEE (2016)
92. Ni, P., Hanai, M., Tan, W.J., Cai, W.: Efficient closeness centrality computation in time-evolving graphs. In: Proceedings of the 2019 IEEE/ACM International Conference on Advances in Social Networks Analysis and Mining, pp. 378–385 (2019)
93. Shao, Z., Guo, N., Gu, Y., Wang, Z., Li, F., Yu, G.: Efficient closeness centrality computation for dynamic graphs. In: International Conference on Database Systems for Advanced Applications, pp. 534–550. Springer (2020)
94. Vignesh, B., Ramachandran, S., Iyengar, D., Rangan, D.C.P., et al.: A lookahead algorithm to compute betweenness centrality. arXiv:1108.3286 (2011)
95. Lee, G.S., Djauhari, M.A.: An overall centrality measure: the case of us stock market. Int. J. Electr. Comput. Sci. **12**(6), (2012)
96. Green, O., McColl, R., Bader, D.A.: A fast algorithm for streaming betweenness centrality. In: Privacy, Security, Risk and Trust (PASSAT), 2012 International Conference on and 2012 International Confernece on Social Computing (SocialCom), pp. 11–20 (2012)
97. Kas, M., Wachs, M., Carley, K.M., Carley, L.R.: Incremental algorithm for updating betweenness centrality in dynamically growing networks. In: Proceedings of the 2013 IEEE/ACM International Conference on Advances in Social Networks Analysis and Mining, ASONAM '13, pp. 33–40, New York, USA. ACM (2013)
98. Ramalingam, G., Reps, T.: On the Computational Complexity of Incremental Algorithms. University of Wisconsin-Madison, Computer Sciences Department (1991)
99. Kas, M., Carley, K.M., Carley, L.R.: An incremental algorithm for updating betweenness centrality and k-betweenness centrality and its performance on realistic dynamic social network data. Soc. Netw. Anal. Min. **4**(1), 1–23 (2014)
100. Nasre, M., Pontecorvi, M., Ramachandran, V.: Betweenness centrality–incremental and faster. In: Mathematical Foundations of Computer Science 2014, pp. 577–588. Springer (2014)
101. Nasre, M., Pontecorvi, M., Ramachandran, V.: Decremental all-pairs all shortest paths and betweenness centrality. In: Algorithms and Computation, pp. 766–778. Springer (2014)
102. Demetrescu, C., Italiano, G.F.: A new approach to dynamic all pairs shortest paths. J. ACM (JACM) **51**(6), 968–992 (2004)
103. Kourtellis, N., Morales, G.D.F., Bonchi, F.: Scalable online betweenness centrality in evolving graphs. arXiv:1401.6981 (2014)
104. Pontecorvi, M., Ramachandran, V.: Fully dynamic all pairs all shortest paths. arXiv:1412.3852 (2014)
105. Pontecorvi, M., Ramachandran, V.: A faster algorithm for fully dynamic betweenness centrality. arXiv:1506.05783 (2015)
106. Pontecorvi, M., Ramachandran, V.: Fully dynamic betweenness centrality. In: Algorithms and Computation, pp. 331–342. Springer (2015)
107. Singh, R.R., Goel, K., Iyengar, S.R.S., Gupta, S., et al.: A faster algorithm to update betweenness centrality after node alteration. In: Algorithms and Models for the Web Graph, pp. 170–184. Springer (2013)
108. Singh, R.R., Goel, K., Iyengar, S.R.S., Gupta, S.: A faster algorithm to update betweenness centrality after node alteration. Internet Math. **11**(4–5), 403–420 (2015)
109. Hayashi, T., Akiba, T., Yoshida, Y.: Fully dynamic betweenness centrality maintenance on massive networks. Proc. VLDB Endow. **9**(2), 48–59 (2015)
110. Bergamini, E., Meyerhenke, H., Ortmann, M., Slobbe, A.: Faster betweenness centrality updates in evolving networks. In: 16th International Symposium on Experimental Algorithms (SEA 2017). Schloss Dagstuhl-Leibniz-Zentrum fuer Informatik (2017)

111. Tsalouchidou, I., Baeza-Yates, R., Bonchi, F., Liao, K., Sellis, T.: Temporal betweenness centrality in dynamic graphs. Int. J. Data Sci. Anal. 1–16 (2019)
112. Bahmani, B., Chowdhury, A., Goel, A.: Fast incremental and personalized pagerank. arXiv:1006.2880 (2010)
113. Rossi, R.A., Gleich, D.F.: Dynamic pagerank using evolving teleportation. In: International Workshop on Algorithms and Models for the Web-Graph, pp. 126–137. Springer (2012)
114. Rozenshtein, P., Gionis, A.: Temporal pagerank. In: Joint European Conference on Machine Learning and Knowledge Discovery in Databases, pp. 674–689. Springer (2016)
115. Zhan, Z., Hu, R., Gao, X., Huai, N.: Fast incremental pagerank on dynamic networks. In: International Conference on Web Engineering, pp. 154–168. Springer (2019)
116. Nathan, E., Bader, D. A.: A dynamic algorithm for updating katz centrality in graphs. In: Proceedings of the 2017 IEEE/ACM International Conference on Advances in Social Networks Analysis and Mining 2017, pp. 149–154 (2017)
117. Sarmento, R.P., Cordeiro, M., Brazdil, P., Gama, J.: Efficient incremental laplace centrality algorithm for dynamic networks. In: International Conference on Complex Networks and their Applications, pp. 341–352. Springer (2017)
118. Holme, P., Saramäki, J.: Temporal networks. Phys. Rep. **519**(3), 97–125 (2012)
119. Masuda, N., Lambiotte, R.: A Guide to Temporal Networks, vol. 4. World Scientific (2016)
120. Wang, Y., Yuan, Y., Ma, Y., Wang, G.: Time-dependent graphs: definitions, applications, and algorithms. Data Sci. Eng. **4**(4), 352–366 (2019)
121. Kim, H., Anderson, R.: Temporal node centrality in complex networks. Phys. Rev. E **85**(2), 026107 (2012)
122. Pan, R.K., Saramäki, J.: Path lengths, correlations, and centrality in temporal networks. Phys. Rev. E **84**(1), 016105 (2011)
123. Taylor, D., Myers, S.A., Clauset, A., Porter, M.A., Mucha, P.J.: Eigenvector-based centrality measures for temporal networks. Multiscale Model. Simul. **15**(1), 537–574 (2017)
124. Rocha, L.E.C., Masuda, N.: Random walk centrality for temporal networks. New J. Phys. **16**(6), 063023 (2014)
125. Lv, L., Zhang, K., Zhang, T., Bardou, D., Zhang, J., Cai, Y.: Pagerank centrality for temporal networks. Phys. Lett. A **383**(12), 1215–1222 (2019)
126. Bergamini, E., Meyerhenke, H., Staudt, C.L.: Approximating betweenness centrality in large evolving networks. arXiv:1409.6241 (2014)
127. Bergamini, E., Meyerhenke, H.: Fully-dynamic approximation of betweenness centrality. arXiv:1504.07091 (2015)
128. Riondato, M., Upfal, E.: Abra: approximating betweenness centrality in static and dynamic graphs with rademacher averages. ACM Trans. Knowl. Discov. Data (TKDD) **12**(5), 1–38 (2018)
129. Chehreghani, M.H., Bifet, A., Abdessalem, T.: Dybed: an efficient algorithm for updating betweenness centrality in directed dynamic graphs. In: 2018 IEEE International Conference on Big Data (Big Data), pp. 2114–2123. IEEE (2018)
130. Zhang, H., Lofgren, P., Goel, A.: Approximate personalized pagerank on dynamic graphs. In: Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, pp. 1315–1324 (2016)
131. Nathan, E., Bader, D.A.: Approximating personalized katz centrality in dynamic graphs. In: International Conference on Parallel Processing and Applied Mathematics, pp. 290–302. Springer (2017)
132. Bader, D.A., Madduri, K., et al.: Parallel algorithms for evaluating centrality indices in real-world networks. In: International Conference on Parallel Processing, ICPP 2006, pp. 539–550. IEEE (2006)
133. García, J.F., Carriegos, M.V.: On parallel computation of centrality measures of graphs. J. Supercomput. **75**(3), 1410–1428 (2019)
134. Santos, E.E., Pan, L., Arendt, D., Pittkin, M.: An effective anytime anywhere parallel approach for centrality measurements in social network analysis. In: 2006 IEEE International Conference on Systems, Man and Cybernetics, vol. 6, pp. 4693–4698. IEEE (2006)

135. Shukla, K., Regunta, S.C., Tondomker, S.H., Kothapalli, K.: Efficient parallel algorithms for betweenness-and closeness-centrality in dynamic graphs. In: Proceedings of the 34th ACM International Conference on Supercomputing, pp. 1–12 (2020)
136. Wang, W., Tang, C.Y.: Distributed computation of classic and exponential closeness on tree graphs. In: 2014 American Control Conference, pp. 2090–2095. IEEE (2014)
137. Wang, W., Tang, C.Y.: Distributed estimation of closeness centrality. In: 2015 54th IEEE Conference on Decision and Control (CDC), pp. 4860–4865. IEEE (2015)
138. You, K., Tempo, R., Qiu, L.: Distributed algorithms for computation of centrality measures in complex networks. IEEE Trans. Autom. Control **62**(5), 2080–2094 (2016)
139. Bernaschi, M., Carbone, G., Vella, F.: Scalable betweenness centrality on multi-gpu systems. In: Proceedings of the ACM International Conference on Computing Frontiers, pp. 29–36 (2016)
140. Castiello, A., Fucci, G., Furno, A., Zimeo, E.: Scalability analysis of cluster-based betweenness computation in large weighted graphs. In: 2018 IEEE International Conference on Big Data (Big Data), pp. 4006–4015. IEEE (2018)
141. Edmonds, N., Hoefler, T., Lumsdaine, A.: A space-efficient parallel algorithm for computing betweenness centrality in distributed memory. In: 2010 International Conference on High Performance Computing (HiPC), pp. 1–10. IEEE (2010)
142. Madduri, K., Ediger, D., Jiang, K., Bader, D.A., Chavarria-Miranda, D., et al.: A faster parallel algorithm and efficient multithreaded implementations for evaluating betweenness centrality on massive datasets. In: IEEE International Symposium on Parallel & Distributed Processing, 2009. IPDPS 2009, pp. 1–8. IEEE (2009)
143. McLaughlin, A., Bader, D.A., et al.: Revisiting edge and node parallelism for dynamic gpu graph analytics. In: 2014 IEEE International Parallel & Distributed Processing Symposium Workshops (IPDPSW), pp. 1396–1406. IEEE (2014)
144. Prountzos, D., Pingali, K.: Betweenness centrality: algorithms and implementations. In: ACM SIGPLAN Notices, vol. 48, pp. 35–46. ACM (2013)
145. Solomonik, E., Besta, M., Vella, F., Hoefler, T.: Scaling betweenness centrality using communication-efficient sparse matrix multiplication. In: Proceedings of the International Conference for High Performance Computing, Networking, Storage and Analysis, pp. 1–14 (2017)
146. van der Grinten, A., Angriman, E., Meyerhenke, H.: Parallel adaptive sampling with almost no synchronization. In: European Conference on Parallel Processing, pp. 434–447. Springer (2019)
147. van der Grinten, A., Angriman, E., Meyerhenke, H.: Scaling up network centrality computations–a brief overview. IT-Inf. Technol. **62**(3–4), 189–204 (2020)
148. van der Grinten, A., Angriman, E., Meyerhenke, H.: Scaling betweenness approximation to billions of edges by mpi-based adaptive sampling. arXiv:1910.11039 (2019)
149. Vella, F., Bernaschi, M., Carbone, G.: Dynamic merging of frontiers for accelerating the evaluation of betweenness centrality. J. Exp. Algorithmics (JEA) **23**, 1–19 (2018)
150. Crescenzi, P., Fraigniaud, P., Paz, A.: Simple and fast distributed computation of betweenness centrality. arXiv:2001.08108 (2020)
151. Hoang, L., Pontecorvi, M., Dathathri, R., Gill, G., You, B., Pingali, K., Ramachandran, V.: A round-efficient distributed betweenness centrality algorithm. In: Proceedings of the 24th Symposium on Principles and Practice of Parallel Programming, pp. 272–286 (2019)
152. Wang, W., Tang, C.Y.: Distributed computation of node and edge betweenness on tree graphs. In: 52nd IEEE Conference on Decision and Control, pp. 43–48. IEEE (2013)
153. Wang, W., Tang, C.Y.: Distributed estimation of betweenness centrality. In *2015 53rd Annual Allerton Conference on Communication, Control, and Computing (Allerton)*, pages 250–257. IEEE, 2015
154. Bian, R., Koh, Y.S., Dobbie, G., Divoli, A.: Identifying top-k nodes in social networks: a survey. ACM Comput. Surv. (CSUR) **52**(1), 1–33 (2019)
155. Bisenius, P., Bergamin, E., Angriman, E., Meyerhenke, H.: Computing top-k closeness centrality in fully-dynamic graphs. In: 2018 Proceedings of the Twentieth Workshop on Algorithm Engineering and Experiments (ALENEX), pp. 21–35. SIAM (2018)

156. Okamoto, K., Chen, W., Li, X.Y.: Ranking of closeness centrality for large-scale social networks. In: International Workshop on Frontiers in Algorithmics, pp. 186–195. Springer (2008)
157. Mahyar, H., Hasheminezhad, R., Ghalebi, E., Nazemian, A., Grosu, R., Movaghar, A., Rabiee, H.R.: Compressive sensing of high betweenness centrality nodes in networks. Phys. Stat. Mech. Appl. **497**, 166–184 (2018)
158. Nakajima, K., Shudo, K.: Estimating high betweenness centrality nodes via random walk in social networks. J. Inf. Process. **28**, 436–444 (2020)
159. Zhan, J., Gurung, S., Parsa, S.P.K.: Identification of top-k nodes in large networks using katz centrality. J. Big Data **4**(1), 1–19 (2017)
160. Saxena, A., Gera, R., Iyengar, S.R.S.: Estimating degree rank in complex networks. Soc. Netw. Anal. Min. **8**(1), 42 (2018)
161. Saxena, A., Gera, R., Iyengar, S.R.S.: A heuristic approach to estimate nodes' closeness rank using the properties of real world networks. Soc. Netw. Anal. Min. **9**(1), 3 (2019)
162. Newman, M.E.J.: Scientific collaboration networks. ii. shortest paths, weighted networks, and centrality. Phys. Rev. E **64**(1), 016132 (2001)
163. Qi, X., Fuller, E., Qin, W., Yezhou, W., Zhang, C.-Q.: Laplacian centrality: a new centrality measure for weighted networks. Inf. Sci. **194**, 240–253 (2012)
164. Wei, D., Deng, X., Zhang, X., Deng, Y., Mahadevan, S.: Identifying influential nodes in weighted networks based on evidence theory. Phys. A **392**(10), 2564–2575 (2013)
165. Candeloro, L., Savini, L., Conte, A.: A new weighted degree centrality measure: The application in an animal disease epidemic. PLoS One **11**(11), e0165781 (2016)
166. Li, M., Wang, J., Wang, H., Pan, Y.: Essential proteins discovery from weighted protein interaction networks. In: International Symposium on Bioinformatics Research and Applications, pp. 89–100. Springer (2010)
167. Tang, X., Wang, J., Zhong, J., Pan, Y.: Predicting essential proteins based on weighted degree centrality. IEEE/ACM Trans. Comput. Biol. Bioinf. **11**(2), 407–418 (2013)
168. Abbasi, A.: h-type hybrid centrality measures for weighted networks. Scientometrics **96**(2), 633–640 (2013)
169. Abbasi, A., Hossain, L.: Hybrid centrality measures for binary and weighted networks. In: Complex Networks, pp. 1–7. Springer (2013)
170. Akanmu, A.A., Wang, F.Z., Yamoah, F.A.: Clique structure and node-weighted centrality measures to predict distribution centre location in the supply chain management. In: Science and Information Conference (SAI), 2014, pp. 100–111. IEEE (2014)
171. Singh, A., Singh, R.R., Iyengar, S.R.S.: Hybrid centrality measures for service coverage problem. In: International Conference on Computational Data and Social Networks, pp. 81–94. Springer (2019)
172. Singh, A., Singh, R.R., Iyengar, S.R.S.: Node-weighted centrality: a new way of centrality hybridization. Comput. Soc. Netw. **7**(1), 1–33 (2020)
173. Wiedermann, M., Donges, J.F., Heitzig, J., Kurths, J.: Node-weighted interacting network measures improve the representation of real-world complex systems. EPL (Eur. Lett.) **102**(2), 28007 (2013)
174. Ni, C., Sugimoto, C., Jiang, J.: Degree, closeness, and betweenness: application of group centrality measurements to explore macro-disciplinary evolution diachronically. In: Proceedings of ISSI, pp. 1–13 (2011)
175. Zhao, J., Lui, J.C., Towsley, D., Guan, X.: Measuring and maximizing group closeness centrality over disk-resident graphs. In: Proceedings of the 23rd International Conference on World Wide Web, pp. 689–694 (2014)
176. Chen, C., Wang, W., Wang, X.: Efficient maximum closeness centrality group identification. In: Australasian Database Conference, pp. 43–55. Springer (2016)
177. Dolev, S., Elovici, Y., Puzis, R., Zilberman, P.: Incremental deployment of network monitors based on group betweenness centrality. Inf. Process. Lett. **109**(20), 1172–1176 (2009)
178. Halappanavar, M., Chen, Y., Adolf, R., Haglin, D., Huang, Z., Rice, M.: Towards efficient nx contingency selection using group betweenness centrality. In: 2012 SC Companion: High Performance Computing, Networking Storage and Analysis, pp. 273–282. IEEE (2012)

179. Brandes, U.: On variants of shortest-path betweenness centrality and their generic computation. Soc. Netw. **30**(2), 136–145 (2008)
180. Kolaczyk, E.D., Chua, D.B., Barthélemy, M.: Group betweenness and co-betweenness: interrelated notions of coalition centrality. Soc. Netw. **31**(3), 190–203 (2009)
181. Puzis, R., Elovici, Y., Dolev, S.: Fast algorithm for successive computation of group betweenness centrality. Phys. Rev. E **76**(5), 056709 (2007)
182. Qiu, L.Q., Liang, Y.Q., Chen, Z.Y., Fan, J.C.: A new measurement for the importance of nodes in networks. Control Eng. Inf. Syst. 483–486 (2014)
183. Li-Qing, Q., Yong-Quan, L., Zhuo-Yan, C.: A novel algorithm for detecting local community structure based on hybrid centrality. J. Appl. Sci. **14**, 3532–3537 (2014)
184. Zhang, X.J., Wang, Z.L., Zhang, Z.X.: Finding most vital node in satellite communication network. In: Applied Mechanics and Materials, vol. 635, pp. 1136–1139. Trans Tech Publications Ltd. (2014)
185. Buechel, B., Buskens, V.: The dynamics of closeness and betweenness. J. Math. Sociol. **37**(3), 159–191 (2013)
186. Qiao, S., Peng, J., Li, H., Li, T., Liu, L., Li, H.: Webrank: a hybrid page scoring approach based on social network analysis. In: Rough Set and Knowledge Technology, pp. 475–482. Springer (2010)
187. Wang, J., Rong, L., Guo, T.: A new measure of node importance in complex networks with tunable parameters. In: 4th International Conference on Wireless Communications, Networking and Mobile Computing, 2008. WiCOM'08, pp. 1–4. IEEE (2008)
188. Avrachenkov, K., Litvak, N.: The effect of new links on google pagerank. Stoch. Model. **22**(2), 319–331 (2006)
189. de Kerchove, C., Ninove, L., Van Dooren, P.: Maximizing pagerank via outlinks. Linear Algebra Appl. **429**(5–6), 1254–1276 (2008)
190. Olsen, M.: Maximizing pagerank with new backlinks. In: International Conference on Algorithms and Complexity, pp. 37–48. Springer (2010)
191. Demaine, E.D., Zadimoghaddam, M.: Minimizing the diameter of a network using shortcut edges. In: Scandinavian Workshop on Algorithm Theory, pp. 420–431. Springer (2010)
192. Perumal, S., Basu, P., Guan, Z.: Minimizing eccentricity in composite networks via constrained edge additions. In: MILCOM 2013-2013 IEEE Military Communications Conference, pp. 1894–1899. IEEE (2013)
193. Crescenzi, P., d'Angelo, G., Severini, L., Velaj, Y.: Greedily improving our own closeness centrality in a network. ACM Trans. Knowl. Discov. Data (TKDD) **11**(1), 1–32 (2016)
194. Bergamini, E., Crescenzi, P., D'angelo, G., Meyerhenke, H., Severini, L., Velaj, Y.: Improving the betweenness centrality of a node by adding links. J. Exp. Algorithmics (JEA) **23**, 1–32 (2018)
195. D'Angelo, G., Severini, L., Velaj, Y.: On the maximum betweenness improvement problem. Electron. Notes Theor. Comput. Sci. **322**, 153–168 (2016)
196. Shan, L., Yi, Y., Zhang, Z.: Improving information centrality of a node in complex networks by adding edges. In: Proceedings of the 27th International Joint Conference on Artificial Intelligence, pp. 3535–3541 (2018)
197. D'Angelo, G., Olsen, M., Severini, L.: Coverage centrality maximization in undirected networks. Proc. AAAI Conf. Artif. Intell. **33**, 501–508 (2019)
198. Medya, S., Silva, A., Singh, A., Basu, P., Swami, A.: Group centrality maximization via network design. In: Proceedings of the 2018 SIAM International Conference on Data Mining, pp. 126–134. SIAM (2018)
199. Angriman, E., van der Grinten, A., Bojchevski, A., Zügner, D., Günnemann, S., Meyerhenke, H.: Group centrality maximization for large-scale graphs. In: 2020 Proceedings of the Twenty-Second Workshop on Algorithm Engineering and Experiments (ALENEX), pp. 56–69. SIAM (2020)
200. Girvan, M., Newman, M.E.J.: Community structure in social and biological networks. Proc. Natl. Acad. Sci. **99**(12), 7821–7826 (2002)

201. Yan, E., Ding, Y.: Applying centrality measures to impact analysis: a coauthorship network analysis. J. Am. Soc. Inform. Sci. Technol. **60**(10), 2107–2118 (2009)
202. Ghosh, R., Lerman, K.: Predicting influential users in online social networks. In: SNAKDD Proceedings of KDD Workshop on Social Network Analysis. Citeseer (2010)
203. Ilyas, M.U., Radha, H.: Identifying influential nodes in online social networks using principal component centrality. In: 2011 IEEE International Conference on Communications (ICC), pp. 1–5. IEEE (2011)
204. Mehrotra, A., Sarreddy, M., Singh, S.: Detection of fake twitter followers using graph centrality measures. In: 2016 2nd International Conference on Contemporary Computing and Informatics (IC3I), pp. 499–504. IEEE (2016)
205. Riquelme, F., González-Cantergiani, P.: Measuring user influence on twitter: a survey. Inf. Process. Manag. **52**(5), 949–975 (2016)
206. Lohmann, G., Margulies, D.S., Horstmann, A., Pleger, B., Lepsien, J., Goldhahn, D., Schloegl, H., Stumvoll, M., Villringer, A., Turner, R.: Eigenvector centrality mapping for analyzing connectivity patterns in fmri data of the human brain. PloS One **5**(4), e10232 (2010)
207. Zinoviev, D.: A social network of russian "kompromat". arXiv:2009.08631 (2020)
208. Kim, J., Jeong, D., Choi, D., Park, E.: Exploring public perceptions of renewable energy: evidence from a word network model in social network services. Energ. Strat. Rev. **32**, 100552 (2020)
209. Nurrokhman, N., Purnomo, H.D., Hartomo, K.D.: Utilization of social network analysis (SNA) in knowledge sharing in college. INTENSIF: Jurnal Ilmiah Penelitian dan Penerapan Teknologi Sistem Informasi **4**(2), 259–271 (2020)
210. Stelzhammer, P.: Efficient detection of influential users in social recommender systems. Ph.D. thesis, Wien (2020)
211. Trach, R., Bushuyev, S.: Analysis communication network of construction project participants. Przegląd Naukowy Inżynieria i Kształtowanie Środowiska, 29 (2020)
212. Yuan, F.-C.: Intelligent forecasting of inbound tourist arrivals by social networking analysis. Phys. A **558**, 124944 (2020)
213. Neuberger, J.: Centrality and centralisation a social network analysis of the early soviet film industry, 1918–1953. Apparatus. Film, Media Digit. Cult. Cent. East. Eur. (10), (2020)
214. Nagdive, A.S., Tugnayat, R., Peshkar, A.: Social network analysis of terrorist networks. Int. J. Eng. Adv. Technol. **9**(3), 2553–2559 (2020)

Chapter 2
Network Centrality Measures: Role and Importance in Social Networks

Rahul Saxena and Mahipal Jadeja

Abstract Social Networks, in the twentieth century, have emerged as the greatest source of mass communication and possibly the best means of information propagation whether it is through Facebook, Twitter, Instagram, WhatsApp, or any other social platform. With the emerging use of ICT and digital globalization, social connections have increased at a rapid rate and are evolving faster with time. This calls for the need for the identification of entities that hold high importance in the network to strategize for information flow in the network. Network centrality or identifying central nodes in the network is a kind of study of this aspect only. There exist nodes of high value based on parameters like high reachability, high accessibility, closeness, etc. which makes the network traffic induced more toward them. Identification of such nodes helps in decision making for propagating information in short steps or in less number of communications in the network, avoiding information traverse from certain paths or curbing information flow, etc. For example, news published in New York Times will certainly be wildfire fast in comparison to when the news is being circulated in social ties of a community only. In this chapter, we come up with an interesting exploration of the centrality measures concept and theory for a network. The chapter focuses on how different centralities play a crucial role in determining the guiding nodes of any action in the network. Further, some existing prevailing works in the literature will be discussed indicating about how the information on the web can help us in identifying the nodes, set of nodes, or networks that are of prime importance based upon which the information flow in the network takes its shape. Experimental simulations over *SNAP* (*Stanford Network Analysis Platform*) are conducted to understand the concepts in a more applicative manner. The chapter serves as a basic exploration of the network centrality measures, their applications,

R. Saxena (✉) · M. Jadeja
Malaviya National Institute of Technology, Jaipur, India
e-mail: 2019rcp9153@mnit.ac.in

M. Jadeja
e-mail: mahipaljadeja.cse@mnit.ac.in

R. Saxena
Manipal University Jaipur, Jaipur, India

A. Biswas et al. (eds.), *Principles of Social Networking*,
Smart Innovation, Systems and Technologies 246,
https://doi.org/10.1007/978-981-16-3398-0_2

and viewpoints to further investigate the measures over more real-world networks to find out more interesting results.

2.1 Introduction

In this section, we will discuss about what a social network is, what are its components, and how a social network can be visualized as a graphical network. After discussing about these basic preliminaries, we will focus on the centrality analysis and its measures to understand how they can play a crucial role in understanding the evolution of social web graphs.

2.1.1 What is Social Networking?

Before the late 90s, people heavily relied on the information, mainly news-related content on newspapers, television, radios, etc. With the inception of social networking websites in the early 2000s, the paradigm shifted from hand-delivered news material in the form of newspapers, telegrams, etc. to electronic mode [1]. By 2005–06, Facebook and Twitter had entered the arena of the social networking web and are still highly popular social interacting platforms. Other sites like Tumblr, Spotify, Foursquare, and Pinterest tried to fill up specific social networking niches. Since the last decade, social networking websites have captivated the communication needs so well that now it has become an integral part of almost every human being's daily routine life. The impact has been so much powerful that there has been a drastic increase in the number of online apps, news apps, and channels. The news channels have expanded their horizon running from national television broadcasters to live updates on social sites and their own apps [2]. This growth in people's involvement over social networking websites has not constrained to this but has seen a rapid growth in the e-commerce trading [3] as well. The recent emergence of Amazon, Flipkart, and other E-commerce websites has seen a high jump in the digital and retail market. Similarly, YouTube, Netflix, Amazon Prime, etc. (offering various channels and web series) have created a whole new world of viewers. In simple words, social networking has made the world connected remaining at their places, still being able to do the most of what they can.

This discussion brings to the conclusion to define Social Networks as [4]*:* "*Social networking is the use of Internet-based social media sites to stay connected with friends, family, colleagues, customers, or clients. Social networking can have a social purpose, a business purpose, or both, through sites such as Facebook, Twitter, LinkedIn, and Instagram, among others. Social networking has become a significant base for marketers seeking to engage customers.*" Following this, we will now discuss some interesting insights and analyses over the social web graphs.

2.1.2 Social Networks as Graph

Social Networks analysis is stated as the study of investigating social structures, behaviors, and interactions between individuals using network analysis and graph theory. To perform analysis over edge links and nodes of the graph are used to model this interaction and relationship [5]. Each individual in the graph represents a *node* and the connecting edge between these individuals is drawn based upon some notion like *relationship as a friend, similar likes or recommendations,* etc. This connecting tie is of very high importance as this forms the basis of analysis of the network. There is a wide literature available on how to define social ties between the entities in the graph [6–8]. The connecting edges stores the information database which enables us to associate nodes referred to as generating graph embeddings [9].

2.1.3 Why Centrality Analysis?

Given a connected graph simulating a real scenario, it can be visualized that few node positions are more central while the rest are peripheral, in context to the view of the graph at that instance. These key positions help in identifying the nodes of prime importance and deciding upon their roles in the network. This notion was first discussed by Roethlisberger et al. [10]. This information can be of great significance in determining the flow of information, forming various strategies, and many more so that information can be channelized in the network as soon as possible and in a more appropriate manner. Consider a small example of a connectivity network as shown in Fig. 2.1.

In this small network, if *node* 1 is chosen as a central node, then, it may not be a good choice to circulate information as it is more distant to more number of nodes in the network. On the contrary, *node 2* will be a good choice for the node to be central as it is closer (just an edge away) to all the nodes in the graph. However, the

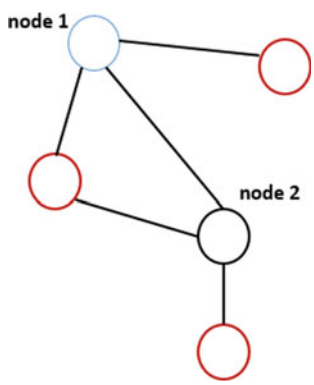


Fig. 2.1 Connectivity network

concept of centrality varies depending upon the objective and purpose. In the same manner, the applicability of the centrality measure may also vary. In the rest of the chapter, we discuss various centrality measures, their advantages, limitations, and their implication aspects.

2.2 Network Centrality: Measures and Concepts

Before delving into the details of network centrality measures and their types, let's first understand the categorization. The network centrality measures can be broadly classified into three classes:

- **Geometric Measures**
  In this class of measure, network centrality for a node is a *function of distances* to other nodes in the network. The importance of a node is determined based on how much a node is approachable to other nodes or how approachable a specific node is by other nodes.
- **Spectral Measures**
  In this class of centrality measure, a node's importance depends upon the eigen-structure of some graph-related matrix. In other words, a node is central depending upon its neighborhood nodes. Here the centrality is a function of the nodes associated with the node of interest.
- **Path-Based Measures**
  Here the centrality measure depends upon the fact that how often a node is visited between a defined source and destination. The concept originates from the idea of *edge betweenness* which gives the number of shortest paths passing over the edge.
  Apart from these, there exist other centrality measures but they may be considered as the variation to the basic versions. The deviation in the applicability procedure is problem and application specific. We will now put our focus on the centrality measures based upon these three categorizations.

2.2.1 Geometric Measures

- **Degree Centrality**

It is the simplest and historically first centrality measure that accounts for the count of the number of ties. It simply indicates the size of an individual's network. For a directed graph network, this centrality measure may have in-degree and outdegree centrality defined separately. For example, in the case of a web page navigation network, where nodes in the network represent the web pages. Incoming edge may be defined as the number of web pages referring to a particular page (say x). The

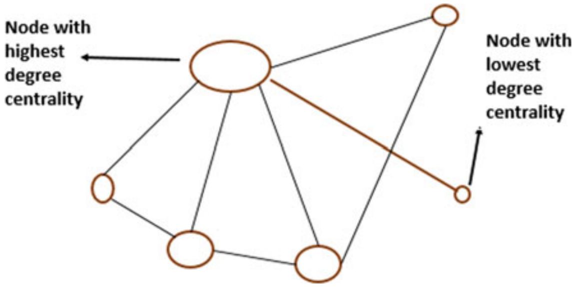


Fig. 2.2 Graph Instance representing degree centralities; (i) Node having a large number of connections has high centrality measure and importance. (ii) Node with the smallest size has only one connection and hence has the lowest centrality score. (iii) Rest all nodes have the same centrality score owing to the same number of connections

number of outgoing edges from the web page '*x*' stands for all the web pages which are being referred to by '*x*'. Depending upon the situation or need, the centrality measure aspect is taken into consideration. Functionally, it can be defined as per the following equation (see Fig. 2.2 for an example).

$$c_{deg}\,(x) = d_{in}\,(x)$$

Liu et al. [11] have taken into account these centrality measures to study the effect of networked criterion-based community engagement on their performance. The in-degree centrality measure analysis in the study accounted for the popularity or measure of how much popularity index a student has in the network. Similarly, the outdegree centrality measure defines how actively a student links to other students in the network. Ergun et al. [12] used the concept of degree centrality to study the effect of social networking structure formed in an Online Learning Environment. Similarly, there are other implications of this centrality measure-based result mentioned in the reported literature from [13–15].

- **Closeness Centrality**

Alexander Bavelas (December 26, 1913 [16]–August 16, 1993) was an American psych sociologist credited as the first to define closeness centrality. Degree centrality only takes into account the connections and weight each link equally important. However, that may not be true for many real-world networks. For example, in a road traffic network, nodes which have high connectivity to many nodes may not be as equally important to the nodes which have reachability to the nodes in the least time. In these situations, nodes that are more central and have smaller distances from other nodes in the network are considered to have high significance. Based on this notion, the functional definition can be given as

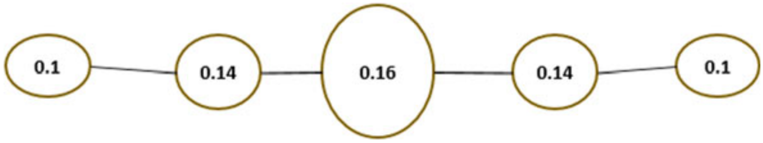


Fig. 2.3 Closeness centrality scores

$$c_{close}(\boldsymbol{x}) = \frac{\mathbf{1}}{\sum_y d(\boldsymbol{y}, \boldsymbol{x})}$$

Here $d(y, x)$ represents the shortest path from node$y$ to x. Let us consider a case as shown in Fig. 2.3.

Here the closeness centrality for the first node is calculated as

$$c_{close}(1) = \frac{1}{1+2+3+4} = 0.1$$

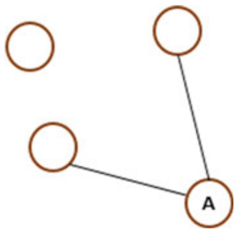
Similarly, for other nodes, the closeness centrality measures are calculated. Clearly, for the middle node, the centrality score will be highest as it has reachability to any node in the network in maximum 2 steps or can reach any node with maximum path length ($c_{close}(3) = {}^{1}\!/_{(2+1+1+2)} = 0.16$). The notion here is how much a vertex can communicate with other nodes without the help of in-between nodes to propagate the message. However, the problem that persists with this centrality measure is if the graph is disconnected, then this centrality measure fails. For example, in Fig. 2.4 shown, the centrality score calculation for any node will be undefined as the distance of any node 'x' with a disconnected 'y' will be defined as ∞.

The closeness centrality measure for node A will be

$$c_{close}\ (A) = \frac{1}{1+1+\ \infty} = 0$$

To counter this, the measure was remodeled by replacing the average distance with the harmonic mean of all the distances.

Fig. 2.4 Disconnected graph



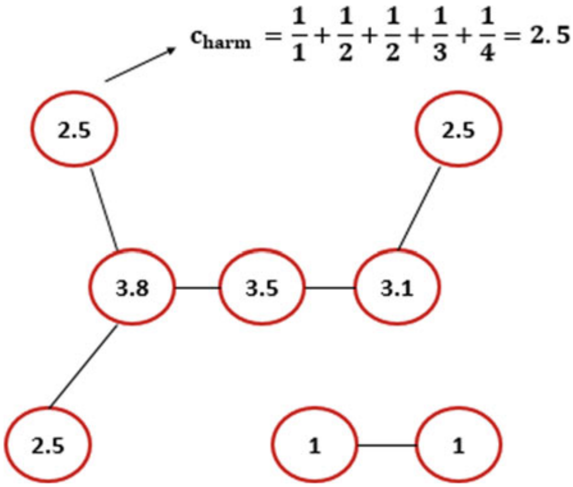


Fig. 2.5 Harmonic centrality scores

$$c_{harm}(x) = \sum_{d(y,x) < \infty,\quad y \neq x} \frac{1}{d(y, x)}$$

This modification helps in addressing the anomaly caused due to non-connected nodes and thus can be applied to graphs that are not strongly connected (Fig. 2.5).

Kas et al. [17] have proposed an incremental closeness centrality algorithm for dynamic social networks which has continuous addition and removal of edges and nodes. Mateusz et al. [18] used this centrality measure to identify the bus stops common to the several bus lines using the idea of Overlapping Community Structure. Likewise, there are various implications of this centrality measure [19–21].

Geometric measures discussed so far account for the node's importance based on the node's position in the network. In the next section, the discussion is focused upon how the centrality score of a node depends on the neighborhood nodes and how the centrality scores of the neighbor nodes too get influenced by central nodes.

2.2.2 *Spectral Measures*

The basic intuition of this class of centrality measure is that the nodes in contact with the central nodes have high centrality scores and those far away from these central nodes are considered to be low significance nodes.

- **Eigenvector Centrality**

Unlike degree centrality, the score calculation is done based on the fact that to which kind of nodes, the node 'x' is connected. It is better to be connected with a few popular (well connected) nodes than being connected to many nodes of low importance [22].

This measure of influence of a node proposed by *Phillip Bonacich*, in his 1986 paper *Power and Centrality: A Family of Measures* [23].

$$\boldsymbol{c_{eig}}\ (\boldsymbol{x}) = \frac{1}{\boldsymbol{\lambda}} \sum_{\boldsymbol{y \to x}} \boldsymbol{c_{eig}}\ (\boldsymbol{y})$$

where λ is defined as normalization constant $= \left\| c_{eig} \right\|_2$.

Here c_{eig} converges to dominant eigenvector of adjacency matrix, λ converges to the dominant eigenvalue of adjacency matrix A. Initially, each node is assigned a centrality score of 1. Then, in each successive iteration, the score gets revised as per the formula mentioned above. The matrix formulation of the same can be given as

$$AX = \lambda X$$

To understand it more clearly, let us consider an illustration for the graph shown below.

Matrix A for this graph will be defined as $A = \begin{bmatrix} 0 & 1 & 1 & 0 & 0 \\ 1 & 0 & 1 & 1 & 1 \\ 1 & 1 & 0 & 0 & 1 \\ 0 & 1 & 0 & 0 & 0 \\ 0 & 1 & 1 & 0 & 0 \end{bmatrix}$ and initial centrality score, $c = \begin{bmatrix} 1 \\ 1 \\ 1 \\ 1 \\ 1 \end{bmatrix}$. So for the first iteration, centrality scores will be evaluated as

$$\boldsymbol{\mathit{Iteration}\ 1}:\quad \mathbf{A.c}^{(0)} = \begin{bmatrix} 1 & 0 & 1 & 1 & 1 \\ 1 & 1 & 0 & 0 & 1 \\ 0 & 1 & 0 & 0 & 0 \\ 0 & 1 & 1 & 0 & 0 \end{bmatrix} \begin{bmatrix} 1 \\ 1 \\ 1 \\ 1 \end{bmatrix} = \begin{bmatrix} 4 \\ 3 \\ 1 \\ 2 \end{bmatrix} \underset{\equiv}{\boldsymbol{\mathit{def}}}\ c^{(1)}$$

And, finally defining the normalized scores as

$$\mathrm{c}^{(1)} / \left\| \mathrm{c}^{(1)} \right\|_2 = \begin{bmatrix} \frac{2}{\sqrt{(2^2+4^2+3^2+1^2+2^2)}} \\ \frac{4}{\sqrt{(2^2+4^2+3^2+1^2+2^2)}} \\ \frac{3}{\sqrt{(2^2+4^2+3^2+1^2+2^2)}} \\ \frac{1}{\sqrt{(2^2+4^2+3^2+1^2+2^2)}} \\ \frac{2}{\sqrt{(2^2+4^2+3^2+1^2+2^2)}} \end{bmatrix} = \begin{bmatrix} 0.34 \\ 0.68 \\ 0.51 \\ 0.17 \\ 0.34 \end{bmatrix}$$

$$\textbf{\textit{Iteration 2}}:\quad A.\,c^{(1)} = \begin{bmatrix} 0\ 1\ 1\ 0\ 0 \\ 1\ 0\ 1\ 1\ 1 \\ 1\ 1\ 0\ 0\ 1 \\ 0\ 1\ 0\ 0\ 0 \\ 0\ 1\ 1\ 0\ 0 \end{bmatrix} \begin{bmatrix} 0.34 \\ 0.68 \\ 0.51 \\ 0.17 \\ 0.34 \end{bmatrix} = \begin{bmatrix} 1.19 \\ 1.36 \\ 1.36 \\ 0.68 \\ 1.19 \end{bmatrix} \underline{\underline{\underline{\mathrm{def}}}} \begin{bmatrix} 0.45 \\ 0.51 \\ 1.36 \\ 0.68 \\ 1.19 \end{bmatrix}$$

Progressing in this manner, the final convergence for the centrality scores attained for the example is $c = \begin{bmatrix} 1 \\ 1.41 \\ 1.27 \\ 0.52 \\ 1 \end{bmatrix}$

Carreras et al. [24] used this centrality measure to analyze the spread of the epidemic in a highly decentralized mobile network. Baldesi et al. [25] used this centrality measure to have a cooperative distribution of streamlined content efficiently. Determining the centrality scores help in having the idea of the topology of the network. Like this, there are a number of related articles which discuss the use of this centrality measure. However, this centrality measure has its limitations. Eigenvector centrality will only work for connected and undirected graphs. To counter these, the Katz centrality index was proposed by making a slight modification to the centrality calculation measure discussed.

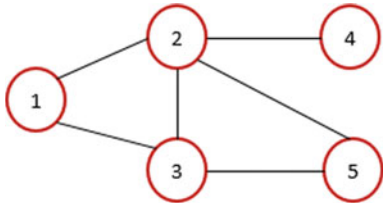
- **Katz's Centrality**

This centrality measure proposed by Leo Katz [26] defines a node's importance by taking into account the total number of walks between a pair of nodes, defined as

$$c_{katz}\,(x) = \beta \sum_{k=0}^{\infty} \sum_{x \to y} \alpha^k \left(A^k\right)_{xy}$$

where α is defined as the attenuation factor ranging from $\left(0, \frac{1}{\lambda}\right)$, λ being the largest eigenvalue of A. The attenuation factor penalizes the connection made with distant neighbors by factor *k*. A^k represents the path between nodes x and y with length *k*. β is to assign some importance to some particular nodes. Ideally, its value is kept one if none of the nodes in the network is to be assigned some special privilege. For the graph as per Fig. 2.6, the matrix A^k can be defined as

Fig. 2.6 Connected graph



$$A^1 = \begin{bmatrix} 0\ 1\ 1\ 0\ 0 \\ 1\ 0\ 1\ 1\ 1 \\ 1\ 1\ 0\ 0\ 1 \\ 0\ 1\ 0\ 0\ 0 \\ 0\ 1\ 1\ 0\ 0 \end{bmatrix}, \quad A^2 = \begin{bmatrix} 0\ 1\ 1\ 0\ 0 \\ 1\ 0\ 1\ 1\ 1 \\ 1\ 1\ 0\ 0\ 1 \\ 0\ 1\ 0\ 0\ 0 \\ 0\ 1\ 1\ 0\ 0 \end{bmatrix}^2 = \begin{bmatrix} 2\ 1\ 1\ 1\ 2 \\ 1\ 4\ 2\ 0\ 1 \\ 1\ 2\ 3\ 1\ 1 \\ 1\ 0\ 1\ 1\ 1 \\ 2\ 1\ 1\ 1\ 2 \end{bmatrix}$$

$$A^3 = \begin{bmatrix} 0\ 1\ 1\ 0\ 0 \\ 1\ 0\ 1\ 1\ 1 \\ 1\ 1\ 0\ 0\ 1 \\ 0\ 1\ 0\ 0\ 0 \\ 0\ 1\ 1\ 0\ 0 \end{bmatrix}^3 = \begin{bmatrix} 2\ 6\ 5\ 1\ 2 \\ 6\ 4\ 6\ 4\ 6 \\ 5\ 6\ 4\ 2\ 5 \\ 1\ 4\ 2\ 0\ 1 \\ 2\ 6\ 5\ 1\ 2 \end{bmatrix}$$

The entry in A^3 matrix in second row fifth column indicates there exists 6 paths of length 3 between vertices 2 and 5 [(2,1,3,5), (2,4,2,5), (2,3,2,5), (2,1,2,5), (2,5,3,5), (2,5,2,5)]**.** So, redefining Katz centrality as

$$c_{katz}(x) = \alpha \sum_{y \to x} (c_{katz}(y) + \beta)$$

This measure looks suitable for *directed acyclic graphs*. Since β is to assign a prioritized weightage to the nodes in the graph and is kept constant initially for a graph, it is α over which the centrality score of the node depends:

- For $\alpha \approx 0$, paths with length > 1 have low contribution and are less influential.
- For a large value of α, Katz scores are more influenced by topology and long paths are penalized gently.
- Measure diverges at $\alpha > \frac{1}{\lambda}$ and hence is the limit.

For the graph shown in Fig. 2.7, the initial centrality scores for the nodes are calculated for $\alpha = 0.85$ and $\beta = 1$ (for all nodes). For high α value, we have more paths greater than length 1 ending at node U than V. Changing the value of $\alpha = 0.15$ will revise the scores making node V's importance score closer to node U as longer paths will be penalized and shorter paths will be more important. Further, it can also be observed that increasing the β value for node B to 2 will make the centrality scores of node A, U, and all the nodes in contact with node B to rise [27].

Zhao et al. [28] used this centrality measure to rank the candidate disease gene and protein–protein interaction to predict the disease occurrence. Zhang et al. [29]

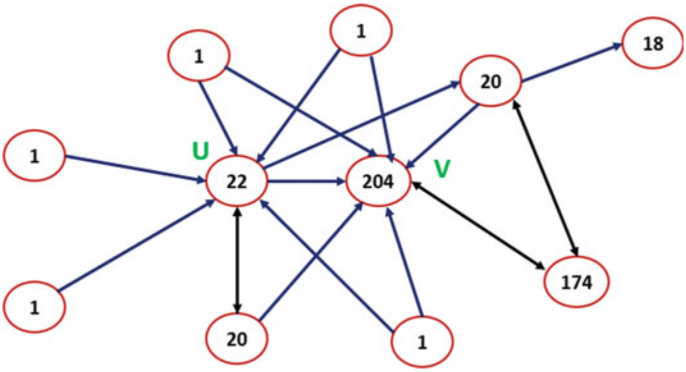


Fig. 2.7 Instance graph with Katz index for each node

use Katz's centrality measure to identify important nodes in a graph where each path has a different weightage. The results were found to have close coherence with the local path index. Similarly, there has been a lot of interesting research articles which have utilized Katz's centrality measure to identify nodes of importance and interest in a network. Landherr et al. [30] have given a comprehensive survey over the usage of various centrality measures and algorithm.

- **Page Rank and HITs Centrality Measure**

PageRank algorithm developed by Larry Page and *Sergey Brin* in *1996* at *Stanford University* is still used by Google to rank web pages. PageRank algorithm assign scores to the nodes in its simplest as

$$r_j^{(t+1)} = \sum_{i \to j} \frac{r_i^{(t)}}{d_i} \tag{2a}$$

where r_j is the score for the node at time $t + 1$ and r_i is the importance contribution of node i to node j normalized by its outdegree $d_{i.}$ Normalization is done due to the fact that the same node i also makes a contribution to other nodes as well. The process assigns each node with an initial score *(say 1)* and the scores are updated for each node in every iteration till the time scores for the nodes do not converge, where the convergence criteria is given by

$$\sum_i \left| r_i^{(t+1)} - r^i \right| < \epsilon$$

Based on this, algorithmic steps can be defined as

- Set $r_j = \frac{1}{N'}$ where N are the total number of nodes in the graph.
- 1: $r_j' = \sum_{i \to j} j \frac{r_i}{d_i}$

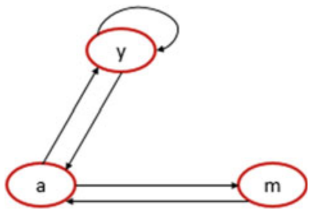


Fig. 2.8 Graph Instance for PageRank algorithm

- 2: $\boldsymbol{r} \leftarrow \boldsymbol{r}'$
- If $|r - r'| > \epsilon$: ***goto*** 1.

Tracing the above algorithm over an example as shown in Fig. 2.8.
Score calculation equations over this graph can be defined as

$$r_y = {}^{r_y}\!/_2 + {}^{r_a}\!/_2 \tag{2.1}$$

$$r_a = {}^{r_y}\!/_2 + r_m \tag{2.2}$$

$$r_m = {}^{r_a}\!/_2 \tag{2.3}$$

Based on these flow equations, the algorithm can be run to get the final PageRank scores of the nodes as

$$\begin{bmatrix} r_y \\ r_a \\ r_m \end{bmatrix} = \begin{matrix} 1/3 & 1/3 & 5/12 & 9/24 & & 6/15 \\ 1/3 & 3/6 & 1/3 & 11/24 & \cdots & 6/15 \\ 1/3 & 1/6 & 3/12 & 1/6 & & 3/15 \\ \mathit{Iteration}\ 0 & \mathit{Iteration}\ 1 & \mathit{Iteration}\ 2 & \mathit{Iteration}\ 3 & & \mathit{Final\ Scores} \end{matrix}$$

Thus, we get the final scores for all the nodes once the algorithm converges. However, the algorithm may not converge under two conditions:

- The algorithm may get stuck up to *dead ends*, i.e., the flow equations get stuck up to the nodes having no out links. These pages cause the importance to leak out.
- Sometimes the flow equations stuck up, sending and receiving all the flow within a constrained group. This is known as the problem of *Spider traps*. These spider traps absorb all importance.

The solution to these problems was a slight modification to Eq. (2a) as per [31]

$$r_j = \sum_{i \to j} \beta \frac{r_i}{d_i} + (1 - \beta)\frac{1}{n}$$

where β being the probability of following a link randomly. Thus, (1 − β) is the probability of teleporting, i.e., jumping to a random page to get out of the stuck.

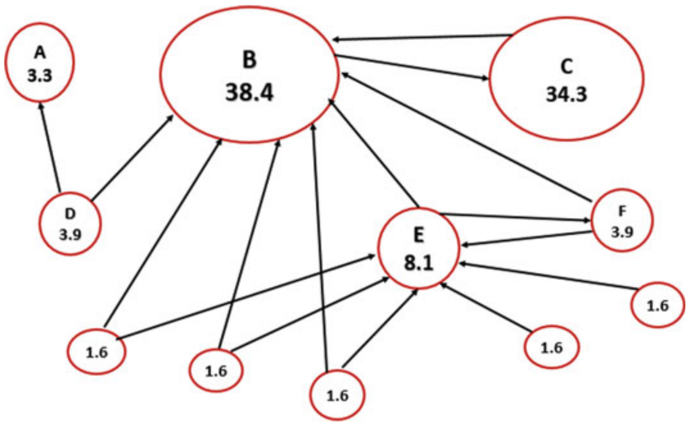


Fig. 2.9 Graph instance with PageRank scores of the nodes

Generally, the values of β range from 0.8 to 0.9. The above equation is equivalent to the dominant eigenvector:

$$r_j = \beta A_r + (1-\beta)1^T\frac{1}{n}$$

Here A_r represents graph adjacency matrix, in which rows are normalized to row sum one. Figure 2.9 shows an instance of a graph with PageRank scores inside the nodes.

Node *B* with more in links has a more importance contribution from a greater number of nodes in comparison to others. Thus, it has the highest PageRank score. In contrast, node C although has one in link but it is being referred to by a node of high importance in the network; hence, its popularity score also becomes high. With the same explanation, node E although have a number of in links making a contribution in imparting and enhancing its popularity score but it is being referred to by the nodes of low importance in the network.

The above discussion gives rise to the concept of *Hubs* and *Authorities* in a social network and *HITS* centrality algorithm. The basic ideology behind the concept follows from what we have discussed for the PageRank algorithm so far. The pages of interest hold their importance based upon the kind of links (in links or out links) the node exhibit and thus are categorized into two classes:

- ***Authorities*** are nodes containing useful information (like the homepage of newspapers, course homepages, Wikipedia Web page, etc.). They have high incoming links or visits.
- ***Hubs*** are nodes that link to authorities (like List of newspapers, Course bulletin, etc.). These nodes have high outgoing links or visits made.

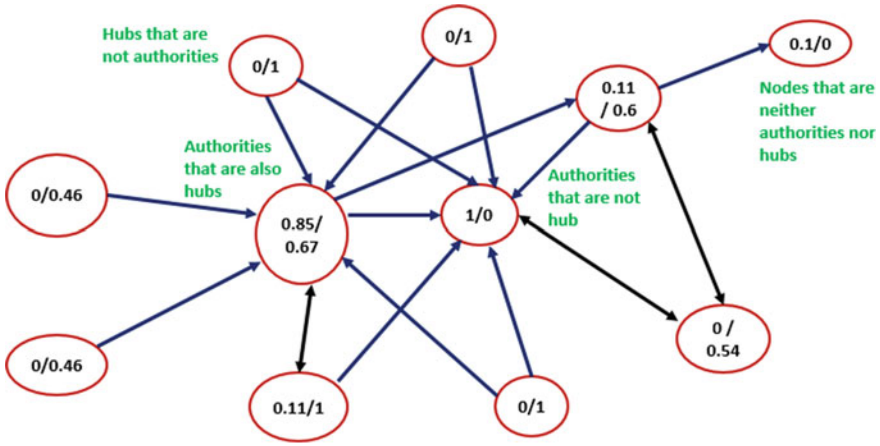


Fig. 2.10 Graph instance with authority and hub scores of the nodes

These two notions of nodes have a mutually recursive definition given as: *A good hub links to many good authorities and a good authority is linked from many good hubs*. Based on this, the *authority* and *hub* scores for a node can be defined as

$$c_{aut}(x) = \sum_{y \to x} c_{hub}(y) \quad and \quad c_{hub}(x) = \sum_{x \to y} c_{aut}(y)$$

Each page i thus has two scores; *Authority score*: $\mathbf{a_i}$and *Hub score*: $\mathbf{h_i}$**.** Thus, HITs algorithm can be defined as

- Initialize: $a_j^{(0)} = {}^1\!/\!\sqrt{n}\,,\quad h_j^{(0)} = {}^1\!/\!\sqrt{n}$
- Keep iterating till convergence:

$$\forall\, i: \; Authority: \; a_i^{(t+1)} = \sum\nolimits_{j \to i} h_j^{(t)}$$
$$\forall\, i: \; Hub: \; h_i^{(t+1)} = \sum\nolimits_{j \to i} a_j^{(t)}$$
$$\forall\, i: \; Normalize: \quad \sum\nolimits_i \left(a_i^{(t+1)}\right)^2 = 1, \quad \sum\nolimits_j \left(h_i^{(t+1)}\right)^2 = 1$$

In vector notation, these formulas can be expressed as per the following explanation:

- Vector $\boldsymbol{a} = (\boldsymbol{a}_1, \boldsymbol{a}_2, \ldots, \boldsymbol{a}_n), \quad \boldsymbol{h} = (\boldsymbol{h}_1, \boldsymbol{h}_2, \ldots, \boldsymbol{h}_n)$
- Adjacency matrix A(n x n): $A_{ij} = 1$ if $i \to j$
- Can rewrite $h_i = \sum_{i \to j} a_j \; as \; h_i = \sum_j A_{ij} a_j$
- So: $\mathbf{h} = \mathbf{A.a}$ and similarly: $\mathbf{a} = \mathbf{A^T.h}$

An interesting result to note by combining the two expressions is that the ***authority score a is an eigenvector corresponding to the largest eigenvalue of*** A^TA. Similarly, ***hub score h is the eigenvector corresponding to the largest eigenvalue of*** AA^T**.**

Figure 2.10 shows the graphical instance of the nodes having authority and hub scores. Hub scores are accumulated based on the outgoing links to the node. Similarly, authority scores are based on the incoming links to the nodes [27]. Moreover, there are nodes that are acting both as hubs and authorities.

This proposed algorithm has found its importance in several fields. Coppola et al. [32] have used the concept of evaluating PageRank scores to evaluate and optimize the global performance of a swarm-based path evaluation for a robot. Zhao et al. [33] have proposed a motif-based PageRank mechanism to find out the top researchers in a citation network. Yin et al. [34] have proposed a variant of the PageRank algorithm, termed as *Signed PageRank* algorithm, to include both positive and negative recommendations from neighbors simultaneously for product recommendation.

De Blas et al. [35] used a weighted HITs centrality algorithm to identify and rank the most influential nodes by considering the impact of relations between the DMUs (Decision Making Units). There are few others reported in the literature [36, 37] which express high utility of the concept in social networks and varied fields. The centrality measure is highly popular in social networks analysis in the field of influence maximization, influencer detection, etc. and thus the class of algorithms belonging to it have a high significance in the current scenario.

2.2.3 Path-Based Measures

In this category of centrality measures, the centrality scores are defined based on the fact that how often a particular path or edge contributes for a node to make its information travel from one part of the network to other parts. This measure is often referred to as the *betweenness centrality measure* which has a close similarity to the *closeness centrality*. Betweenness centrality is the count of the number of times a given node is encountered in the shortest path between the two nodes. On the contrary, closeness centrality weighs the score based on the shortest path only. For example, if there are three shortest paths from node A to node Z, and node B is along two of them, B will be given two-thirds of a point for A to Z pair.

- **Betweenness Centrality**

The notion of *betweenness centrality*, proposed by *Freeman in 1977* [38], has two conjectures: *edge betweenness* and *node betweenness*. However, the notion of edge betweenness finally coincides with the latter, but provides a useful insight of path contribution or the number of paths through which a node ‘x’ can reach node ‘y’ [27]. Let us consider an example for the same as per Fig. 2.11. The figure shows the number of shortest paths from node A to all other nodes in the network. Based on this, the node flow can be defined as

$$node\ flow = 1 + \sum child\ edges$$

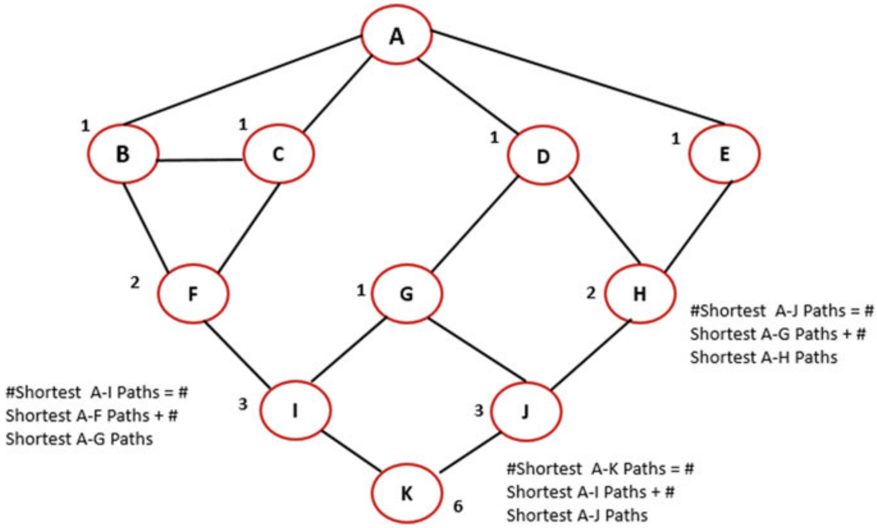


Fig. 2.11 Count of number of the shortest path from node

Further, the flow is split up based on the parent node's contribution. We have to keep exploring the path using BFS (Breadth First Search) mechanism. Multiple paths in between a given source and destination need to be counted fractionally as shown in Fig. 2.12.

This edge betweenness centrality can help us leverage the information to evaluate node betweenness centrality as well. The betweenness centrality for node x can be defined as the probability that the shortest path passes through x. Thus, we have node centrality measure defined as

$$c_{bet}(x) = \sum_{y,\ z \neq x,\ \sigma_{yz} \neq 0} \frac{\sigma_{yz}(x)}{\sigma_{yz}}$$

$$\sigma_{yz}: \quad number\ of\ shortest\ paths\ going\ from\ y\ to\ z$$

$$\sigma_{yz}(x): \quad number\ of\ such\ paths\ that\ pass\ through\ x$$

Removal of nodes in betweenness order causes the network to disrupt as removal of a node with high centrality measure acts as a mediator between the nodes.

As per Fig. 2.13,

- A lies between no other two vertices
- B lies between A and 3 other vertices: C, D, and E
- C lies between 4 pairs of vertices (A, D), (A, E), (B, E)

There are no alternate paths for these pairs to take without C; thus, C has high betweenness centrality. Consider another example.

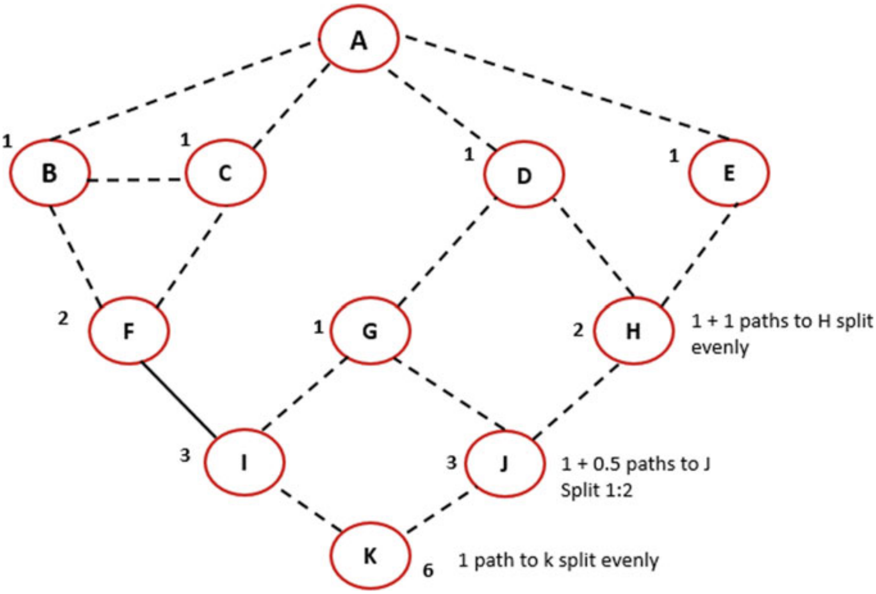


Fig. 2.12 Node flows to the path

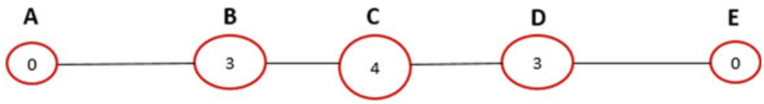


Fig. 2.13 Line graph with betweenness centrality scores of each node

Betweenness centrality score for the graph shown in Fig. 2.14 can be done as follows:

$$\begin{aligned} Betweenness(E) =& A \to F + A \to G + A \to D + A \to C \\ & + B \to F + B \to G + B \to D + B \to C = 8 \end{aligned}$$

Similarly,

$$\begin{aligned} Betweenness(F) =& A \to G + A \to D + A \to C + B \to G + B \to D \\ & + B \to C + E \to G + E \to C + E \to D = 9 \end{aligned}$$

In the same manner, betweenness centrality score calculations for every node of the graph can be done. Being one of the powerful centrality measure, a lot of applications have used this as a metric to develop a problem-solving approach where the interest is to find out the bridges of the network. Daly et al. [39] used this metric

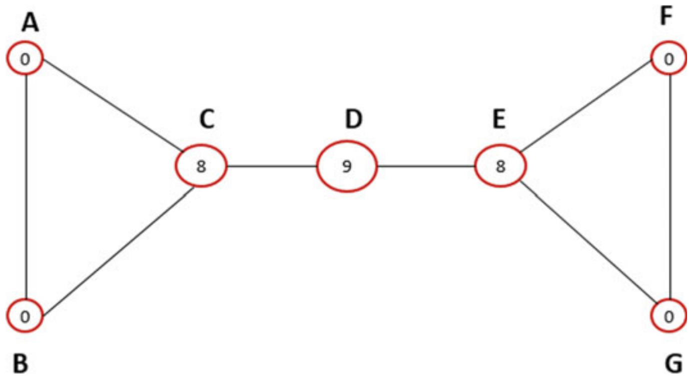


Fig. 2.14 Graph with Betweenness centrality scores of each node

to find out routes in a MANET environment by mapping the concept of small-world dynamics to find out the best message delivery routes. Kazerani et al. [40] discussed how betweenness centrality can be used to model the traffic flow of the cities. Haghir et al. [41] proposed a novel *k-path betweenness centrality* measure where start and endpoints are sampled for path evaluation until we have enough samples to converge. The method is found to have superior performance over the conventional algorithm. Likewise, there are many papers citing the importance of the metric to identify influential or highly important entities in a network that governs the flow of information.

Apart from this categorization of centrality measures, there exists modified versions like applying betweenness and PageRank centrality measure in combination. Then, there exists a notion of *Induced Centrality* measure which is explained at the end of *Katz Centrality* measure which suggests that the importance score of a node raises as soon as it comes in contact with an influential node. Likewise, there are derived versions and variations possible over these centrality measures which provide new evaluation metrics to judge for importance. In the next section, we will see the evaluation of these centrality metrics over real-world graph networks using **SNAP** (***Stanford Network Analysis Platform***).

2.3 Experimental Results and Analysis

To conduct experimental simulations, we have considered *gemsec_facebook_dataset* [42], which contains datasets of 8 different categories of Facebook Page network. The data was collected in November, 2017 through a framework *Graph Embedding with Self Clustering: Facebook* proposed in [43]. The dataset contains a network of various government websites, TV shows' actors, etc. Here the *nodes* represent the *individual entities* while the *edges* between the nodes represent the *mutual likes*. These edge networks have edge lists stored in CSV files where the nodes have been number from index value zero to maintain anonymity. For the purpose of comparative analysis, we considered the graphical network of TV shows where the file contains the edge list and the two TV shows are connected if they are mutually liked upon (undirected graph). Graph contains *3,892 nodes* and *17,662 edges*. The top-10 central nodes identified from various measures are as follows:

These results have been evaluated using SNAP centrality functions. From this score's table, few interesting facts can be determined:

- Node with *node id 2008* has high centrality scores rated by Degree centrality, Closeness centrality, Betweenness centrality, and PageRank centrality measure. Thus, it can be inferred that the TV show is being liked upon the most.
- Eigenvector Centrality scores and HITs centrality scores for the graph have the same top-10 nodes with identical scores. The obvious reason is due to the fact that the graph is undirected and the number of nodes in the shortest path coincides with the hub scores of the node.
- There are a number of nodes in closeness and betweenness centrality that appear in the top-10 central nodes. This is in relation to the first point where the nodes may be ranked.

Different centrality measures have different implications and meanings in the context of the network. In this case, high degree centrality refers to that the node has mutual liking with any other nodes, i.e., a TV show is being mutually liked with many other TV shows. Closeness centrality refers to the close association of the TV shows that have more likings together. Betweenness centrality refers to the shows that are more central in the graph and share likings from one kind of shows to other kinds of shows. In some cases, the centralities too may have a correlation with each other. However, this notion cannot be specific as it entirely depends upon the topology of the graphical network. However, to study upon a highly dense network like this, the centrality trends may be beneficial to identify influential nodes depending upon the objective to be attained. Like high degree nodes will transmit the information and cover the span of the graph. If we want to make the information to pass through particular nodes in maximum routes, betweenness centrality is to be weighted high. If we want to have information localization fast, closeness and eigenvector centrality measures are of high importance. Based upon the scores as per Table 2.1, a scatter plot of **Node ids versus centrality scores** can be determined as per Figs. 2.15 and 2.16.

The degree centrality distribution plot indicates that there are nodes in different regions of the graph having a high degree but are few that lies in the top region of the curve. The majority of the graph settles to the bottom. Closeness centrality seems to have uniform distribution as the closeness centrality takes into account the node's access in minimum distance to other nodes. The curve of the betweenness centrality measure has a smooth increasing trend which suggests there are nodes after every local structure to communicate information from one local region to another. The same is suggested by eigenvector centrality but the increasing trend is rapid as there are a high number of nodes with the shortest path to the majority of nodes in the

Table 2.1 Top-10 central nodes based on various centrality measures

Centrality measure	(Node id, Score)
Degree centrality	(2008, 0.03238), (3254, 0.03238), (3525, 0.02775), (1177, 0.0267), (1673, 0.02621), (3156, 0.02595), (1595, 0.02570), (3122, 0.02570), (2659, 0.02492), (1840, 0.02492)
Closeness centrality	(3254, 0.26098), (2008, 0.25938), (2895, 0.25618), (819, 0.25186), (2751, 0.24777), (211, 0.24425), (160, 0.24422), (3837, 0.24265), (2885, 0.24197), (2035, 0.24014)
Eigenvector centrality	(3525, 0.13852), (1673, 0.13848), (1840, 0.13726), (2659, 0.13690), (3156, 0.13683), (566, 0.13667), (1595, 0.13647), (2036, 0.13641), (1177, 0.13617), (1073, 0.13587)
Betweenness centrality	(3254, 798, 006.33), (2008, 707, 799.88), (819, 609, 148.19), (2170, 565, 442.70), (2751, 565, 010.67), (2895, 522, 963.48), (3038, 307, 027.24), (2682, 294, 900.79), (211, 286, 920.23), (2589, 259, 189.24)
PageRank centrality	(2008, 0.00306), (3254, 0.00305), (2170, 0.00271), (2589, 0.00164), (2076, 0.00161), (412, 0.00161), (2895, 0.001427), (2993, 0.00141), (819, 0.001405), (2524, 0.00136)
HITS centrality	(3525, 0.13852), (1673, 0.13848), (1840, 0.13726), (2659, 0.13690), (3156, 0.13683), (566, 0.13667), (1595, 0.13647), (2036, 0.13641), (1177, 0.13617), (1073, 0.13587)

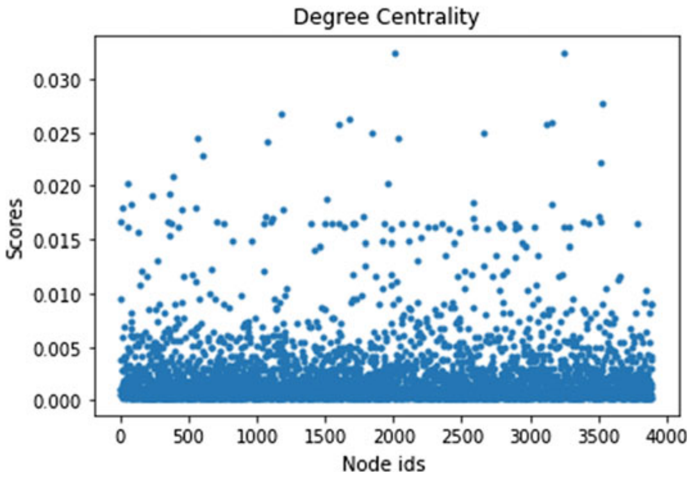


Fig. 2.15 Scatter plot for degree centrality

network. PageRank and HITs centrality have similar trends (Figs. 2.17, 2.18, 2.19, and 2.20).

Another analysis carried out over these centrality measures is how well they are correlated for this graph to each other. Table 2.2 represents the Spearman correlation matrix between the centrality measures. Each cell represents the correlation measure along with the p-value. Correlation between two factors under study is defined in the

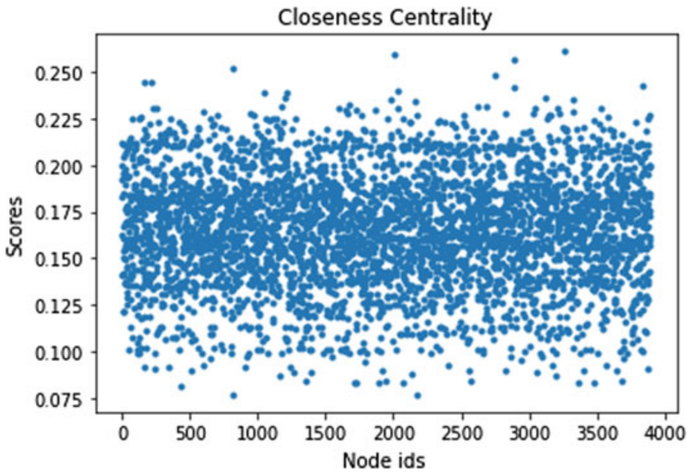


Fig. 2.16 Scatter plot for closeness centrality

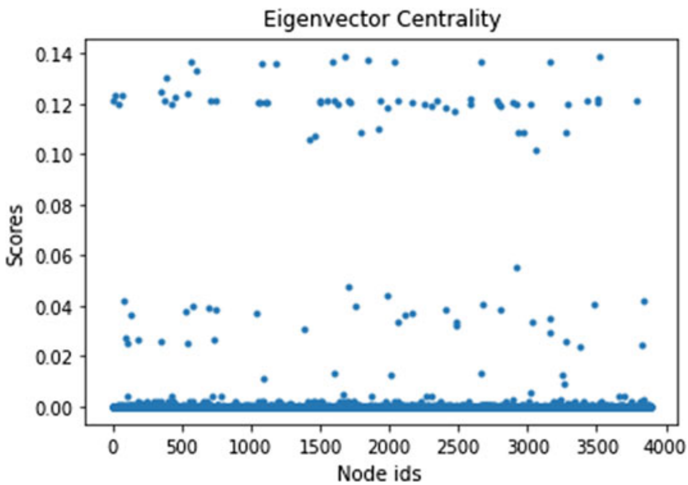


Fig. 2.17 Scatter plot for eigenvector centrality

range [–1, 1]. The strength of the correlation is defined as per the following rules [44]:

- **0.00–0.19—"*very weak*"**
- **0.20–0.39—"*weak*"**
- **0.40–0.59—"*moderate*"**
- **0.60–0.79—"*strong*"**
- **0.80–1.0—"very strong"**

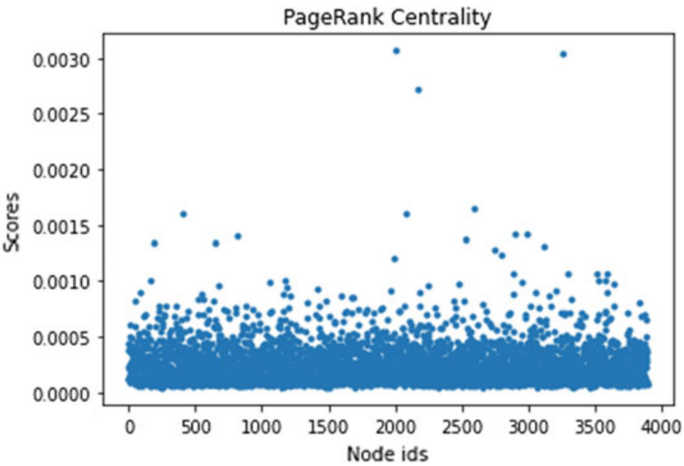


Fig. 2.18 Scatter plot for PageRank centrality

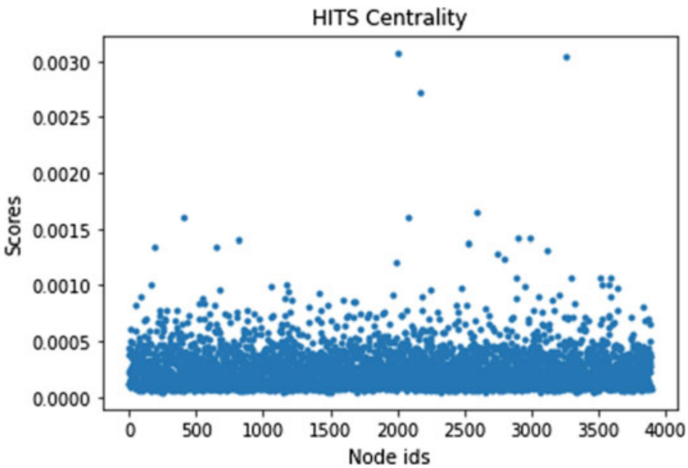


Fig. 2.19 Scatter plot for HITS centrality

The choice for Spearman correlation is due to the fact that it is observed that the centrality distributions are not necessarily normal. The matrix values have been evaluated with the p-value being zero or approximately zero. Degree Centrality has a strong association with the Eigenvector and PageRank Centrality matrix (in the case of undirected network). Similarly, Closeness Centrality has a very high correlation with HITs centrality which suggests that as more nodes accumulate closer, there are more chances of having more hits. There is a strong correlation between the Betweenness as well as Eigenvector Centrality which means that nodes having high betweenness in the network emerge out to be the most liked nodes. Being an

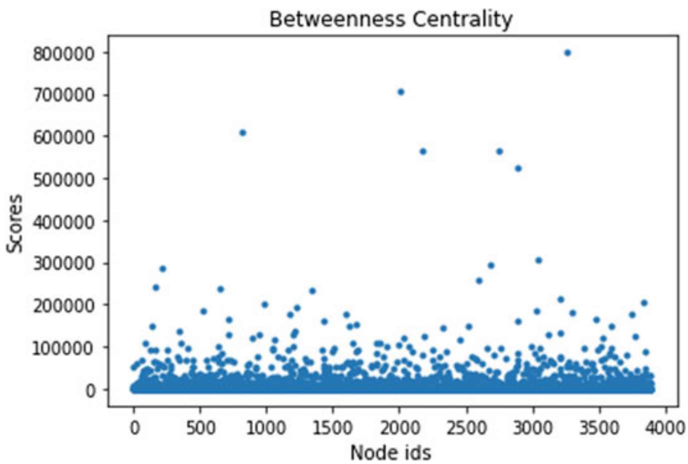


Fig. 2.20 Scatter plot for betweenness centrality

Table 2.2 Spearman correlation matrix

	Degree	Closeness	Betweenness	Eigenvector	PageRank	HITs
Degree		0.552	0.645	0.892	0.892	0.511
Closeness			0.418	0.327	0.327	0.901
Betweenness				0.746	0.349	0.349
Eigenvector					1.0	0.275
PageRank						0.275

undirected graph, Eigenvector and PageRank centrality stand out to be a similar concept as the in links and out links are equated. However, there exists a very weak correlation between the PageRank and HITs Centrality.

2.4 Conclusions

Social Networks being one of the prime sources of connecting real world virtually, the information over it is vast and can be utilized in various ways to earn value from it. The information flow in any network is governed by the number of high importance nodes in the network, and the importance of a particular node is measured on the basis of its position, linking, and its capacity to deliberate the information flow to maximum nodes in the network. This notion gives rise to the concept of network centrality.

This chapter focuses on various centrality measures and deciding criteria to certify a node's importance. Various centrality measures have been categorized into three categories depending upon the referential idea of importance. A detailed investigation has been presented with algorithms and examples for all centrality measures. Further, how a particular centrality measure has been investigated and used by various researchers to solve a particular problem of various domains is also mentioned as and when needed. To understand the concept and significance of centrality, the chapter takes into consideration real-world network's graph (edge list) over which each centrality measure is evaluated, and the results are analyzed over SNAP graphical simulation tool. This detailed analysis and description of the concepts motivate to utilize the knowledge in various domains like protein–protein interaction network, road traffic network, social networks, etc. to evaluate results of significance and identify hotspots of the network. Further, as discussed previously, various combinations of the centrality measures, variation in the conventional centrality measure, etc. can be exploited to identify nodes of high significance and help in building a decision model.

References

1. The Complete History of Social Media: Then And Now. https://smallbiztrends.com/2013/05/the-complete-history-of-social-media-infographic.html. Accessed 19 Apr 2020
2. The rise of Social Media-Our World in Data Homepage. https://ourworldindata.org/rise-of-social-media. Accessed 19 Apr 2020
3. Burt, S., Sparks, L.: E-commerce and the retail process: a review. J. Retail. Consum. Serv. **10**(5), 275–286 (2003)
4. Social Networking Definition-Investopedia. https://www.investopedia.com/terms/s/social-networking.asp. Accessed 19 Apr 2020
5. Graph Theory for skillted. http://blog.soton.ac.uk/skillted/2015/04/05/graph-theory-for-skillted/. Accessed 19 Apr 2020
6. Hogan, B.J.: Networking in everyday life. ON, Canada, University of Toronto, IGI Global, Toronto (2009)
7. Guo, J., Sun, J.: Link intensity prediction of online dating networks based on weighted information. In: 2010 International Conference On Computer Design and Applications, pp. 375. IEEE (2010)
8. Tang, W., Zhuang, H., Tang, J.: Learning to infer social ties in large networks. In: Joint European Conference on Machine Learning and Knowledge Discovery in Databases, pp. 381–397. Springer, Berlin, Heidelberg (2011)
9. Goyal, P., Ferrara, E.: Graph embedding techniques, applications, and performance: a survey. Knowl.-Based Syst. **151**, 78–94 (2018)
10. Roethlisberger, F.J., Dickson, W.J.: Management and the Worker. Psychology Press (2003)
11. Liu, C.C., Chen, Y.C., Tai, S.J.D.: A social network analysis on elementary student engagement in the networked creation community. Comput. Educ. **115**, 114–125 (2017)
12. Ergün, E., Usluel, Y.K.: An analysis of density and degree-centrality according to the social networking structure formed in an online learning environment. J. Educ. Technol. Soc. **19**(4), 34–46 (2016)
13. Zhao, X., Guo, S., Wang, Y.: The node influence analysis in social networks based on structural holes and degree centrality. In: 2017 IEEE International Conference on Computational Science and Engineering (CSE) and IEEE International Conference on Embedded and Ubiquitous Computing (EUC), pp. 708–711. IEEE (2017)
14. Gaharwar, R.D., Shah, D.B.: Use of degree centrality principle in deciding the future leader of the terrorist network. Int. J. Sci. Res. Sci. Technol. **4**(9), 303–310 (2018)

15. Jiang, K., Ding, L., Li, H., Shen, H., Zheng, A., Zhao, F., Yu, S.: Degree centrality and voxel-mirrored homotopic connectivity in children with nocturnal enuresis: a functional MRI study. Neurol. India **66**(5), 1359 (2018)
16. Wikipedia. https://en.wikipedia.org/wiki/Alex_Bavelas. Accessed 21 Apr 2020
17. Kas, M., Carley, K.M., Carley, L.R.: Incremental closeness centrality for dynamically changing social networks. In: Proceedings of the 2013 IEEE/ACM International Conference on Advances in Social Networks Analysis and Mining, pp. 1250–1258 (2018)
18. Tarkowski, M.K., Szczepański, P., Rahwan, T., Michalak, T.P.,Wooldridge, M.: Closeness centrality for networks with overlapping community structure. In: 30th AAAI Conference on Artificial Intelligence, pp. 622–629 (2016)
19. Bergamini, E., Borassi, M., Crescenzi, P., Marino, A., Meyerhenke, H.: Computing top-k closeness centrality faster in unweighted graphs. ACM Trans. Knowl. Disc. Data (TKDD) **13**(5), 1–40 (2019)
20. Wei, B., Deng, Y.: A cluster-growing dimension of complex networks: from the view of node closeness centrality. Phys. A **522**, 80–87 (2019)
21. Goldstein, R., Vitevitch, M.S.: The influence of closeness centrality on lexical processing. Front. Psychol. **8**, 1683 (2017)
22. Bonacich, P.: Some unique properties of eigenvector centrality. Soc. Netw. **29**(4), 555–564 (2017)
23. Neo4j. https://neo4j.com/docs/graph-algorithms/current/labs-algorithms/eigenvector-centrality/. Accessed 19 Apr 2020
24. Carreras, I., Miorandi, D., Canright, G.S., Engø-Monsen, K.: Eigenvector centrality in highly partitioned mobile networks: Principles and applications. In: Advances in biologically inspired information systems, pp. 123–145. Springer, Berlin, Heidelberg (2007)
25. Baldesi, L., Maccari, L., Cigno, R.L.: On the use of eigenvector centrality for cooperative streaming. IEEE Commun. Lett. **21**(9), 1953–1956 (2007)
26. Katz, L.: A new status index derived from sociometrist analysis. Psychometrical **18**(1), 39–43 (1953)
27. CS224W Analysis of Networks. http://snap.stanford.edu/class/cs224w-2018/data.html. Accessed 19 Apr 2020
28. Zhao, J., Yang, T.H., Huang, Y., Holme, P.: Ranking candidate disease genes from gene expression and protein interaction: a Katz-centrality based approach. PloS ONE, **6**(9) (2011)
29. Zhang, Y., Bao, Y., Zhao, S., Chen, J., Tang, J.: Identifying node importance by combining betweenness centrality and katz centrality. In: 2015 International Conference on Cloud Computing and Big Data (CCBD), pp. 354–357. IEEE (2015)
30. Landherr, A., Friedl, B., Heidemann, J.: A critical review of centrality measures in social networks. Bus. Inf. Syst. Eng. **2**(6), 371–385 (2010)
31. Berkhin, P.: A survey on PageRank computing. Internet Math. **2**(1), 73–120 (2005)
32. Coppola, M., Guo, J., Gill, E., de Croon, G.C.H.E.: The PageRank algorithm as a method to optimize swarm behavior through local analysis. Swarm Intell. **13**(3–4), 277–319 (2019)
33. Zhao, H., Xu, X., Song, Y., Lee, D.L., Chen, Z., Gao, H.: Ranking users in social networks with Motif-based PageRank. IEEE Trans. Knowl. Data Eng. (2019)
34. Yin, X., Hu, X., Chen, Y., Yuan, X., Li, B.: Signed-PageRank: an efficient influence maximization framework for signed social networks. IEEE Trans. Knowl. Data Eng. (2019)
35. de Blas, C.S., Martin, J.S., Gonzalez, D.G.: Combined social networks and data envelopment analysis for ranking. Eur. J. Oper. Res. **266**(3), 990–999 (2018)
36. Liu, C., Tang, L., Shan, W.: An extended hits algorithm on bipartite network for features extraction of online customer reviews. Sustainability **10**(5), 1425 (2018)
37. Ka-Wei Lee, R., Hoang, T.A., Lim, E.P.: discovering hidden topical hubs and authorities in online social networks. IEEE Trans. Knowl. Data Eng. 1–1 (2018)
38. Wikipedia. https://en.wikipedia.org/wiki/Betweenness_centrality. Accessed 20 Apr 2020
39. Daly, E.M., Haahr, M.: Social network analysis for routing in disconnected delay-tolerant Manets. In: Proceedings of the 8th ACM international symposium on Mobile ad hoc networking and computing, pp. 32–40. ACM (2018)

40. Kazerani, A., Winter, S.: Can betweenness centrality explain traffic flow. In: 12th AGILE International Conference on Geographic Information Science, pp. 1–9 (2018)
41. Haghir Chehreghani, M., Bifet, A., & Abdessalem, T.: Adaptive algorithms for estimating betweenness and k-path centralities. In: Proceedings of the 28th ACM International Conference on Information and Knowledge Management, pp. 1231–1240. ACM (2019)
42. Stanford University. http://snap.stanford.edu/data/gemsec-Facebook.html. Accessed 14 Apr 2020
43. Cornell University. https://arxiv.org/abs/1802.03997. Accessed 14 Apr 2020
44. Statstutor. http://www.statstutor.ac.uk/resources/uploaded/spearmans.pdf. Accessed 19 Apr 2020

Chapter 3
Temporal Network Motifs: Structure, Roles, Computational Issues, and Its Applications

Atul Kumar Verma and Mahipal Jadeja

Abstract Network is a web of interconnected nodes forming various patterns called motifs. These motifs are also present in the temporal networks, i.e., networks which are tamped with timestamped edges. With the advancement of this temporal network, it is observed that the network is oriented toward the evolution of the motifs with time. So it is crucial to comprehend temporal motifs to study dynamic networks and provide important insights/predictions for them. In this chapter, we study the dynamic structure of motifs. All the key aspects related to the motif and temporal motif structures are explained in depth along with existing work. These multiple temporal motifs provide a finer picture of the network which congruence to the real-world data. It is interesting to note how these temporal motifs elaborated with time. We provide an example of progression in terms of motifs having minimum two nodes, three edges and maximum four nodes, four edges. While computing and analyzing different temporal motifs, a need of parallel algorithm is realized to identify the multiple temporal motifs of various patterns. The chapter also explains a basic application of these concepts by demonstrating the spread of infection such as COVID-19. The chapter serves as a basic starting point for understanding temporal network motifs and these ideas can be used for analyzing many real-world networks/social networks.

3.1 Introduction

In today's era, the social network becomes an indispensable part of people's life everywhere in the world. At present, there are 3.08 billion users across the world and expected to increase to approximately 3.43 billion by 2023 [1]. Let us have a look at its emergence. In early 1960s, the Internet has marked its beginning by the contribution of different public and private sectors in order to enhance the communication through

A. K. Verma (✉) · M. Jadeja
Malaviya National Institute of Technology, Jaipur, India
e-mail: 2019rcp9050@mnit.ac.in

M. Jadeja
e-mail: mahipaljadeja.cse@mnit.ac.in

A. Biswas et al. (eds.), *Principles of Social Networking*,
Smart Innovation, Systems and Technologies 246,
https://doi.org/10.1007/978-981-16-3398-0_3

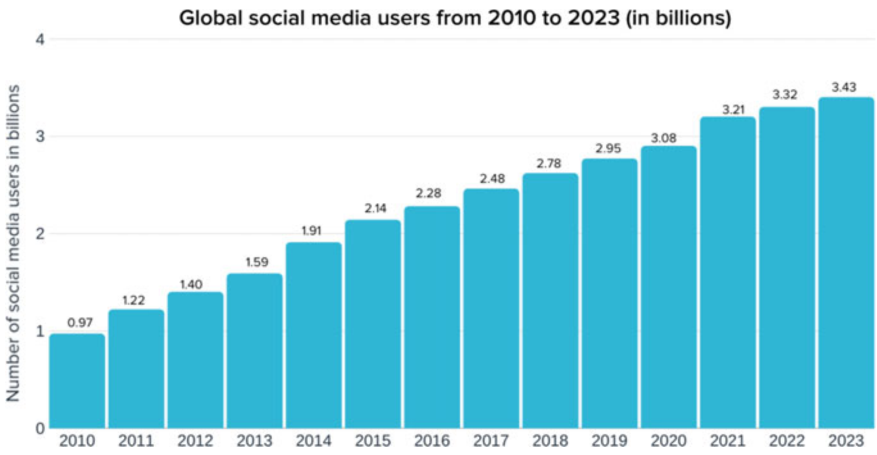


Fig. 3.1 Number of social network users across the world (2010–2023) [1]

computers. By twentieth century, computers become every ones cup of tea, which paved the way to social network emergence. Since the last decade, social networking websites have captivated communication needs so well that now it has become an integral part of almost every human being's daily routine life. The impact has been so powerful that there has been a drastic increase in the number of online apps, news apps, and channels. The news channels have expanded their horizon running from national television broadcasters to live updates on social sites and their own apps. Figure 3.1 indicates the number of social network users across the world in successive years from 2010 to 2023 [1].

This growth in people's involvement over social networking websites has not constrained to this but also results in rapid growth in e-commerce trading [2]. The recent emergence of Amazon, Flipkart, and other E-commerce websites have seen a high jump in the digital as well as retail market. Similarly, YouTube, Netflix, Amazon Prime, etc. have created a whole new world for viewers. In simple words, social networking has made the people of the world to be connected while remaining at their places, still being able to do the most what they can, while being together close physically.

This discussion brings to the conclusion to define social networks as [3]: "Social networking is the use of Internet-based *social media* sites to stay connected with friends, family, colleagues, customers, or clients. Social networking can have a social purpose, a business purpose, or both, through sites such as Facebook, Twitter, LinkedIn, and Instagram, among others. Social networking has become a significant base for marketers seeking to engage customers."

In this section, we will discuss about network motifs, role, and their significance. After discussing upon these basic preliminaries, we will focus on the temporal networks, temporal motifs and their variations, motif counts, and understand evolution of motifs with time.

3.1.1 Network Motifs

Network motifs are significant pattern or subgraphs which occur many times in a random network formed by the interaction of communities [4]. Many networks like the Internet, email networks, social network, and citation network of documents show community structure which includes set of nodes that are interconnected [5]. Community detection displays the concealed yet significant structure in the network such as groups in the underline social network [6]. The network motifs are proved to be helpful in understanding the working of networks and describing the operations and reactions of networks in different scenarios [7].

Consider following examples of network motifs (Fig. 3.2):

i. Feedforward loops like the network of neurons [8], where edge goes from "A" to "B" to "C" and there is a short cut from "A" to "C."
ii. Parallel loop as in food web [9], a network of the food chain or how animals preying on each other. Here animal "A" feeds on animals "B" and "C" as well as animals "B" and "C" are preying on "D."
iii. Single-input modules like networks in gene control, where one node is connected to many nodes which are not connected to one other.

3.1.2 How to Define a Network Motif (Subgraph)

The pattern we are interested in is the induced subgraphs [10]. Induced subgraph is the pattern which is having nodes ($S \subset V$) interconnected with all the edges having both the end point in set (S). We cannot drop out edges in a motif. If we are taking a subgraph of interest then we have to search our induced subgraph of interest, Fig 3.3, instances in the given graph. There is a need of sampling framework for determining the number of temporal subgraphs instances in the temporal network [2]. For example, here we are getting instances of the induced subgraph. We need to check for the number of edges in the same pattern like induced graph then we will

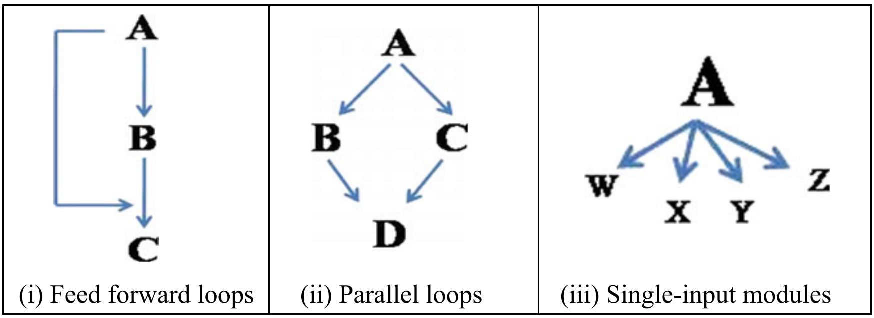


Fig. 3.2 Examples of network

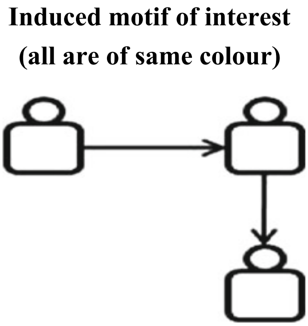


Fig. 3.3 Showing induced subgraph of interest

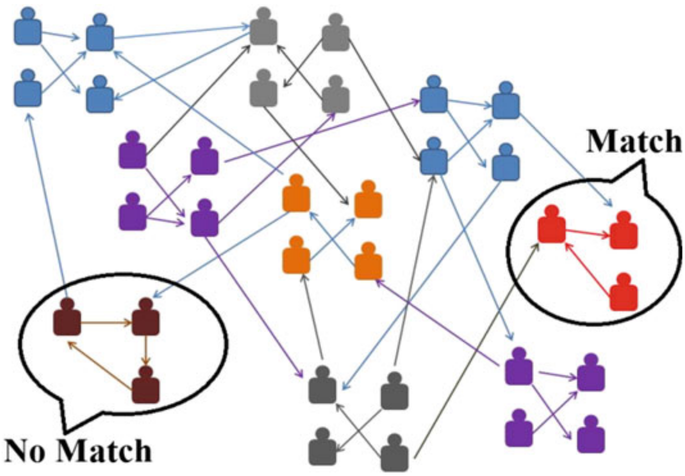


Fig. 3.4 Induced (non) subgraph of the network

declare it as a "Match" and if the pattern does not match with the induced subgraph of our interest then we declare it as a "No Match." Consider the given Fig 3.4, we can find out the same motif in the "Match" section and in case of "No Match" we observe that the motif partially matches because of an extra edge connecting between the nodes so this is a non-induced motif.

3.1.3 *Motifs: Recurrence*

Let us take an example if the considered motif of interest is present or not in the given network. Now we will compare the motifs of interest in the given network and also count how many times the underline motifs is appearing.

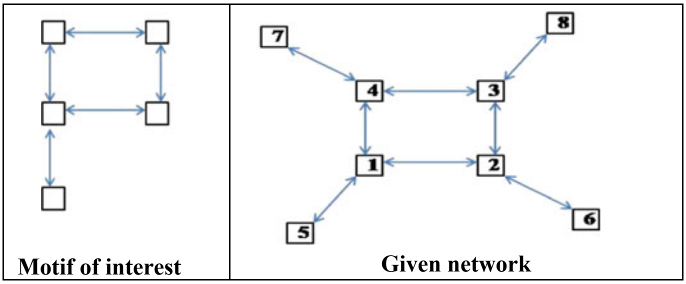


Fig. 3.5 Motif of interest in the network

After observing and analysis of the instances appearing in Fig 3.5, we conclude that there are four instances where the motif of interest is appearing in the given network.

{1, 2, 3, 4, 5}
{1, 2, 3, 4, 6}
{1, 2, 3, 4, 7}
{1, 2, 3, 4, 8}

3.1.4 Significance of Motif

Key idea: If occurrences of a motif are significantly more in a real network as compared to a random network, then the underlying motif is motif of interest. Motifs have to occur in a network much more often or less than what we would expect them under some null model. The idea is that, if we have real network and a motif, then we will ask how often this motif appears in my real network. We will count it. In order to identify whether the underlying motif is motif of interest or not, we have to consider a null model and compare the counts of occurrences of motif in real network as well as null model [11]. For example, we can take four different randomized networks and compare it with the real network for motifs count [12]. Based on the difference of number of motifs (between the real network and randomized network), we can conclude for the chosen motif as significant or not. If the difference is positive we will call it as "Overexpress" and if negative then "Underexpress." Now let us see how computation goes, if we want to know whether the motif is overexpressed or underexpressed in the network compared to some null model, we need to consider Z-score which captures the statistical significance of motif x.

3.1.4.1 Calculation for the Z-Score

- When the real graph is compared with randomized graphs, it is found that subgraphs are over represented in a real once.
- Let Z_x capture statistical importance of subgraph x.

$$Z_x = \left(N_x^{real} - \bar{N}_x^{rand}\right)/std\left(N_x^{rand}\right)$$

where N_x^{real} represents number of motifs of type x in real network (G^{real}).
N_x^{rand} represents number of motifs of type x in randomized network (G^{rand}).

3.1.4.2 Network Significance Profile

$$SP_x = Z_x/\sqrt{\sum Z_y^2}$$

where vector of normalized Z-score represents SP which lay stress on the relative importance of subgraphs. Generally, higher values of Z-scores are shown by a large network.

3.1.4.3 Detection of Motif

We represent networks with their number of nodes, edges, occurrences of motifs in the real network, and randomized network using node degree sequences. Consider three-node connected subgraph, i.e., triads. The triad significance profile (TSP) represents the normalized z-score for each of the triads [13]. Then we can calculate z-score value from Sect. 3.1.4.1. After analyzing the z-score data, it is concluded that the networks of the food web, neuron, and regulation of the gene show overexpress z-scores [12].

3.1.4.4 Role of Motifs

Motifs are important in complex system like a social network, the network of neuroscience, biology, and communication network [14]. These systems can be represented using graphs. In temporal networks, statistical inference methods (analyzing the result as random variation) are still exceptional, even though this problem is vital in temporal graphs of gene regulation [15]. However, these systems evolve with time and show a vigorous change. Such networks or system termed as temporal networks characterized by temporal edges, temporal motifs. For example, models/graphs of temporal networks prove to be fruitful in many ways. Since they show the salient characteristics of a network that can be used in the computation of data.

3.2 Temporal Network

A sequence of static directed graphs over the same (static) set of nodes (V). Each temporal edge is a timestamped ordered pair of nodes $\{e_i = (s, d), t_i\}$ where s, d ϵ set of edges(V), s, d is starting node and end node, respectively, and t_i is the timestamp at which the edge arrives with the evolution of the network (network grows over time when we just look at the edges and nodes). In general, temporal graph is a sequence of the static directed graph overtime, over a fixed set of edges or fixed set of nodes. The notion of the temporal edge is defined by three parameters [16].

(a) source;
(b) the destination; and
(c) the time.

Here in Fig. 3.6, temporal graph of six nodes has some edges, each edge contains time stamp and in Fig. 3.7, at time (t = 1), the edge set {(d, e), (a, b), (d, f), (b, c)} is active and in time (t = 2), the edge set {(a, d), (b, d), (d, e)} is active, similarly at time (t = 3), the edge set {(a, b), (d, e), (d, f)} is active. For example, the edge (e, d) is active for three times, i.e., t = 1, 2, 3. So the temporal graph has a set of edges and nodes which are active at every time stamp.

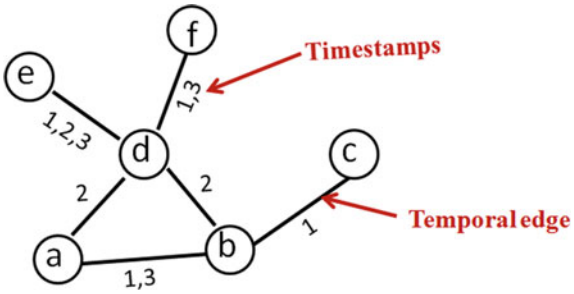


Fig. 3.6 Temporal graph showing temporal edges in different timestamps

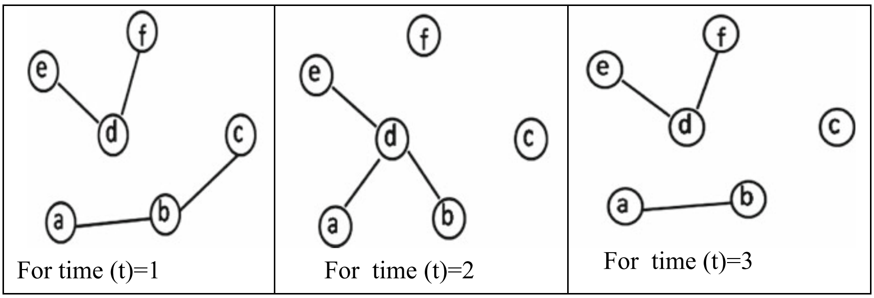


Fig. 3.7 Temporal network with active nodes at different time intervals

3.2.1 Applications of Temporal Network

- Communication networks like email, phone call, and face-to-face network.
- Any kind of interesting proximity networks that create networks of people.
- Transportation network flights and trains. Such things are only active at certain days or a certain part of the year.
- In cell biology interactions like protein–protein, gene regulation.

There are many different examples of networks and that basically have the same nodes but they have different types of relationships as a utility of the time [17].

3.3 Temporal Motifs

Temporal motif is a series of edges which become active in given fixed Δ-temporal time. Let p-nodes, q-edge, and Δ-temporal motif be a series of q-edges $(s_1, d_{1,} t_1)$, $(s_2, d_2, t_2) \ldots \left(s_q, d_q, t_q\right))$ such that

- $t_1 < t_2 \ldots t_q$and $\Delta \geq t_q - t_1$
- The induced static network contains p-nodes.

In other words, the temporal subgraphs are an arrangement of edges in a particular sample, in a particular duration of Δ (time) in the temporal network [18]. Here, the temporal network is shown in Fig. 3.8(X) and (Y) represents temporal motifs. Figure 3.9 indicates different motifs occurring during time $\Delta = 12$ s.

Given timestamp $(\Delta) = 12$ s.

In Fig. 3.8(X) to find induced subgraph (motif), we need to follow two criteria:

Criteria 1: Structure in the sequence of edges means $t_1 < t_2 < t_3$.

Criteria 2: The maximum time between first and last edges timestamps in the motifs $t_q - t_1 \leq \Delta$.

As per scenario in Fig. 3.9:

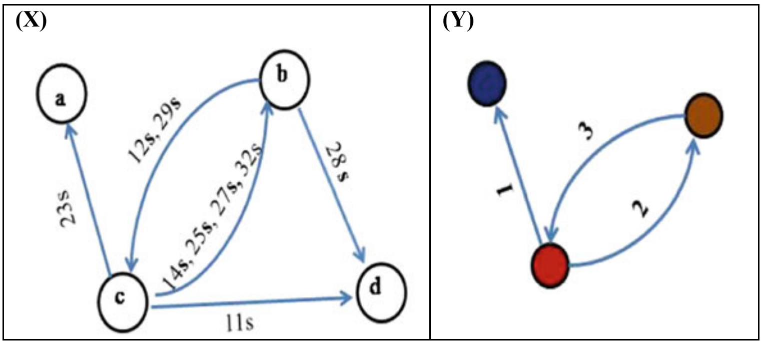


Fig. 3.8 (X) Temporal network, (Y) temporal motif

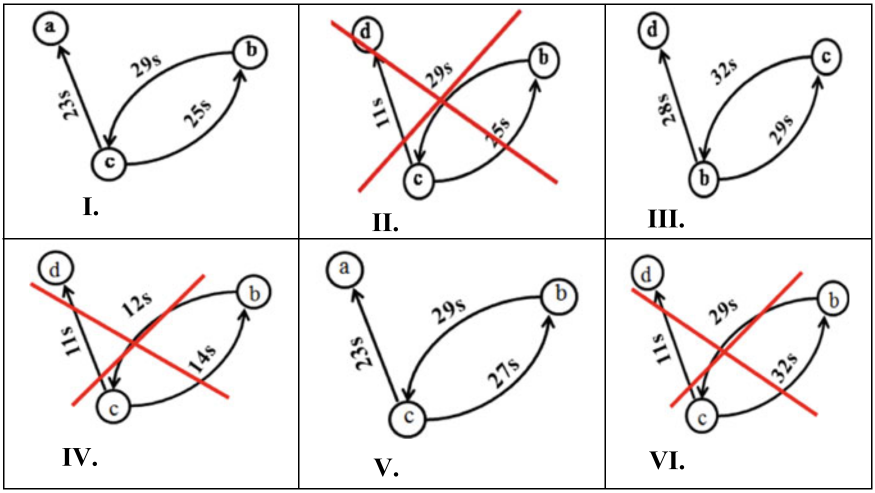


Fig. 3.9 Different motifs occurring during time $\Delta = 12$ s in the temporal network

Case I—$t_1 < t_2 < t_3$ means 23 s < 25 s < 29 s as per given motif and 29 s – 23 s = 6 s, i.e., less than Δ, so both criteria satisfied.

Case II—$t_1 < t_2 < t_3$ means 11 s < 25 s < 29 s as per given motif and 29 s – 11 s = 18 s, i.e., greater than Δ, so criteria 2 not satisfied.

Case III—$t_1 < t_2 < t_3$ means 28 s < 29 s < 32 s as per given motif and 29 s – 23 s = 6 s, i.e., less than Δ, so both criteria satisfied.

Case IV—$t_1 < t_2 < t_3$ means 11 s < 14 s < 12 s as per given motif and 12 s – 11 s = 1 s, i.e., less than Δ, so criteria 1 not satisfied.

Case V—$t_1 < t_2 < t_3$ means 23 s < 27 s < 29 s as per given motif and 29 s – 23 s = 6 s, i.e., less than Δ, so both criteria satisfied.

Case VI—$t_1 < t_2 < t_3$ means 11 s < 32 s < 29 s as per given motif and 29 s – 11 s = 18 s, i.e., less than Δ, so both criteria not satisfied.

Hence, in above example, Case I, III, V are right structures and Case II, IV, VI are wrong structures according to given temporal motif and timestamp ($\Delta = 12$ s).

3.3.1 Temporal Motifs of Two and Three Nodes with Three Edges

In a temporal network, if we observe two- and three-node motifs with three edges, there will be 36 temporal motifs in the give temporal network. We can generate all possible subgraphs and classify them according to their occurrences at different time intervals [19, 20]. For example, yellow color box in the given set defines four temporal motifs having two nodes and three edges. The reason behind these four motifs is the different sequences of edges present in four temporal motifs. Light blue

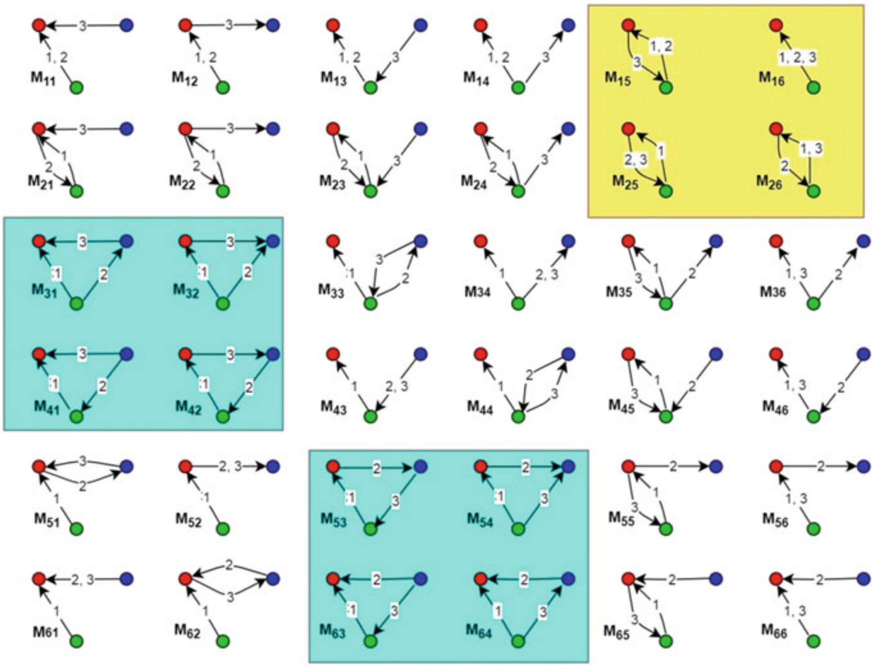


Fig. 3.10 All Δ-temporal motifs with two nodes and three nodes, three edges [19]

color box in the given set defines eight triangle temporal motifs having three nodes and three edges, as well as there are 24 other motifs in the shape of the star. The given set of 36 temporal motifs coordinated in six rows and six columns as $M_{i,j}$ where i and j represent the number of rows and column, respectively. Each motif contains the first edge between orange and green nodes, the second edge of each row is of the same prototype, and the third edge is of the same prototype along with each column (Fig. 3.10).

3.3.1.1 Temporal Motifs of Four Nodes with Four edges

In the given set of temporal network, temporal motifs with four nodes and four edges contain node u in top left and node v in bottom left and two neighbor nodes, i.e., x and y. These temporal motifs further separated in five types, namely, Fig. 3.11. Here, we represent temporal motifs by index according to the parentheses (e.g., tt) [21].

i. Square or circle motifs (sq)—subgraphs forming square.
ii. Tailed-triangle motifs (tt)—subgraphs forming triangle and have an extra "tail."
iii. Star motifs (st)—subgraphs with a single node attached with all edges.
iv. Mid-path motifs (mp)—subgraphs forming path of length three with a double edge at its center.

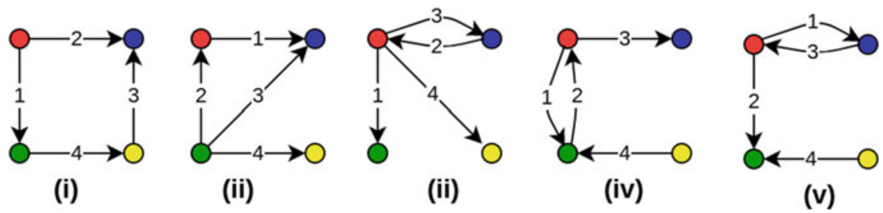


Fig. 3.11 Temporal motif types having four nodes and four edges, (**i**) circle motif, (**ii**) tailed triangle motif, (**iii**) star motif, (**iv**) mid-path motif, (**v**) head-path motif [21]

v. Head-path motifs (hp)—subgraphs forming double edge at the beginning of the path and path of length is three.

In Fig. 3.12, generating all possible subgraphs, all motifs are isomers to those five types of motifs. On analysis, we found that there occurred 39 temporal motifs having undirected four nodes and four edges.

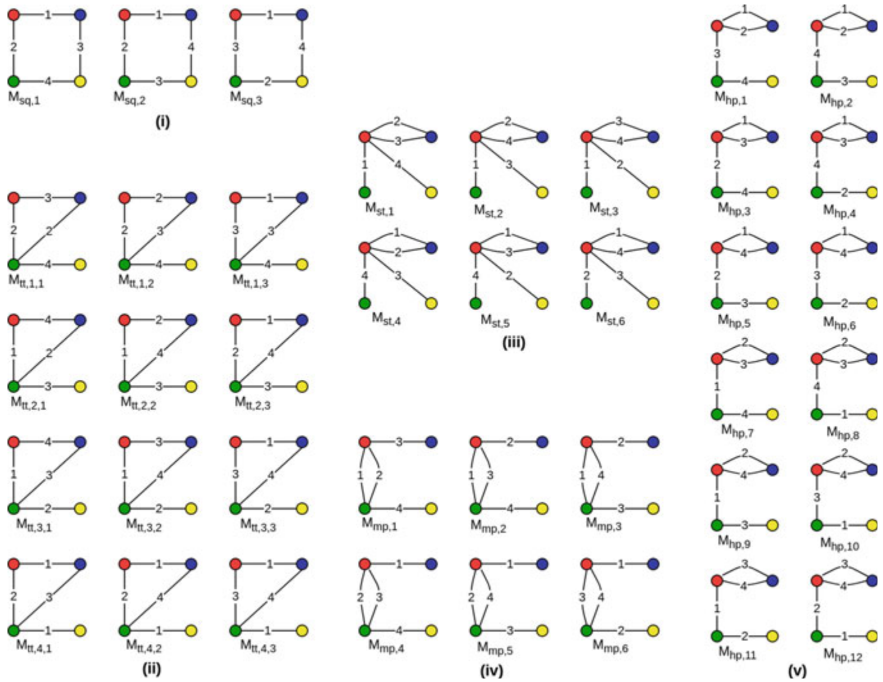


Fig. 3.12 Temporal motifs with four nodes, four edges, and its five types (i) square, (ii) tailed triangle, (iii) star, (iv) mid-path, (v) head-path [21]

3.3.1.2 Computational issues

In finding induced motif in a temporal network, there are some issues that arises such as

i. For identification, there exists a need of computational method for identifying subgraph clusters [12].
ii. It has been observed that there is an exponential growth in number of subgraphs with the number of nodes [12]. This makes computation of motifs slow.
iii. It is difficult to prepare comprehensive record for larger graphs with respect to all motifs and computation of their clusters [12].
iv. Count of temporal star motifs (having four edges, four nodes) is NP-complete problem [21].

3.4 Case Study of Pandemic COVID-19 Involving Data Observation and Analysis

In the early twenty-first century, a pandemic disease called COVID-19 has spread across many parts of the world. The disease reportedly followed a dramatic increase in the number of infected persons and this infectious disease has proven fatal for the life of human beings as well. We discuss the spreading of COVID-19 through network analysis using a temporal network.

3.4.1 Data Analysis

From the centuries, scientists and researchers have been giving their best to understand the spreading dynamics of various epidemic infectious diseases like plague, swine flu, Zika, SARS, and pandemic COVID-19. The analysis involves mathematical modeling in predicting how spreading dynamics of each disease evolve with time. In Fig. 3.13, an analysis of the spreading of pandemic COVID-19 across the world with time is shown [22].

3.4.1.1 Growth of Infected Individuals in the Temporal Network

From Fig. 3.13, it has been realized that growth graph of the spread of COVID-19 is changing dramatically showing an exponential increase with time [22]. In Fig. 3.14, a diagrammatical representation of the growth of infected individuals increasing at an exponential rate is shown. The diagram contains violet color blocks which define the infected individuals or nodes in the network, whereas black arrows indicate the interaction or edges between two individuals.

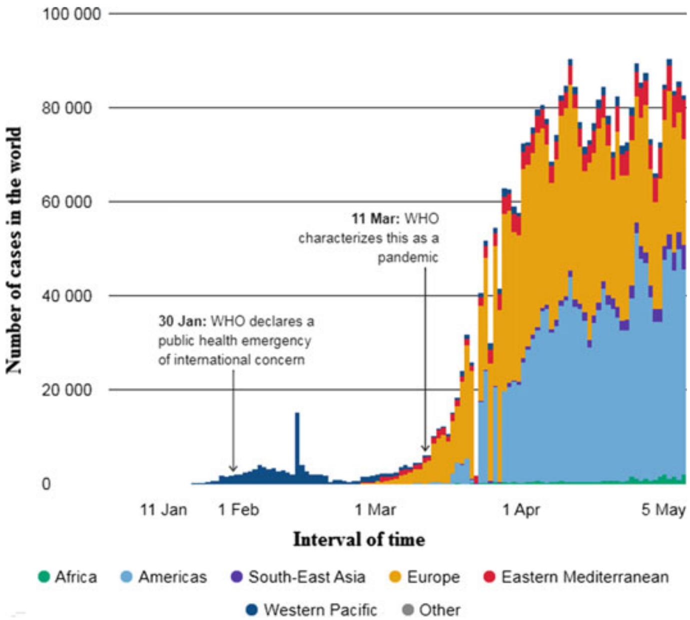
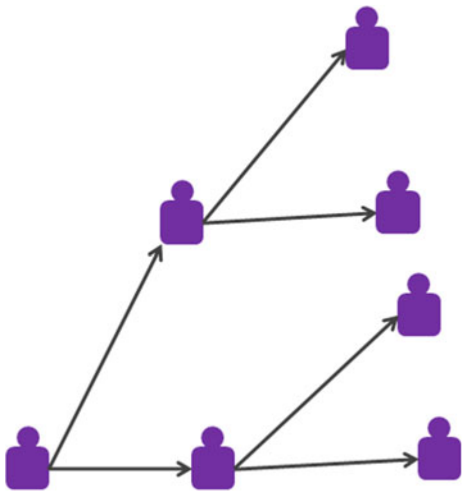


Fig. 3.13 Spreading of pandemic COVID-19 across the world [22]

Fig. 3.14 Diagrammatical representation of the growth of infected individuals in the temporal network



Consider a preliminary disease spread model giving the constant multiplication factor for the growth of infected individuals in a network (see in Fig. 3.14). However, a more realistic approach to model the disease is SIR model, which takes into consideration a more number of factors and variables. In an SIR model, population over which the disease spread pattern is to be studied is divided into three categories: susceptible, infected, and recovered.

3.4.1.2 Infectious Disease Spread Pattern in the Temporal Network

Considering Fig. 3.15, it is a diagrammatical representation of the successive growth of infectious disease (COVID-19). A spread pattern in the timestamp ($\Delta = 15$ days) is given in the randomized temporal network (i). Now let us understand the spreading pattern according to SIR model.

Step (i): This stage is represented as a randomized temporal network, in which each uninfected individual is denoted by white block and is a part of the network. In the span of 5 days, few infected individuals have become part of the randomized network causing the further spread of disease.

Step (ii): This stage is represented as susceptible, in which each uninfected person is denoted by white block, infected individuals by red blocks, and people who are exposed to infection are denoted by blue blocks. In the timestamp of the next 5 days, disease is further transmitted by infected and exposed individuals as a carrier of the disease.

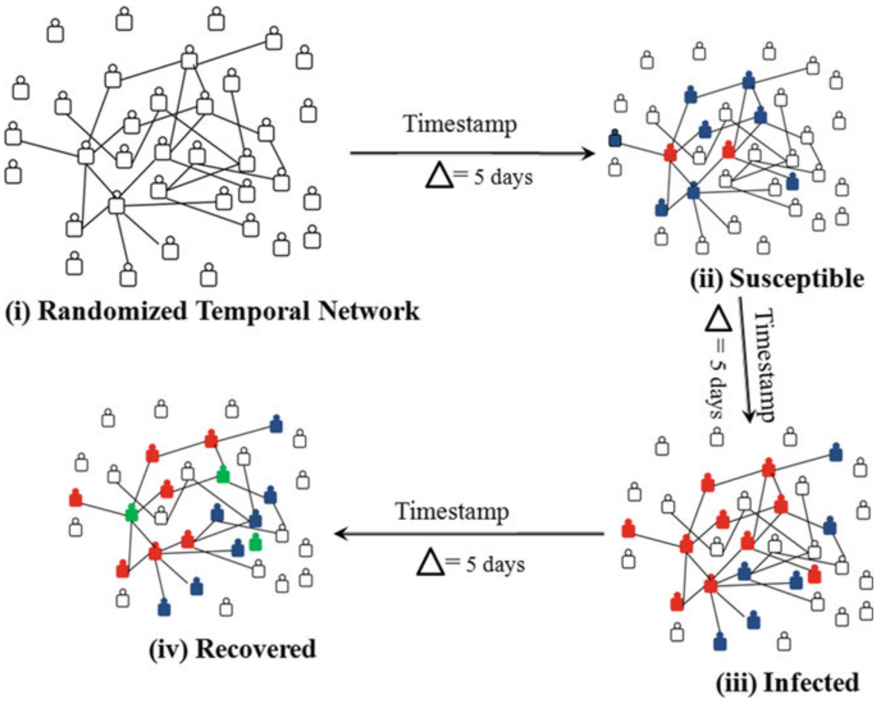


Fig. 3.15 Diagrammatical representation of the successive growth of infectious disease spread pattern in the temporal network at different timestamps (Δ)

Step (iii): This stage is represented as infected, in which each uninfected person is denoted by white block, infected individuals by red blocks, and people who are exposed to infection are denoted by blue blocks. At this stage, the number of infected individuals increased as we observe more number of red blocks and more people got exposed to infection when having interaction with the infected ones. In the timestamp of the next 5 days, more infected people in the network further increase the contamination rate.

Step (iv): This stage is represented as recovered, in which each uninfected person is denoted by white block, infected individuals by red blocks, people who are exposed to infection are denoted by blue blocks, and the individuals who recovered from disease are denoted by green blocks.

Therefore, SIR model represents the successive growth of infectious disease spread pattern in the temporal network at different timestamps (Δ) and helps us to predict the growth rate of COVID-19 infection to some extent.

3.5 Conclusion and Future Scope

Social networks is one of the virtual modern systems for the interaction of people, sharing of thought, information, ideas, opportunities, etc. The data captured through these real networks is accessed and analyzed by the experts. The information flow in any network is done by the people acting as nodes and link between them govern their connectivity. This notion gives rise to the concept of temporal motifs. This chapter focuses on the temporal networks, temporal motifs and its variations, motif counts and explains its evolution with time. A detailed exploration has been presented with examples for some types of temporal motifs. Further, there are few detailed studies that provide how particular temporal motifs have been identified in dynamic temporal networks and are part of the various dynamic real networks as well. Many variants of temporal motif have been used by researchers for different real networks. To explain the concept and significance of temporal motif, the chapter takes into consideration real-world network's graph progression in terms of motifs having minimum two nodes, three edges and maximum four nodes, four edges using different approaches.

This detailed analysis and description of the concept set a pathway to utilize the concept in various domains like communication networks, transportation network, protein–protein interaction network, gene regulation network, social networks, networks of neurons, etc. While computing and analyzing different temporal motifs, a need of parallel algorithm is realized to identify the multiple temporal motifs of various patterns. This future scope will surely help the expansion of the exploration and will be benefited in future research.

References

1. Clement, J.: Number of global social network users 2010–2023. Statistica **4**(7) (2019)
2. Burt, S., Sparks, L.: E-commerce and the retail process: a review. J. Retail. Consum. Serv. **10**(5), 275–286 (2003)
3. Social Networking Definition-Investopedia. https://www.investopedia.com/terms/s/social-networking.asplast. Accessed 19 Apr 2020
4. Braines, D., Felmlee, D., Towsley, D., Tu, K., Whitaker, R.M., Turner, L.D.: The role of motifs in understanding behavior in social and engineered networks.. Next-Gener. Anal. VI **10653**, 106530W (2018). International Society for Optics and Photonics
5. Newman, E.J., Girvan, M.: Finding and evaluating community structure in networks. Phys. Rev. E **69**, 026113 (2004)
6. Fortunato, S.: Community detection in graphs. Phys. Rep. **486**, 75–174 (2010)
7. Alon, U.: Network motifs: theory and experimental approaches. Nat. Rev. Genet. **8**(6), 450–461 (2007)
8. Krauss, P., Zankl, A., Schilling, A., Schulze, H., Metzner, C.: Analysis of structure and dynamics in three-neuron motifs. Front. Comput. Neurosci. **13**, 5 (2019)
9. Paulau, P.V., Feenders, C., Blasius, B.: Motif analysis in directed ordered networks and applications to food webs. Sci. Rep. **5**, 11926 (2015)
10. Stone, L., Simberloff, D., Randrup, Y.A.: Network motifs and their origins. PLoS Comput. Biol. **15**(4) (2019)
11. Tu, K., Li, J., Towsley, D., Braines, D., Turner, L.D.: Network classification in temporal networks using motifs (2018). arXiv:1807.03733
12. Milo, R., Shen-Orr, S., Itzkovitz, S., Kashtan, N., Chklovskii, D., Alon, U.: Network motifs: simple building blocks of complex networks. Science **298**(5594), 824–827 (2002)
13. Milo, R., Itzkovitz, S., Kashtan, N., Levitt, R., Shen-Orr, S., Ayzenshtat, I., Sheffer, M., Alon, U.: Superfamilies of evolved and designed networks. Science **303**(5663), 1538–1542 (2004)
14. Hong-lin, X., Han-bing, Y., Cui-fang, G., Ping, Z.: Social network analysis based on network motifs. J. Appl. Math. **2014** (2014)
15. Holme, P., Saramäki, J.: Temporal networks. Phys. Rep. **519**(3), 97–125 (2012)
16. Sun, X., Tan, Y., Qingbo, Wu., Chen, B., Shen, C.: TM-miner: TFS-based algorithm for mining temporal motifs in large temporal network. IEEE Access **7**, 49778–49789 (2019)
17. Tang, J., Musolesi, M., Mascolo, C., Latora, V., Nicosia, V.: Analysing information flows and key mediators through temporal centrality metrics. In: Proceedings of the 3rd Workshop on Social Network Systems, pp. 1–6 (2010)
18. Takes, F.W., Kosters, W.A., Witte, B., Heemskerk, E.M.: Multiplex network motifs as building blocks of corporate networks. Appl. Netw. Sci. **3**(1), 39 (2018)
19. Paranjape, A., Benson, A. R., & Leskovec, J. (2017, February). Motifs in temporal networks. In Proceedings of the Tenth ACM International Conference on Web Search and Data Mining (pp. 601–610).
20. Li, Y., Lou, Z., Shi, Y., Han, J.: Temporal motifs in heterogeneous information networks. In: MLG Workshop@ KDD (2018)
21. Boekhout, H.D., Kosters, W.A., Takes, F.W.: Efficiently counting complex multilayer temporal motifs in large-scale networks. Comput. Soc. Netw. **6**(1), 1–34 (2019)
22. WHO. https://www.who.int/data/gho/whs-2020-visual-summary. Accessed 23 Apr 2020

Chapter 4
Link Prediction on Social Networks Based on Centrality Measures

Shashank Sheshar Singh, Shivansh Mishra, Ajay Kumar, and Bhaskar Biswas

Abstract In order to understand and compare different social networks, the role of nodes, edges, and their relative importance needs to be investigated. The importance measures of nodes and edges within networks are useful for many social network analysis tasks and are known as centrality measures. These centrality measures allow analysts to investigate network structure, identify influential users, predict future connections, and perform many other related tasks. This chapter presents a study of centrality measures to predict future links in the network. We also choose four different metrics of Recall, Precision, AUPR, and AUC and evaluate the performance of different centrality measures corresponding to the link prediction problem on real-world social networks.

4.1 Introduction

In the twenty-first century, there has been rapid development of online social network unicorns such as Facebook, Twitter, and LinkedIn. This has bought the attention of analysts and researchers toward social network analysis (SNA). The objectives of social network analysis are to analyze the connections between nodes [1–5], node's relative importance [6–10], and investigating network sub-structures [11, 12]. This

S. S. Singh
Department of Computer Science and Engineering, Bennett University, Greater Noida, India
e-mail: Shashank.Singh@bennett.edu.in

S. Mishra (✉) · A. Kumar · B. Biswas
Department of Computer Science & Engineering, Indian Institute of Technology (BHU), Varanasi 221 005, India
e-mail: shivanshmishra.rs.cse18@iitbhu.ac.in

A. Kumar
e-mail: ajayk.rs.cse16@iitbhu.ac.in

B. Biswas
e-mail: bhaskar.cse@iitbhu.ac.in

A. Biswas et al. (eds.), *Principles of Social Networking*,
Smart Innovation, Systems and Technologies 246,
https://doi.org/10.1007/978-981-16-3398-0_4

analysis helps the analysts and researchers to understand what roles a user plays in a network, who are the influential users, analyzing subgroups' connection density, how the epidemics and rumors are spreading through the network, and how users participate in these general processes. There are extensive applications of this analysis that are present in the real world. For example, in the COVID-19 outbreak, the organizations can control and prevent epidemic disease spreading, social media can support contributions and participation of users, business enterprises can help assist suspected and infected customers, and people can gel-up together to understand the situation better and take collective action to support each other and the government.

The measures and methods to analyze the networks have been around for decades but are not entirely satisfactory due to new challenges and opportunities which arise on social media. In practice, social media has a large amount of data making it challenging to analyze influential nodes and predict future links. The identification of influential nodes or central nodes is a fundamental task in SNA. In its elementary form, centrality finds the most central nodes or edges within the network. Therefore, estimating the centrality value is an essential task in SNA. Different centrality measures are suitable for different applications. Time complexity is also responsible for selecting centrality measures for an application due to the sheer size of data as well as the nature of the network. Various centrality measures have been developed over the years based on varying notions of importance. In this chapter, we have reviewed some well-known centrality measures and performed experiments to highlight the centrality selection issue corresponding to the link prediction problem.

The rest of the chapter is organized as follows. Section 4.2 discusses the background information like basic concepts and notations. Section 4.3 reviews the well-known centrality measures. Section 4.4 discusses the link prediction problem and the methodology. Section 4.5 performs experiment corresponding to different applications. Section 4.6 is devoted to conclusion and future works.

4.2 Preliminaries

4.2.1 Notations

Notations used in this study are listed in Table 4.1.

4.2.2 Basic Concepts

A basic social network can be constituted as an undirected graph structure $G(V, E)$, where V and E represent the set of nodes and edges, respectively. The network can be also represented by an adjacency matrix A whose elements are $a_{u,v} \in (0, 1)$, where

Table 4.1 Notations

$G(V, E)$	$\triangleq$	A social network graph with edge set E and vertex set V
$\|N\|$	$\triangleq$	Cardinality of node set V in network
$\|M\|$	$\triangleq$	Cardinality of edge set E in network
$N(u)$	$\triangleq$	The connected neighbors of node u
S	$\triangleq$	Seed set
k	$\triangleq$	Cardinality of seed set ($\|S\|$)
$\sigma(S)$	$\triangleq$	The calculated influence of seed set
$C_i(u)$	$\triangleq$	The node centrality of node u based on any centrality measure i
$C_i(u, v)$	$\triangleq$	The edge centrality of edge (u, v) based on any centrality measure i

$$a_{u,v} = \begin{cases} 1 & \text{if}(u, v) \in E \\ 0 & \text{Otherwise} \end{cases} \tag{4.1}$$

$N(u)$ is the neighbors set of a node u and is formulated as $N(u) = \{v|a_{u,v} = 1\}$. The degree $D(u)$ of a node u in network is defined as $D(u) = |N(u)|$. The walk is a succession of nodes where nodes are connected with links. Nodes and edges can be repeated more than once in a walk. A walk with no repetition of edges and nodes is known as trial and path, respectively. If a path exists between a pair of individuals, then the shortest path between individuals is known as geodesic distance. A network is known as a connected network if there is a path between all pairs of individuals. A network is called directed and weighted if every link in the network is associated with a direction and weight, respectively.

4.2.3 Taxonomy of Centrality Measures

Centrality measures are well-known methods to analyze social networks and investigate the most influential or important elements of the network. Many centrality measures are presented over the decades and used of several important SNA tasks. In 2005, Koschutzki et al. [13] presented a taxonomy for centrality measures as shown in Fig. 4.1.

– **Reachability**. These centrality measures analyze the ability of reaching other nodes of a network when starting a trial walk from a particular source node. For example, degree centrality, eccentricity centrality, closeness centrality, etc.
– **Shortest Path**. These centrality measures evaluate the shortest-path distance from the source node to destination nodes. For example, betweenness centrality, stress centrality, etc.

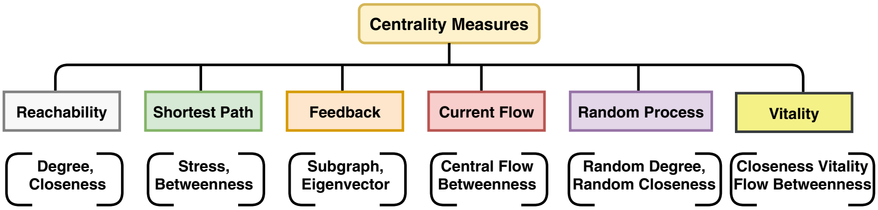


Fig. 4.1 The taxonomy of centrality measures presented by Koschutzki et al. [13]

– **Feedback**. These measures state that a node's importance also depends on its neighbors. For example, functional centrality, eigenvector centrality, subgraph centrality, etc.
– **Current Flow**. These measures consider a scenario when network behavior can be measured by the flow of information or data within the network. For example, current flow closeness/betweenness centrality, etc.
– **Random Process**. These centrality measures are useful for application when a node follows any arbitrary path to propagate information to another node. For example, random betweenness centrality, random degree centrality, random closeness centrality, etc.
– **Vitality**. These measures select an arbitrary function and compute the difference of quantities of function with and without a node considering. For example, flow betweenness vitality, closeness vitality, etc.

4.3 Centrality Measures

Centrality is a fundamental measure for analyzing social networks and it evaluates how central a node or an edge is to the network. It estimates the importance of a node or edge in the network. However, the context of "central" may be varied based on application point of view, therefore the estimation of central also varied. Correspondingly, the centrality of an individual can be measured in different ways. In this section, a discussion on both node and edge centrality is presented.

– **Degree Centrality**. It is the most basic centrality measure. Degree centrality is equal to the node's degree, i.e., $C_D(u) = D(u)$. Degree centrality can be normalized by $C'_D(u) = \frac{D(u)}{N-1}$. The time complexity to measure degree centrality is $O(M)$ for an unweighted network. Degree centrality considers local structure rather than global information. Therefore, it is less complex and suitable for many fast applications.
– **Eigenvector Centrality**. This centrality evaluates the importance of nodes based on their neighbors' importance. Let $A = [a_{uv}]_{N\times N}$ be an adjacency matrix then eigenvector centrality of node u can be computed as $Ax = \lambda x, \lambda x_u = \sum_{v\in V} a_{uv}x_v$,

where $u \in V$ and λ is a constant. The time complexity of measuring eigenvector centrality is of the order $O(N^2)$. It is most suitable and useful for neural networks.

- **Katz Centrality**. This centrality estimates the node's influence by considering the number of possible walks between an arbitrary source and target node pair. The Katz centrality $C_K(u)$ can be calculated as follows.

$$C_K(u) \leftarrow \sum_{i=1}^{\infty} \sum_{v \in V} \alpha^i \left(A^i\right)_{uv} \tag{4.2}$$

where A is adjacency matrix and α is less than the inverse of the value of the largest eigenvalue of A. And $(A^i)_{uv}$ is 1 if there is a possible path between u and v of length i. It is most suitable and applicable for WWW.

- **Closeness Centrality**. This centrality measure estimates the closeness of a node to rest of the network. It is evaluated as average of all the geodesic distances from a node to all other nodes within the network. Therefore, it can be computed as follows.

$$C_C(u) \leftarrow \sum_{v \in V} \frac{1}{gd(u, v)} \tag{4.3}$$

where $gd(u, v)$ denotes geodesic distance between a pair of individuals u and v. The closeness centrality $C_C(u)$ of a node u can be normalized as follows.

$$C'_C(u) \leftarrow \sum_{v \in V} \frac{N-1}{d(u, v)} \tag{4.4}$$

The time complexity of closeness centrality calculation is of the order $O(MN)$ by Brandes' algorithm. It is more preferable than degree centrality. It is useful to identify locations for service facilities such as shopping markets.

- **Subgraph Centrality**. The subgraph centrality of a node is proportional to the reciprocal of order of closed walks. It is defined by the number of possible closed walks and estimated by local spectral moments, i.e., $\mu_i(u) = (A^i)_{uu}$. The subgraph centrality $C_{SG}(u)$ of a node u is calculated as follows.

$$C_{SG}(u) \leftarrow \sum_{i=0}^{\infty} \frac{\mu_i(u)}{i!} \tag{4.5}$$

The time complexity of subgraph centrality calculation is of the order $O(N^2)$ and best suited applicable to molecular connection of spectral structure.

- **Harmonic Centrality**. Harmonic centrality of a node u is an inverse centrality measure equal to the summation of the inverse of all the shortest-path distances between all nodes and u.

$$C_{HM}(u) \leftarrow \sum_{v \neq u} \frac{1}{sd(v,u)} \tag{4.6}$$

Here $sd(v,u)$ is the number of nodes in shortest path between v and u. The complexity of this centrality calculation is of the order $O(M + NlogN)$.

- **PageRank Centrality**. PageRank is an adjustment of the Katz centrality that takes into consideration the fact that the centrality gain using a link from an important node should be decreased depending on how many nodes the central node is connected to (resource allocation). The PageRank centrality of node u is given by

$$C_{PG}(u) \leftarrow c_1 \sum_{k \in N(u)} \frac{a_{k,i}}{d_k} x_k + c_2 \tag{4.7}$$

where c_1 and c_2 are constants and $d_k \implies |N(k)|$. The complexity of calculation of PageRank centrality is of the order $O(M + N)$.

- **Edge Betweenness Centrality**. This centrality measure is different from other shortest-path and closed-path-based centralities as it attempts to quantize the importance of a node to all the shortest paths of the network. The betweenness centrality $C_B(u)$ of a node u is calculated as follows.

$$C_B(u) \leftarrow \sum_{v,w \in V} \frac{sp_{vw}(u)}{sp_{vw}} \tag{4.8}$$

where $sp_{vw}(u)$ and sp_{vw} are equal to the number of shortest paths between v and w which contain u and all possible shortest paths between v and w, respectively. Edge betweenness centrality of an edge e is similarly formulated as the importance of a particular edge to all possible shortest paths in the graph and is equal to the summation of the ratio of all possible shortest paths that pass through the edge with total number of shortest paths.

$$C_{EB}(e) \leftarrow \sum_{u,v \in V} \frac{spe_{uv}(e)}{spe_{uv}} \tag{4.9}$$

where spe_{uv} is the number of shortest paths between u and v nodes, and $spe_{uv}(e)$ is the number of those paths containing edge e. The time complexity is of the order $O(MN)$.

- **Edge Load Centrality**. Another edge-based centrality measure is edge load centrality which is equal to the fraction of all paths of less than a particular length that passes through that edge. It is a slight modification to the edge betweenness centrality such that instead of shortest paths it considers all paths less than a particular cut-off length. The time complexity for edge load centrality is $O(N^3)$.

4.4 Link Prediction

In the current technological scenario, networks have become an important tool in representing complicated relationships between connected components. These components interact with each other, and studying the behavior of links between these components provides important insight into the properties of the underlying network. The problem of predicting new connections between these components is called link prediction. In the case of link prediction, the basic premise is finding out the possibility of a link between two nodes based on the properties and behavior of the nodes displayed up until a given moment of time. The solution to this problem can also lead to a better understanding of the overall network architecture as well as new insight into the modeling and evaluation of such a network, classification of unclassified components, as well as its overall network behavior.

The link prediction problem [1–5] has attracted significant attention of researchers in past few years. Newman [4] was the first to explore the link prediction problem in academic citation networks. This problem was modified to fit the social networking model by Liben-Nowell et al. [14]. Similarity-based link prediction approaches have been explored quiet thoroughly due to their low computational cost for larger graphs and easier algorithmic implementations. In structural similarities-based approaches, different levels of structure (local,global) of the network are extracted to compute the probability score of non-existent links. Some of the structural similarity-based methods which are based on local structure are CN [4], AA [15], RA [16], PA [17], Jaccard [18], CAR [19], LNBCN [20], and NLC [21]. Global similarity indices which are computed based on the entire topological structure of the network are the Katz index [22], Rooted PageRank [23], SimRank [24], etc. Quasi local similarity indices are based on the middle ground between local and global indices which try to incorporate the important characteristics of both approaches. Some examples are Local Path [25] and L3 [26].

Centrality-based link prediction is the problem of predicting future links by estimating the likelihood score of non-existing links using the centrality index of existing links. In order to compute the likelihood score $L_S(u, v)$ of non-existing link (u, v), similarity index of existing link (x, y) is utilized. The similarity index $S_I(x, y)$ of existing link (x, y) can be computed by edge centrality and node centrality, as shown in Eqs. 4.10–4.11, respectively. We can observe that in this case, the edge centrality values can be directly used for similarity index calculation. The dilemma arises in the case of node-based centrality values and how those values can be imposed on the existing edges. For the sake of simplicity, we assume the average of node centrality values to be the similarity index. But other approaches can also be possible such as the one in which different centralities with different attributes (path-based, local information-based, etc.) are used in some combination to calculate a single similarity value of the edge.

$$S_I(x, y) \leftarrow C_i(x, y) \tag{4.10}$$

$$S_I(x, y) \leftarrow \frac{C_j(x) + C_j(y)}{2} \tag{4.11}$$

where i and j represent any edge and node centrality measures, respectively. Now, likelihood score $L_S(u, v)$ can be computed based on common neighbors feature set, as shown in Eq. 4.12.

$$L_S(u, v) \leftarrow \sum_{z} \frac{S_I(z, u) + S_I(z, v)}{\sum_{b \in N(z)} S_I(z, b)} \tag{4.12}$$

where $z \in (N(u) \cap N(v))$.

4.5 Empirical Analysis

4.5.1 Setup Information

In order to analyze the performance link prediction corresponding to different centrality measures, we utilize nine centrality measures and performed experiments on six real-world networks as shown in Table 4.2. All of the experiments were performed on a 64-bit Linux Mint 19.3 PC with Intel(R) Core(TM) i7-3632QM CPU@ 2.20GHz processor and 16GB memory.

4.5.2 Performance Analysis

In this section, the performance of link prediction based on different centrality measures is evaluated and compared based on four accuracy metrics [27]: recall, precision, AUPR, and AUC. In order to obtain different levels of sparsification as well as to better observe the patterns of changes with respect to different centralities, we utilized randomly dividing the original dataset graphs into different sets of observed links (10-50%) as the training set and the remaining links were considered as the test set. The experimental results for these accuracy measures on different datasets are presented in Tables 4.3, 4.4, 4.5, and 4.6.

Table 4.2 The basic information of utilized datasets

Dataset	Nodes	Edges	Avg degree	Diameter
Jazz	198	2742	27.69	6
Celegansneural	297	2148	14.46	5
Airlines	235	1297	11.03	4
NS_U	379	914	4.82	17
SmaGri_U	1059	4917	9.28	−1
Email.Eu.core	1005	16706	33.24	−1

Recall—In link prediction, which can essentially be considered as a binary classification problem, recall measures the ratio of links that are correctly classified as positives to that of the sum of themselves and wrong negative classifications. Table 4.3 demonstrates the recall accuracy measure for the different centralities on different datasets. It can be seen that the edge betweenness and subgraph centrality measures are the best- and worst-performing methods, respectively, for the Jazz dataset over each observed set of links. For the Celegansneural dataset, the Katz and subgraph centrality measures are the best- and worst-performing methods, respectively. The subgraph and harmonic centrality measures are best-performing methods for Airlines and NS_U dataset, respectively. Similarly, the eigenvector and Katz centrality measures are the best-performing methods for SamGri_U and Email.Eu.core datasets, respectively. Edge load centrality performs worst in Airlines, NS_U, and SmaGri_U datasets.

Precision—In link prediction, precision measures the fraction of links that are correctly classified as positives to that of the total number of positive classifications. Table 4.4 demonstrates the precision accuracy measure for different centralities on different datasets. It can be seen that except for the case of the Airlines and Email.Eu.core dataset, the edge load centrality type can be considered to be the one with the worst performance. For the exceptions, the Katz is the one that performs the worst. Degree centrality performs best for the Celegansneural, NS_U, and SamGri_U dataset. Harmonic centrality has the best performance on the Jazz dataset while subgraph centrality performs best on the Airlines dataset. For the biggest dataset Email.Eu.core, we can observe the best performance with closeness centrality.

AUPR—With respect to the AUPR, in Table 4.5, we can see that the Katz centrality has the worst performance in three datasets—Celegansneural, Airlines, and SamGri_U. Edge load centrality has produced the worst values for NS_U, and Email.Eu.core datasets, and edge betweenness centrality has the worst performance for the Jazz dataset. When we observe the best performance, degree centrality performs best in NS_U and SmaGri_U datasets. Closeness centrality performs best in the Email.Eu.core dataset. Harmonic centrality performs best in the Jazz dataset while subgraph centrality performs best in the Airlines dataset. Finally, for the Celegansneural dataset, we observe the best performance in PageRank centrality.

AUC—With respect to AUC, in Table 4.6, for larger datasets NS_U, SmaGri_U, and Email.Eu.core, we observe very minor changes across the different centralities used. Edges betweenness centrality performs worst in the Celegansneural, NS_U, and SmaGri_U datasets while edge load centrality shows worst performance in the Jazz and Email.Eu.core datasets. For the Airlines dataset, we observe the worst performance with the Katz dataset. When best-performing centralities are studied, we observe that degree centrality performs best in the Celegansneural, Airlines,

Table 4.3 Recall metric comparison of different centrality measures for link prediction task

Dataset	Ratio	Centrality measures								
		Degree	Eigenvector	Katz	Closeness	Subgraph	Harmonic	PageRank	Edge betweenness	Edge load
Jazz	0.1	0.97745	0.97236	0.98448	0.98036	0.96145	0.98157	0.98133	0.98642	0.97600
	0.2	0.97194	0.96721	0.97170	0.97000	0.95021	0.97267	0.96891	0.97874	0.96320
	0.3	0.95714	0.94888	0.96371	0.95731	0.92928	0.96290	0.95682	0.96290	0.95002
	0.4	0.93418	0.92871	0.94074	0.94159	0.90106	0.94457	0.93491	0.93631	0.92348
	0.5	0.89720	0.89608	0.89929	0.90892	0.86399	0.90722	0.89589	0.90542	0.88597
Celegansneural	0.1	0.83752	0.83535	0.85488	0.84899	0.78047	0.84961	0.83194	0.83070	0.82202
	0.2	0.81302	0.81209	0.80884	0.80884	0.74837	0.80372	0.81008	0.79659	0.79163
	0.3	0.75804	0.75214	0.74997	0.74109	0.70543	0.74946	0.75783	0.74884	0.74191
	0.4	0.68264	0.68256	0.68023	0.68473	0.66791	0.68403	0.68922	0.67589	0.67496
	0.5	0.58821	0.59156	0.58777	0.58777	0.58864	0.59150	0.58510	0.58374	0.58386
Airlines	0.1	0.83538	0.82154	0.78359	0.81128	0.83949	0.80205	0.83692	0.82051	0.78308
	0.2	0.79821	0.78179	0.76308	0.78000	0.78359	0.79103	0.78462	0.77205	0.76308
	0.3	0.75316	0.76154	0.74068	0.74906	0.75778	0.75282	0.75675	0.73829	0.74188
	0.4	0.72100	0.70674	0.69595	0.70083	0.70662	0.70238	0.70751	0.69904	0.69878
	0.5	0.66502	0.66225	0.65249	0.64222	0.66615	0.64848	0.65393	0.65609	0.66020
NS_U	0.1	0.92319	0.90435	0.91594	0.92101	0.91304	0.92971	0.91884	0.90290	0.90362
	0.2	0.85428	0.84226	0.86266	0.85683	0.83242	0.85246	0.85938	0.83898	0.83716
	0.3	0.76752	0.75830	0.75661	0.76242	0.77261	0.74594	0.75758	0.76509	0.75079
	0.4	0.64863	0.65264	0.65100	0.66011	0.64044	0.64699	0.65173	0.64390	0.64973
	0.5	0.52327	0.51597	0.51072	0.52794	0.51962	0.50999	0.51875	0.52093	0.52268

(continued)

Table 4.3 (continued)

Dataset	Ratio	Centrality measures								
		Degree	Eigenvector	Katz	Closeness	Subgraph	Harmonic	PageRank	Edge betweenness	Edge load
SmaGri_U	0.1	0.75081	0.76016	0.74553	0.74973	0.75461	0.74499	0.75379	0.73699	0.74051
	0.2	0.70650	0.71165	0.70075	0.69600	0.70122	0.70156	0.70705	0.69031	0.69851
	0.3	0.65433	0.65216	0.65261	0.65374	0.64705	0.65645	0.65320	0.64511	0.64244
	0.4	0.56879	0.57509	0.57119	0.57184	0.56777	0.57753	0.57272	0.56302	0.56268
	0.5	0.46989	0.46688	0.46903	0.46911	0.47006	0.47228	0.47331	0.47323	0.46694
Email.Eu.core	0.1	0.93586	0.93474	0.94570	0.94420	0.91072	0.94524	0.93748	0.94922	0.94200
	0.2	0.92779	0.92738	0.93738	0.93474	0.89696	0.93632	0.92893	0.93999	0.93325
	0.3	0.91159	0.91250	0.92552	0.92358	0.88563	0.92169	0.91563	0.92526	0.92001
	0.4	0.89725	0.89566	0.90551	0.90604	0.86734	0.90368	0.89777	0.90528	0.89984
	0.5	0.87130	0.86940	0.87732	0.87672	0.83261	0.87698	0.87056	0.87413	0.86800

Table 4.4 Precision metric comparison of different centrality measures for link prediction task

Dataset	Ratio	Centrality measures								
		Degree	Eigenvector	Katz	Closeness	Subgraph	Harmonic	PageRank	Edge betweenness	Edge load
Jazz	0.1	0.32499	0.31620	0.32951	0.33641	0.26514	0.34167	0.32807	0.21070	0.18809
	0.2	0.44067	0.44680	0.44899	0.45468	0.38844	0.46663	0.44481	0.32554	0.30674
	0.3	0.50253	0.49788	0.51499	0.51370	0.45727	0.52360	0.50365	0.39708	0.37583
	0.4	0.53134	0.53416	0.53101	0.54162	0.49097	0.54356	0.53203	0.43735	0.41464
	0.5	0.53797	0.53176	0.52936	0.53747	0.49303	0.54039	0.53385	0.45640	0.43861
Celegansneural	0.1	0.04396	0.04216	0.03995	0.03996	0.04111	0.04240	0.04231	0.04331	0.04016
	0.2	0.07760	0.07253	0.07296	0.07045	0.07354	0.07257	0.07865	0.07016	0.07071
	0.3	0.09942	0.09706	0.09134	0.09326	0.09288	0.09083	0.10174	0.09017	0.08890
	0.4	0.10803	0.10439	0.09876	0.10220	0.10424	0.10090	0.11118	0.10227	0.10200
	0.5	0.10998	0.10642	0.10238	0.10445	0.10900	0.10248	0.10975	0.10166	0.10378
Airlines	0.1	0.16076	0.14210	0.08681	0.10446	0.18237	0.10983	0.16247	0.16131	0.15051
	0.2	0.22587	0.20515	0.14183	0.16373	0.23833	0.16455	0.23223	0.22343	0.22508
	0.3	0.24910	0.23251	0.16951	0.18953	0.26769	0.19868	0.25778	0.25042	0.25611
	0.4	0.26176	0.24232	0.18475	0.20616	0.27198	0.21338	0.26967	0.25304	0.26358
	0.5	0.25324	0.23145	0.22805	0.20173	0.26868	0.20402	0.25452	0.24548	0.26180
NS_U	0.1	0.15183	0.14632	0.14997	0.13620	0.14730	0.14072	0.15647	0.12648	0.11883
	0.2	0.21446	0.19418	0.21007	0.19466	0.19176	0.20109	0.21400	0.17964	0.17698
	0.3	0.22142	0.21376	0.21861	0.21527	0.21753	0.21058	0.21710	0.20108	0.20076
	0.4	0.20551	0.19563	0.20134	0.20361	0.20253	0.19764	0.20591	0.18877	0.19515
	0.5	0.17506	0.16176	0.16551	0.17157	0.17086	0.16436	0.17214	0.16236	0.16341

(continued)

Table 4.4 (continued)

Dataset	Ratio	Centrality measures								
		Degree	Eigenvector	Katz	Closeness	Subgraph	Harmonic	PageRank	Edge betweenness	Edge load
SmaGri_U	0.1	0.03276	0.03126	0.02726	0.02795	0.02876	0.02867	0.03238	0.03104	0.02894
	0.2	0.05313	0.05165	0.04315	0.04543	0.04946	0.04496	0.05288	0.04702	0.04558
	0.3	0.06692	0.06294	0.05416	0.05537	0.06346	0.05651	0.06585	0.05853	0.06138
	0.4	0.07101	0.06773	0.05672	0.05878	0.06837	0.05964	0.07092	0.06325	0.06504
	0.5	0.06661	0.06373	0.05500	0.05738	0.06683	0.05795	0.06768	0.06307	0.06353
Email.Eu.core	0.1	0.11406	0.11462	0.12854	0.12742	0.08828	0.12746	0.11234	0.08141	0.07594
	0.2	0.18596	0.18928	0.20556	0.20600	0.15244	0.20613	0.18500	0.13938	0.13583
	0.3	0.23734	0.23849	0.25582	0.25786	0.19878	0.25681	0.23933	0.18691	0.17955
	0.4	0.27060	0.26984	0.28316	0.28659	0.23005	0.28688	0.26849	0.21759	0.21349
	0.5	0.28675	0.28394	0.29210	0.29406	0.24775	0.29639	0.28626	0.23469	0.23392

Table 4.5 AUPR metric comparison of different centrality measures for link prediction task

Dataset	Ratio	Centrality measures								
		Degree	Eigenvector	Katz	Closeness	Subgraph	Harmonic	PageRank	Edge betweenness	Edge load
Jazz	0.1	0.32292	0.31414	0.32768	0.33454	0.26289	0.33978	0.32590	0.20842	0.18604
	0.2	0.43946	0.44565	0.44798	0.45365	0.38706	0.46561	0.44362	0.32421	0.30543
	0.3	0.50186	0.49727	0.51446	0.51317	0.45651	0.52304	0.50300	0.39632	0.37506
	0.4	0.53131	0.53430	0.53100	0.54169	0.49085	0.54360	0.53200	0.43733	0.41458
	0.5	0.53953	0.53321	0.53101	0.53895	0.49438	0.54179	0.53556	0.45783	0.43998
Celegansneural	0.1	0.04319	0.04135	0.03936	0.03942	0.04034	0.04176	0.04160	0.04245	0.03940
	0.2	0.07747	0.07243	0.07298	0.07054	0.07341	0.07262	0.07851	0.07005	0.07061
	0.3	0.10064	0.09822	0.09266	0.09455	0.09401	0.09214	0.10286	0.09137	0.09009
	0.4	0.11175	0.10820	0.10272	0.10600	0.10783	0.10471	0.11491	0.10613	0.10575
	0.5	0.11827	0.11471	0.11084	0.11268	0.11727	0.11117	0.11806	0.10991	0.11211
Airlines	0.1	0.15661	0.13835	0.08389	0.10134	0.17813	0.10650	0.15848	0.15705	0.14644
	0.2	0.22359	0.20294	0.13998	0.16177	0.23599	0.16259	0.22986	0.22111	0.22275
	0.3	0.24811	0.23152	0.16877	0.18868	0.26671	0.19783	0.25683	0.24943	0.25513
	0.4	0.26254	0.24315	0.18573	0.20694	0.27272	0.21418	0.27045	0.25388	0.26431
	0.5	0.25707	0.23541	0.23180	0.20548	0.27239	0.20783	0.25824	0.24926	0.26557
NS_U	0.1	0.14974	0.14430	0.14802	0.13449	0.14526	0.13881	0.15439	0.12407	0.11658
	0.2	0.21651	0.19640	0.21189	0.19702	0.19444	0.20342	0.21588	0.18165	0.17877
	0.3	0.23117	0.22350	0.22827	0.22502	0.22697	0.22111	0.22682	0.21038	0.21049
	0.4	0.22769	0.21709	0.22378	0.22596	0.22528	0.22010	0.22781	0.21119	0.21747
	0.5	0.21562	0.20241	0.20652	0.21235	0.21143	0.20673	0.21254	0.20190	0.20344

(continued)

Table 4.5 (continued)

Dataset	Ratio	Centrality measures								
		Degree	Eigenvector	Katz	Closeness	Subgraph	Harmonic	PageRank	Edge betweenness	Edge load
SmaGri_U	0.1	0.03255	0.03110	0.02723	0.02789	0.02856	0.02857	0.03219	0.03079	0.02878
	0.2	0.05368	0.05221	0.04382	0.04609	0.04997	0.04561	0.05342	0.04757	0.04615
	0.3	0.06867	0.06470	0.05598	0.05719	0.06516	0.05832	0.06756	0.06027	0.06314
	0.4	0.07468	0.07138	0.06045	0.06246	0.07196	0.06334	0.07457	0.06695	0.06876
	0.5	0.07335	0.07035	0.06173	0.06402	0.07352	0.06476	0.07441	0.06987	0.07025
Email.Eu.core	0.1	0.11385	0.11441	0.12836	0.12724	0.08807	0.12727	0.11214	0.08124	0.07578
	0.2	0.18595	0.18926	0.20556	0.20600	0.15242	0.20613	0.18497	0.13938	0.13583
	0.3	0.23763	0.23877	0.25611	0.25814	0.19903	0.25711	0.23960	0.18719	0.17984
	0.4	0.27146	0.27069	0.28403	0.28744	0.23090	0.28774	0.26936	0.21845	0.21437
	0.5	0.28893	0.28613	0.29430	0.29625	0.24993	0.29861	0.28845	0.23687	0.23614

Table 4.6 AUC metric comparison of different centrality measures for link prediction task

Dataset	Ratio	Centrality measures								
		Degree	Eigenvector	Katz	Closeness	Subgraph	Harmonic	PageRank	Edge betweenness	Edge load
Jazz	0.1	0.96114	0.95861	0.96670	0.96474	0.94648	0.96639	0.96293	0.94730	0.94137
	0.2	0.95669	0.95591	0.95833	0.95848	0.94403	0.96074	0.95649	0.94328	0.93444
	0.3	0.94976	0.94608	0.95324	0.95157	0.93707	0.95410	0.94976	0.93531	0.92813
	0.4	0.93889	0.93664	0.94062	0.94212	0.92766	0.94274	0.93874	0.92338	0.91749
	0.5	0.92098	0.92105	0.91991	0.92389	0.91230	0.92483	0.91913	0.90809	0.90258
Celegansneural	0.1	0.86192	0.85512	0.86428	0.85701	0.85390	0.86165	0.85746	0.85737	0.85445
	0.2	0.84429	0.84705	0.84353	0.84232	0.83854	0.83907	0.84559	0.83551	0.83723
	0.3	0.81837	0.81767	0.81386	0.81327	0.81322	0.81473	0.81793	0.81206	0.80975
	0.4	0.77625	0.77801	0.77520	0.77882	0.77693	0.77748	0.78099	0.77273	0.77318
	0.5	0.73452	0.73411	0.73232	0.73243	0.73416	0.73321	0.73119	0.73073	0.73261
Airlines	0.1	0.89440	0.88683	0.86420	0.88107	0.90012	0.87476	0.89003	0.88155	0.87076
	0.2	0.88063	0.87236	0.85512	0.86632	0.87441	0.87231	0.87662	0.86460	0.86755
	0.3	0.85404	0.85842	0.84039	0.84888	0.85692	0.85368	0.85459	0.84742	0.85332
	0.4	0.83649	0.82800	0.81632	0.82610	0.83116	0.82674	0.83073	0.82207	0.82851
	0.5	0.79805	0.79700	0.79806	0.79367	0.80186	0.79528	0.79726	0.79422	0.79862
NS_U	0.1	0.95571	0.94570	0.95190	0.95456	0.95015	0.95887	0.95364	0.94362	0.94874
	0.2	0.92090	0.91421	0.92500	0.92200	0.90921	0.92005	0.92338	0.91274	0.91721
	0.3	0.87737	0.87240	0.87178	0.87475	0.87964	0.86654	0.87225	0.87533	0.87327
	0.4	0.81823	0.81975	0.81929	0.82394	0.81399	0.81729	0.81966	0.81552	0.82136
	0.5	0.75629	0.75239	0.74994	0.75861	0.75431	0.74973	0.75396	0.75473	0.75645

(continued)

Table 4.6 (continued)

Dataset	Ratio	Centrality measures								
		Degree	Eigenvector	Katz	Closeness	Subgraph	Harmonic	PageRank	Edge betweenness	Edge load
SmaGri_U	0.1	0.84790	0.85396	0.85003	0.85010	0.85074	0.84703	0.84965	0.84793	0.85002
	0.2	0.82330	0.82596	0.82488	0.82132	0.81956	0.82382	0.82370	0.82060	0.82513
	0.3	0.79401	0.79237	0.79318	0.79249	0.78945	0.79380	0.79296	0.79391	0.79436
	0.4	0.75549	0.75834	0.75546	0.75586	0.75404	0.75905	0.75733	0.75371	0.75522
	0.5	0.71135	0.70941	0.71001	0.71022	0.71105	0.71223	0.71308	0.71313	0.71051
Email.Eu.core	0.1	0.94754	0.94754	0.94992	0.94942	0.93902	0.94948	0.94674	0.94193	0.93947
	0.2	0.94225	0.94309	0.94491	0.94423	0.93401	0.94479	0.94310	0.93570	0.93458
	0.3	0.93434	0.93496	0.93741	0.93749	0.92912	0.93657	0.93612	0.92904	0.92753
	0.4	0.92507	0.92477	0.92560	0.92673	0.91825	0.92595	0.92478	0.91833	0.91738
	0.5	0.90769	0.90674	0.90763	0.90805	0.90179	0.90772	0.90768	0.90136	0.90008

and NS_U datasets. For the Jazz dataset, we observe the best performance with harmonic centrality, and for SmaGri_U dataset, we observe the best performance with eigenvector centrality. Lastly, for the Email.Eu.core dataset, closeness centrality performs best.

4.6 Conclusion and Future Directions

In this paper, we present a method of exploiting centrality measures for link prediction in unweighted undirected networks. Different centrality measures provide different results based on the inherent properties of the social network itself. In order to consistently predict possible links between nodes, the centrality measure has to be chosen according to the properties of the graph itself. We can consistently observe the worst performance with edge-based centralities as well as those having path-based attributes (Katz and eigenvector). The only exception to this pattern is the recall metric where subgraph centrality features very prominently among the best as well as worst-performing methods. A conclusion can be drawn that local information-based centrality measures perform better on unweighted undirected networks. In this work, we have only taken one kind of relationship into account, but we can extend this work by taking multiple such relationships into account which exist in complex networks. Also, we can look at properties of centrality measures as well as their effect on link prediction on weighted as well as signed networks and study them. In the future, we will also attempt to study the effect of different centralities on community detection as well as information diffusion tasks.

References

1. Kumar, A., Mishra, S., Singh, S.S., Singh, K., Biswas, B.: Link prediction in complex networks based on significance of higher-order path index (shopi). Phys. Stat. Mech. Appl. 123790 (2019). http://www.sciencedirect.com/science/article/pii/S0378437119321107
2. Kumar, A., Singh, S.S., Singh, K., Biswas, B.: Level-2 node clustering coefficient-based link prediction. Appl. Intell. **49**(7), 2762–2779 (2019)
3. Kumar, A., Singh, S.S., Singh, K., Biswas, B.: Link prediction techniques, applications, and performance: a survey. Phys. Stat. Mech. Appl. 124289 (2020). http://www.sciencedirect.com/science/article/pii/S0378437120300856
4. Newman, M.E.: Clustering and preferential attachment in growing networks. Phys. Rev. E Stat. Nonlinear, Soft Matter Phys. **64**(2), 025102 (2001)
5. Singh, S.S., Mishra, S., Kumar, A., Biswas, B.: Clp-id: community-based link prediction using information diffusion. Inf. Sci. 514, 402–433 (2020). http://www.sciencedirect.com/science/article/pii/S0020025519310734
6. Singh, S.S., Kumar, A., Singh, K., Biswas, B.: C2im: community based context-aware influence maximization in social networks. Phys. A Stat. Mech. Appl. 514, 796–818 (2019). http://www.sciencedirect.com/science/article/pii/S0378437118312822

7. Singh, S.S., Kumar, A., Singh, K., Biswas, B.: Lapso-im: a learning-based influence maximization approach for social networks. Appl. Soft Comput. 105554 (2019). http://www.sciencedirect.com/science/article/pii/S1568494619303345
8. Singh, S.S., Kumar, A., Singh, K., Biswas, B.: Im-sso: maximizing influence in social networks using social spider optimization. Concurr. Comput. Pract. Exp. **32**(2), e5421 (2020). https://onlinelibrary.wiley.com/doi/abs/10.1002/cpe.5421
9. Singh, S.S., Singh, K., Kumar, A., Biswas, B.: Aco-im: maximizing influence in social networks using ant colony optimization. Soft Comput. 1–23 (2019)
10. Singh, S.S., Singh, K., Kumar, A., Biswas, B.: Mim2: multiple influence maximization across multiple social networks. Phys. A Stat. Mech. Appl. 526, 120902 (2019). http://www.sciencedirect.com/science/article/pii/S037843711930500X
11. Biswas, A., Biswas, B.: Fuzag: fuzzy agglomerative community detection by exploring the notion of self-membership. IEEE Trans. Fuzzy Syst. **26**(5), 2568–2577 (2018). Oct
12. Biswas, A., Biswas, B.: Investigating community structure in perspective of ego network. Expert Syst. Appl. **42**(20), 6913–6934 (2015). http://www.sciencedirect.com/science/article/pii/S0957417415003292
13. Koschützki, D., Lehmann, K.A., Peeters, L., Richter, S., Tenfelde-Podehl, D., Zlotowski, O.: Centrality Indices, pp. 16–61. Springer Berlin Heidelberg, Berlin, Heidelberg (2005). https://doi.org/10.1007/978-3-540-31955-9_3
14. Liben-Nowell, D., Kleinberg, J.: The link prediction problem for social networks. In: Proceedings of the Twelfth International Conference on Information and Knowledge Management. pp. 556–559. CIKM ’03, ACM, New York, USA (2003). https://doi.org/10.1145/956863.956972
15. Adamic, L.A., Adar, E.: Friends and neighbors on the web. Soc. Netw. **25**(3), 211–230 (2003)
16. Zhou, T., Lü, L., Zhang, Y.C.: Predicting missing links via local information. Eur. Phys. J. B **71**(4), 623–630 (2009). Oct
17. Barabsi, A., Jeong, H., Nda, Z., Ravasz, E., Schubert, A., Vicsek, T.: Evolution of the social network of scientific collaborations. Phys. A Stat. Mech. Appl. **311**(3), 590–614 (2002). http://www.sciencedirect.com/science/article/pii/S0378437102007367
18. Jaccard, P.: Distribution de la flore alpine dans le bassin des dranses et dans quelques régions voisines. Bull. Soc. Vaudoise Sci. Nat. **37**, 241–272 (1901)
19. Cannistraci, C.V., Alanis-Lobato, G., Ravasi, T.: From link-prediction in brain connectomes and protein interactomes to the local-community-paradigm in complex networks. Sci. Rep. **3**, 1613 (2013). Apr
20. Liu, Z., Zhang, Q.M., Lü, L., Zhou, T.: Link prediction in complex networks: a local naïve bayes model. EPL (Eur. Lett.) **96**(4), 48007 (2011). http://stacks.iop.org/0295-5075/96/i=4/a=48007
21. Wu, Z., Lin, Y., Wan, H., Jamil, W.: Predicting top-L missing links with node and link clustering information in large-scale networks. J. Stat. Mech. Theory Exp. **8**, 083202 (2016)
22. Katz, L.: A new status index derived from sociometric analysis. Psychometrika **18**(1), 39–43 (1953). Mar
23. Brin, S., Page, L.: The anatomy of a large-scale hypertextual web search engine. In: Seventh International World-Wide Web Conference (WWW 1998) (1998)
24. Jeh, G., Widom, J.: Simrank: A measure of structural-context similarity. In: Proceedings of the Eighth ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, pp. 538–543. KDD ’02, ACM, New York, USA (2002). https://doi.org/10.1145/775047.775126
25. Lü, L., Jin, C.H., Zhou, T.: Similarity index based on local paths for link prediction of complex networks. Phys. Rev. E **80**, (2009)
26. Kovács, I.A., Luck, K., Spirohn, K., Wang, Y., Pollis, C., Schlabach, S., Bian, W., Kim, D.K., Kishore, N., Hao, T., Calderwood, M.A., Vidal, M., Barabási, A.L.: Network-based prediction of protein interactions. bioRxiv (2018). https://www.biorxiv.org/content/early/2018/03/02/275529
27. Manning, C.D., Raghavan, P., Schütze, H.: Introduction to Information Retrieval. Cambridge University Press, New York, USA (2008)

Chapter 5
Community Detection in Social Networks

Sara Ahajjam and Hassan Badir

Abstract Community detection is omnipresent in all complex network analyses. It consists of dividing networks into a set of nodes that are highly interconnected. These groups of nodes allow a better understanding of the structure of the networks and widely help in complex network analysis. The community detection problem has been dealt with in many research papers; however, the definition of the community and the proposed methods used for detecting communities differ based on community definition and the type of network dealt with. In this chapter, we provide the state-of-the-art existing community detection methods based on different criteria.

5.1 Introduction

Complex networks are ubiquitous; they represent a new method to model the voluminous quantity of data coming from real systems in graphs, using notions of graph theory [1]. This paradigm is used to model a wide variety of systems in different disciplines. The most explicit examples are social networks in human science, protein networks in biology, citation networks in education and research, and many others. In social networks, we represent the interactions and the relationships between persons; these interactions could be via friendship relation or by sharing the same interests by liking the same Facebook pages and/or following the same persons. Another example of complex networks is biological networks that modeled the interactions between proteins. Besides, we can cite, for example, the citation networks such as Google Scholarship, ResearchGate, and others. All these networks are modeled by graphs where the nodes of the graph describe the actors of the phenomena, i.e. persons in social networks or articles in citation networks, and the links represent the interactions between the nodes, i.e. the friendship between two persons in a social network such as Facebook is represented by a link in the graph, an article that was cited in another article is modeled by a link connecting these two articles....

S. Ahajjam (✉) · H. Badir
Data and Systems Engineering, National School of Applied Sciences of Tangier,
Abdelmalek Essaadi University, Tangier, Morocco

A. Biswas et al. (eds.), *Principles of Social Networking*,
Smart Innovation, Systems and Technologies 246,
https://doi.org/10.1007/978-981-16-3398-0_5

Complex graphs are widely used in modeling any type of complex network, and as its name describes, complex means large-scale networks containing millions and billions of nodes and edges. Generally, analyzing and understanding the structure of these networks and the interactions between nodes remain a complex task. The analysis of complex networks is a multidisciplinary subject that deals with notions of sociology, computer science, and mathematics. Recently, it got a lot of attention in several research works. Despite these varied origins, complex networks have non-trivial properties that they frequently share [2]: nodes have heterogeneous degrees approximated by a power law distribution, i.e. many nodes are very connected acting as centrals. In particular, the average distance between two nodes is often small compared to the total distance of the graph [3]. This property regarding Milgram's experiment is called the "small world" [4]. It has also been shown that the large-scale graphs have a structure in communities, i.e. groups of nodes very connected to each other, and little connection to the other nodes of the network. The study of complex graphs raises many scientific questions. First, their measurement is never a trivial activity. For example, their size is often not the same as while measuring a part, the rest evolves at the same time inducing many biases. The visualization of networks is often delicate and rarely very effective and it is necessary to define analysis tools allowing them to be described and their properties to be easily understood. The size of the networks and their structural peculiarities also give rise to new algorithmic questions and their modeling, which would make it possible to predict their evolution, is still limited because the models often do not reflect reality, hence the detection of communities.

Community detection has been and is still very much investigated in the literature. It represents a field of complex networks analysis [1, 5–10], consisting of characterizing their structure at a mesoscopic level, i.e. which is neither that of the node (microscopic) nor that of the entire network (macroscopic). The problem of detecting communities is akin to partitioning graphs and clustering data. Nevertheless, it differs from it by the structure and the rather large size of the complex networks (large terrain graphs). For now, the problem has not been formally modeled, but the authors agree on an intuitive definition of the objective of this task: to get communities with nodes highly interconnected compared to other nodes of the network [11–14].

In this chapter, we represent the community in the context of complex networks, especially social networks, and then we define the different methods to analyze the community structure in networks. A state of the art of community detection algorithms, resulting from the most fruitful approaches to this problem, is presented. It is difficult to cover all approaches in the field exhaustively; however, we will select methods for their fundamental aspect or their effectiveness. We then give an overview of the methods of community detection using leader nodes.

5.2 Community Definition

A community can be defined intuitively based on use cases like a political party, working groups, interest groups, etc. All these definitions described communities such as clusters, groups, and partitions. The community has been studied in several domains and research studies; however, instead of papers that deal with these problematic, the definition of community remains unclear [10, 13, 15–18]. As cited in [1, 2, 19], "the main problem of graph partitioning is to find a quantitative definition of community. Until today, no definition is universally accepted".

Generally, the community concept is more used in social sciences, where the community could be a group of people living in the same city or a group of students in the same college. If we use the social network example, a community could be a group of people with the same interest. Generally, in all domains, communities are relatively similar, and they refer to a set of nodes highly interconnected and weakly linked to the remaining nodes of the network. Otherwise, the community is defined as a substructure of the network whose density of links between community nodes is greater than the density of links between communities, i.e. the nodes of the community are strongly connected inside the community than outside.

A community could be defined locally and globally.

Locally: Communities are formed with consideration to the local network of nodes, i.e. the node and its neighbors. Communities can be defined in some systems as separate entities or groups with their own autonomy, which does not depend on the entire network/graph. They are considered as parts of the network or graph densely interconnected and sparsely extra-connected [10, 20].

Gobally: The communities are defined by considering the entire graph structure and topology. It seems reasonable when the structure of the community is partitioning the graph into several sets of nodes. Identification of communities is based on the intrinsic idea that the community structure is provided by the graph if it is not random [7, 21].

Several definitions of community exist in the literature. However, works dealing with community detection often focus on creating new methods for detecting communities and less on what is exactly a community. Implicitly, the community is defined based on the method used for detecting communities, and the community definition could differ relative to the application domains.

5.3 Community Structure

Detecting communities is important for better network analysis and the comprehension of complex network structure. The first community structure studied in the literature was disjoint communities with each single node is part of only one community [8, 22–25, 31]. In this case, the community is a partition. Subsequently, another study proposed that a node may belong to one or more communities, hence the notion of overlap between communities [26–29].

Communities could be formed based on different structures:

- Community structure based on nodes connectivity:
  Methods defining community structure based on node connectivity use the links or edges relations to define communities. A clique is the most famous community structure using node connectivity [30]. The work of [31] defines community as a clique, where all the nodes of the community should be adjacent. In literature, there are many derivatives of this definition, where the CFinder algorithm [32] is an example. The CFinder algorithm defines the community as a consecution of adjacent k-cliques where a k-clique is a set of k adjacent nodes. The immediate advantage of such an approach is overlapping communities detection, where a vertex may be connected to several k-cliques that are not necessarily adjacent. Another study [17] using nodes' connectivity for defining the community structure classify the communities into weak and strong communities. Strong communities are communities where nodes' internal degrees are greater than nodes' external degrees. While in a weak community, the internal degree total of these nodes is greater than the external degree total of these nodes. The study proposed by Fu et al. [33] classifies communities into two categories: community leader and self-organized community. The leader-community is the community where there is a node that is considered as the leader, and this node is highly connected to the nodes of its community, which means that its degree is higher. Otherwise, this node is central. While for the other type of community, all nodes have almost equal or similar degrees (Figs. 5.1 and 5.2).
- Community structure based on nodes similarity:
  Another community structure has been proposed, which is based on the similarity between the nodes. Using the example of social networks, members belong to the same Facebook page because they share the same interests and therefore similar. Also in citation networks, papers in the same community shared generally the same themes. Therefore, a community can be considered as a group of similar nodes. Several studies have been interested in the similarity between nodes, and the most basic method to calculate the similarity between two nodes is to calculate the intersection of the neighborhood of these nodes, i.e. how many neighboring nodes they share. The similarity between nodes can be achieved globally or locally [11, 34–39], by considering just the adjacent neighbors or by taking into account the whole connections in the networks. The most similarity functions used in literature are the Jaccard similarity, cosine similarity, HPI similarity, and HDI similarity.
- Community structure based on quality function:
  The quality function is a measure that was developed to quantify the quality of the detected communities. The main usage of the quality function is that at each step of a partitioning process, we compute the quality function to measure the quality of the detected partition, and we stop once the quality function remains stable. The most used quality function is modularity, which was developed by Newman [40],

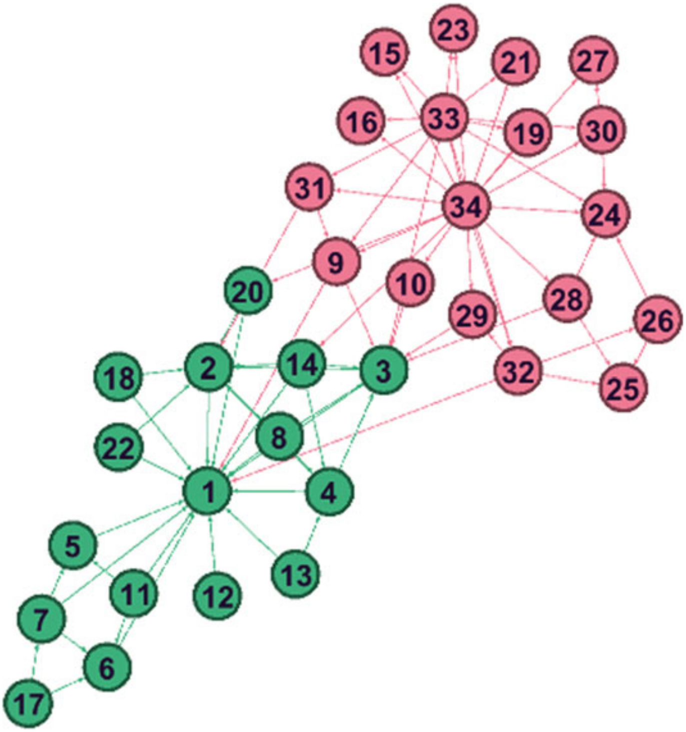


Fig. 5.1 Community structure—nodes with the same color belong to the same community

that consists of measuring the strength of network communities. Networks with high modularity have high inter-community connections and sparse connections between nodes that do not belong to the same communities.

5.4 Community Detection Methods

– Methods using random walks:
  Approaches based on random walks are random processes where a walker positioned on a network node can move randomly on one of the neighboring nodes. The structure of the network influences the result obtained by using random walks. The most used algorithm that uses random walks for community detection is Walktrap [41]. Walktrap uses a distance between vertices based on random steps. A short random walk, starting from a given vertex, tends to stay in the community (s) of that vertex. Thus, the distance between the results of two random walks starting from two distinct vertices effectively reveals whether or not these vertices belong to the same community. This distance allows this method to partition the graph

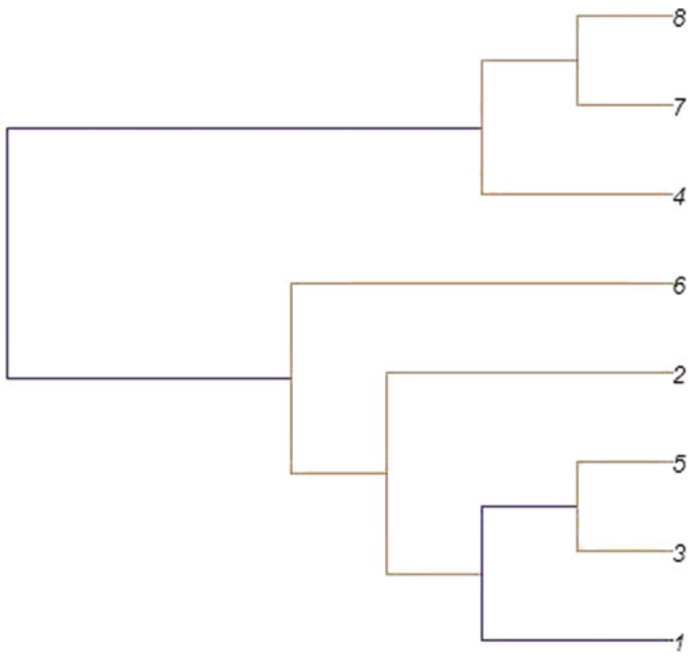


Fig. 5.2 Dendrogram

through a hierarchical clustering algorithm. Another algorithm named Infomap [42] like WalkTrap exploits the fact that a walker randomly following the edges of the graph tends to get stuck in communities. The work of [44] considers that a random walker starting from a node i can choose his path randomly between the available links and can at any iteration t return to his starting point with a probability α. The process used in this work strengthens the connections between nearby nodes for the random walker to move to nodes that are in the same community as the starting node. An overlap measure has been proposed to bring together overlapping communities above a certain threshold β of overlap.

– Methods using graph partitioning:
  Graph partitioning consists of dividing the graph into k subgraphs. Initially, the graph is divided into two partitions and each partition will be divided recursively until the desired number of communities is obtained. Communities are divided in such a way that intra-community ties are weak and inter-community links are strong. The algorithm of [44] seeks to find a cut of the graph by swapping nodes between partitions to decrease the number of links connecting communities. Approaches based on graph partitioning require prior knowledge of the number and size of communities to be detected. Therefore, the final result depends on this information introduced at the start of the detection process.

– Community detection using hierarchical clustering:
  Hierarchical clustering is a method of graph partitioning that consists of dividing the graph hierarchically [45, 46]. This method divides the graph in the form of a hierarchy of partitions represented in a tree, also called a dendrogram. The dendrogram is a tree whose leaves are the points in a dataset where each node represents a cluster. There exists two hierarchical clustering methods: Divisive methods and agglomerative methods. Depending on the used method (divisive or agglomerative), the hierarchical structure is enriched with each merging or separation operation. In all cases, the output is a tree representing the hierarchical structure of the network.

Divisive Methods remove gradually inter-community edges to isolate communities, i.e. all the nodes are considered as a unique graph and edges connected nodes with the weakest similarity are removed. At each step, the graph divides into several connected components (the communities) and, in the end, each node represents a community.

Agglomerative Methods merge small communities into larger based on a similarity or proximity measure between the communities, with atomic communities containing only one vertex as a starting point. We start with each node in a different community and, with each iteration, the two closest or similar communities are merged, until all the nodes are in the same community. The most basic proximity measures used to merge nodes to form communities are as follows:

– **Minimal proximity (Single link)** is a string-like method and is suitable for non-elliptical shapes. This method defines the proximity of the cluster as being the shortest distance connecting two nodes i and j that are in two different clusters *C1* and *C2*. The disadvantage of this method is that it is susceptible to outliers, i.e. nodes that are not linked to other nodes.
– **Maximum proximity (Full link)** can split large clusters and favors globular shapes. This method defines cluster proximity as the farthest distance between two points, i and j, located in different clusters, *C1* and *C2*. It is less sensitive to aberrant nodes.
– **Group average proximity** introduces the cluster proximity as the average distance between two nodes i and j located in different clusters *C1* and *C2*.
– **Structural proximity** was used in the work of [43] that introduces a distance that measures the structural proximity of the vertices of a graph. This distance is based on the analysis of random walks that tend to get trapped in dense areas of communities. The idea is to compare the probability distributions of random walks starting from two vertices to compare the proximity of these two vertices. This distance is used in a hierarchical clustering algorithm to detect a hierarchical structure of communities.

The agglomerative method successively brings together the closest communities to form a hierarchical structure. The result can be seen as a tree structure, called a dendrogram (Fig. 3.2), between communities that divide into sub-communities. This hierarchical structure will be used to determine the most significant partitions that maximize the different possible quality functions. The result obtained using hierarchical clustering depends on the measure of similarity or proximity used. One of the most known agglomerative methods is the Clique Percolation Method [72]. The Clique Percolation (CP) approach considers communities to be a collection of overlapping cliques. First, cliques of size k are identified, and each clique is represented by a node. Then the cliques that share *k-1* nodes are merged to form communities.

– Methods using partitional clustering:
  Partitional clustering consists of dividing a set of nodes into non-overlapped clusters. Each node of the network belongs to only one cluster. The most famous algorithm based on partitional clustering is the "k-means" algorithm [48]. First, k "central" nodes will be chosen at random, which will be designated as the center of each community. Then for each node chosen, the closest nodes will be assigned to form the communities. At each iteration, we recalculate the center of the community, and therefore another node can be chosen as the central node. The algorithm will be iterated until no node changes communities. The disadvantage of this algorithm is that the final result depends on the introduced number k of communities to be detected.

To spread this weakness, several improvements of k-means have been proposed. An approach proposed by MacQueen [48] consists of using the classical algorithm of community detection with the PCA mapping and k-means with local expansion to select the initial nodes that will be introduced for community detection using k-means. Since PCA can keep distance information of each pair of nodes, the improved algorithm uses PCA to map nodes in the lower dimensional complex network and then detects the initial nodes using the improved local expansion strategy. The local expansion strategy suggests that the chosen nodes should have a large degree and use a full graph as input to be better than using simple nodes. The resulting nodes of this strategy will be used as the nodes in the k-means algorithm to form the k communities. Partitional clustering methods are susceptible to outliers.

– Methods using spectral clustering:
  Spectral clustering formalizes the data-clustering problem into a graph-partitioning problem without defining the shape or the community structure. Usually, spectral methods transform the similarity matrix of the network into a more normalized form and use the eigenvalues of the matrix to find the communities. There are several graph splits used in spectral clustering, the most used is "RatioCut" [49, 50], and "NCut" [51] which are both standardizations of the "Cut" method [51]. "Cut" partitions the graph into k communities of the same size by minimizing the weight of the edges connecting the communities. These functions are minimized when the nodes are grouped into large communities with a few inter-community edges.

In the work of [52], the number of communities is concluded by the distribution of eigenvalues in the Laplacian matrix. The k small eigenvalues are computed and the matrix in these k eigenvectors as the column is built. This matrix will be introduced as input for the k-means algorithm for clustering. Another algorithm named Node2vec-SC introduces node2vec to the SC method to detect communities. Node2vec learns models ordering vertices according to their community and their network roles. The node2vec algorithm popularizes prior work and models the full spectrum of equivalences detected in networks. Afterward, the SC algorithm is used to detect the network communities [52].

– Methods based on modularity:
  Modularity, as mentioned above, is a function measuring the quality of partitions, initially introduced by Girvan and Newman, to choose a privileged cut in a dendrogram resulting from a hierarchical clustering [53]. Several studies are based on modularity optimization, the best known are the Louvain algorithm and the FastGreedy [16]. Louvain [54] uses a Greedy modularity optimization method. Initially, each vertex is in its community and each vertex takes the community of one of its neighbors so that the gain is maximum modularity. An improvement of the Louvain algorithm has been proposed by Blondel et al. [55] consists of optimizing Q at each level by the vertex mover method which moves the nodes toward a neighboring community until Q no longer increases. FastGreedy [16] like Louvain method is based on a greedy optimization of modularity. Starting from the finest partitioning (one node per community), communities gathered gradually as the value of modularity increases. One of the strengths of this method is its timeliness. In the work of [56], to detect disjoint communities, edges are deleted iteratively. The algorithm partitions the graph into strongly connected communities based on the connection strength between vertices by optimizing modularity. To optimize community structure, the isolated nodes are affected to their initial communities.
– Community detection using central nodes:
  The majority of complex networks have a Free-Scale structure which results in a minority of nodes having a high degree, and most nodes have few connections so a low degree. In addition, complex networks analysis revealed that it exists, in each community, a node that plays a key role in that community [57–61]. This member is responsible for disseminating information and attracting new members to this community. The central/leader nodes are identified using the centrality measures. Centrality is a measure, defined in graph theory, that identifies the important nodes in a network. Centrality was dealt on many research works in the field of network analysis [62–64]. In the Freeman research paper [65], the centrality was classified into three groups:

- The global measures classify the nodes based on the whole structure of the network. Betweenness centrality and proximity centrality are the most utilized global measures. They are efficient in classifying the central nodes in the network, but they are of great computational complexity.

- The local measures identify the central nodes of the network based on their local network (the node and its neighbors). Degree centrality is one of the most simple local measures, but they ignore the whole structure of the network which makes the local measures less effective.
- Measures based on random walks like PageRank [66] and LeaderRank [67, 68] demonstrate important performances in directed and non-directed networks.

In this paragraph, we illustrate the existing research works that combine the community detection approaches and the leader/central node detection methods. Beni et al. proposed a new method named TI-SC algorithm (TI-SC) for detecting community detection inspired by the Louvain algorithm. This method has four steps: detect communities using the Louvain algorithm. Next, those communities are merged based on relationships between cores nodes. The primary seed node is identified based on the highest score of the adjacent neighborhood and the neighbors of neighbors. Finally, updating the scoring criteria by decreasing the scores of neighbors of seed nodes [60]. The most proposed approaches of community detection detect communities at first after they identify central nodes in each community. However, recently, it was demonstrated in some research works [20, 69] that the central/leader plays an important role in building communities. For that, recent works detect central nodes at first, and for each node, similar nodes are assigned to form communities. The works of [70, 71] start by identifying leader nodes using eigenvalue centrality and for each leader, similar nodes are assigned based on node connectivity or node similarity. The "Leader-Follower" algorithm [31] defines the community as a clique where it is formed by a central node and at least one follower. The loyal follower is a node that belongs to one community and does not have neighbors (adjacent nodes) in another community, i.e. all its adjacent nodes should be belonging to the same community as the nodes. The leader is the node of the highest distance centrality compared to its neighborhood. The communities are formed by deleting edges arbitrarily, and each node will be affected by the community of their neighbors. The main drawback of this method is that the communities with only leaders will be deleted. Another method based on the same community structure named the Clique Percolation Method was proposed by Ahajjamet al. [72].

The work of Javadi et al. identifies the leader nodes as nodes with the greatest degree centrality and the community leaders as the intersection of the nodes existing in all the maximum cliques containing the nodes with the greatest degree (leaders nodes). Other nodes of the networks are affected by the corresponding leader based on the connectivity index. The connectivity index plays an important role in assigning nodes to the community to increase inter-community connectivity and weaken extra-community connectivity. This method does not allow overlap. For that, nodes that belong to multiple communities are assigned to the community with the greatest Jaccard similarity [59].

The "TopLeaders" algorithm [73] covers the drawback of the "Leader-Follower" algorithm [31] by proposing a new method for community detection that deals with outliers, i.e. nodes that do not belong to any leader. Inspired by k-means, the "TopLeaders" algorithm detects k nodes randomly and the communities are formed based on the proximity of the nodes from the central nodes. At each step, the centrality of all the nodes of the community is computed and the leader is the node with the highest degree. In the “Follow the Leader” algorithm, a new guided clustering centrality is proposed for the detection of leader nodes and communities. Nodes with great centrality are chosen as the starting point. The approach has three stages: First, grouping the vertices into different groups, then the groups with a high overlapping factor are merged, and finally, the vertices in the same groups are contracted to form a new vertex [74]. Another hierarchical clustering algorithm uses rival leaders to divide the network into partitions. Leaders are the nodes with the greatest degree centrality. If two leaders belong to the same partition, the partition will be divided so that each contains only one leader. The cut is made by choosing the shortest path between the leaders and the process is stopped if the shortest path from a node to its leader is less than three. Otherwise, a new rival leader is selected, and the process is repeated [75].

Other research uses the spectrum of the graph to define important nodes: The algorithm of [50] is one of the methods that uses the spectrum of the graph to define leader nodes. This approach defines leaders as the relative changes in the x largest eigenvalues of the adjacency matrix after the deletion of the node on the study. In this work, two categories of nodes were identified: the “core nodes” representing the important nodes to the community and are considered as central, and the “bridge nodes” connecting the communities. The work of Chen et al. defines central nodes based on community centrality. Leaders are selected based on a high-density index and the distance from other network nodes with a large density index. The leaders’ neighbors will be assigned to him to form the communities [76]. Fang et al. propose a new algorithm LDA [77] for detecting communities based on the leader through three main stages: identification of leaders, identification of classes of followers, and assignment of leaders to clusters. Leaders are selected using leadership centrality or a node is defined as a leader if its leadership centrality is greater than at least one of its neighbors. Followers are also identified by the centrality of leadership, where the follower with the highest leadership value and his neighbors form the first class of followers. Follower classes are used to form communities, where each leader is assigned to the class that contains their neighbors. Assigning leaders to different classes of followers allows for overlapping communities.

In the same perspective, Lu et al. propose the DPNMF (Density Peaks Nonnegative Matrix Factorization) algorithm that considers that core nodes of the community are distant from each other and are of high density surrounded by low-density nodes forming communities. The authors proposed improved DP to obtain the core nodes and so the number of communities in the network which will be as input to the NMF parameter [78]. Sun et al. propose the AutoLeader algorithm for community detection using leaders’ nodes. In this work, the authors build a forest based on multiple dependent trees, inspiring by the idea that each node is likely attached to its local leader. Tiny branches of the tree are merged, and each tree in the forest is considered as a community where the root node represents the leader of the corresponding cluster [79].

Table 5.1 Synthesis of community detection methods using central nodes

Algorithms	Leader type	Number of leaders	Leader detection	Community detection
LeadersRank	One node	Calculated	Centrality	Proximity
LCDA	One node	Calculated	Centrality	Similarity
Leader-Follower	One node	Calculated	Distance centrality	Proximity
Jiang and Wu [59]	One node	Predefined	Degree	Proximity
TopLeaders	One node	Predefined	Random	Proximity
Follow the Leader	One node	Calculated	Centrality	Proximity
Algorithme de clustering hiérarchique	One node	Calculated	Centrality	Proximity
Wang et al. [50]	One node	–	Eigenvalues changes	Proximity
CPM	Subgraph	Predefined	Random	Proximity
K-means	One node	Predefined	Random	Proximity
Javadi et al. [60]	One node	Calculated	Density	Proximity
LDA	One node	Calculated	Centrality	Proximity
TI-SC	One node or set of nodes	Calculated	Neighborhood	Modularity
DPNMF	One node	Predefined	High density	NMF
AutoLeader	One node (root node)	Calculated	Based on network topology	Proximity

Table 5.1 represents a synthesis of the different approaches cited based on the type of leaders (one node or set of nodes), the number of leaders (calculated or predefined), selection of leaders (randomly or based on a certain measure like centrality), and community detection (based on similarity or proximity). Those criteria have an important impact on the obtained results.

Comparison of the different algorithms showed that the usage of a random number of leaders to select community leaders does not provide a deterministic result and the number of leaders affects the number of detected communities. Besides, as we can observe from our daily usage of social networks, leaders or influential are unique persons. Consequently, the type of leader nodes shouldn't be as group of nodes or subgraphs. Moreover, approaches used for leader detection play an important role in community detection. Methods that randomly select leaders are not deterministic and the constructed communities will change based on the initially selected leaders. Also, the selected leaders differ based on the used centrality measures. The local centrality measures don't take into account the whole structure of the network, which can involve a loss of information, and the detected communities will be constructed based on the local network of leaders. The global centrality measures give efficient results; however, they are time-consuming.

Another important criterion is that leaders should be distant, and even after the detection of leaders, similar leaders should be merged into the same community. Besides, the definition of similarity and proximity between nodes used for building communities influences the allocation of nodes to leaders. If leaders are chosen using local centrality measures and communities are detected using the proximity of other nodes from the leader, so the final communities will be as local networks of the selected central/leader nodes.

5.5 Conclusion

The detection of communities is a much-investigated research subject given its usage. Its plays an important role in analyzing and understanding the structures of complex networks. These application cases are very diverse, ranging from marketing and targeting in social networks and e-commerce sites to testing treatments on proteins in biological networks.

In this chapter, we have illustrated the different definitions and structures of community, as well as the different approaches to detecting communities and leader nodes in complex networks. Several methods and algorithms have been proposed to respond to the problem of detecting communities in complex networks. However, each method differs from the other in its definition of the community and its structure and therefore its application case. Also, a comparative study of the different methods was proposed.

Existing methods provide interesting results, especially methods that use central/leader nodes for community detection. However, these methods deal with complex networks in general, and social networks in particular, as static networks.

The main property of social networks is their dynamic aspect. Some existing approaches in literature take account of the dynamic aspect of social networks. The existing methods of community detection in dynamic networks used the static methods between two snapshots to define the network's changes to decide if there are changes in the detected communities. However, this field remains in progress compared to the proposed methods for community detection in static complex networks.

References

1. Cherifi, H.: Complex Networks and their Applications. Cambridge Scholars Publishing (2014)
2. Graph Partitioning and Graph Clustering in Theory and Practice (195.084). Research Group Parallel Computing. TU Wien. http://www.par.tuwien.ac.at/teaching/2015w/195.084.psp. Last accessed 10 Jan 2017
3. Fortunato, S.: Community detection in graphs. Phys. Rep. **486**, 75–174 (2011). https://doi.org/10.1016/j.physrep.2009.11.002
4. Milgram, S.: The Small World Problem (1967)

5. Orman, G.K., Labatut, V.: A comparison of community detection algorithms on artificial networks. In: Gama, J., Costa, V.S., Jorge, A.M., Brazdil, P.B. (eds.) Discovery Science, pp. 242–256. Springer, Berlin Heidelberg (2009)
6. Davendra, D., Zelinka, I., Senkerik, R., Pluhacek, M.: Complex network analysis of discrete self-organising migrating algorithm. In: Zelinka, I., Suganthan, P.N., Chen, G., Snasel, V., Abraham, A., and Rössler, O. (eds.) Nostradamus 2014: Prediction, Modeling and Analysis of Complex Systems, pp. 161–174. Springer International Publishing (2014)
7. Yang, Z., Algesheimer, R., Tessone, C.J.: A comparative analysis of community detection algorithms on artificial networks. Sci. Rep. **6**, 30750 (2016). https://doi.org/10.1038/srep30750
8. Clustering and community detection in directed networks: a survey (PDF)—semantic scholar. https://www.semanticscholar.org/paper/Clustering-and-Community-Detection-in-Directed-Malliaros-Vazirgiannis/86be7f7c5888013068ccac9095ed7da6282216b7/pdf. Last accessed 12 June 2016
9. Ahajjam, S., Badir, H., Fissoune, R., Haddad, M.E.: Communities identification using nodes features. In: Esposito, F., Pivert, O., Hacid, M.-S., Raś, Z.W., and Ferilli, S. (eds.) Foundations of Intelligent Systems, pp. 303–312. Springer International Publishing (2015). https://doi.org/10.1007/978-3-319-25252-0-33
10. Newman, M.E.J.: Community detection and graph partitioning. EPL Europhys. Lett. **103**, 28003 (2013). https://doi.org/10.1209/0295-5075/103/28003
11. Cheng, J.J., Wang, P.F., Zhang, Q.B., Zhang, Z.Q., Leng, M.W., Xu, H., Chen, X.Y.: A community detection algorithm based on node degree difference and node similarity. In: Applied Mechanics and Materials, vol. 462–463, pp. 458–461. Trans Tech Publications Ltd (2013). https://doi.org/10.4028/www.scientific.net/AMM.462-463.458
12. Schaub, M.T., Delvenne, J.-C., Rosvall, M., Lambiotte, R.: The many facets of community detection in complex networks. Appl. Netw. Sci. **2**, 4 (2017). https://doi.org/10.1007/s41109-017-0023-6
13. Yang, J., Leskovec, J.: Defining and Evaluating Network Communities based on Ground-truth. arXiv:physics/1205.6233 (2012)
14. Survey-community-detection by Lab41. http://lab41.github.io/survey-community-detection/. Last accessed 05 Jan 2017
15. Newman, M.: Networks: An Introduction. Oxford University Press Inc., New York, USA (2010)
16. Liu, W., Pellegrini, M., Wang, X.: Detecting communities based on network topology. Sci. Rep. **4**, (2014). https://doi.org/10.1038/srep05739
17. Radicchi, F., Castellano, C., Cecconi, F., Loreto, V., Parisi, D.: Defining and identifying communities in networks. Proc. Natl. Acad. Sci. **101**, 2658–2663 (2004). https://doi.org/10.1073/pnas.0400054101
18. Newman, M.E.J.: Finding community structure in networks using the eigenvectors of matrices. Phys. Rev. E **74**, (2006). https://doi.org/10.1103/PhysRevE.74.036104
19. Bader, D.A., Meyerhenke, H., Sanders, P., Wagner, D.: Graph Partitioning and Graph Clustering. American Mathematical Society (2013)
20. Ahajjam, S., Badir, H., El Haddad, M.: Detection of leader's nodes in complex networks. Glob. J. Eng. Sci. Res. SESA **2014**, 40–44 (2015)
21. Aggarwal, C.C.: Social network analysis. In: Data Mining, pp. 619–661. Springer International Publishing (2015)
22. Javadi, S.H.S., Gharani, P., Khadivi, S.: Detecting community structure in dynamic social networks using the concept of leadership. In: Amini, M.H., Boroojeni, K.G., Iyengar, S.S., Pardalos, P.M., Blaabjerg, F., Madni, A.M. (eds.) Sustainable Interdependent Networks: From Theory to Application, pp. 97–118. Springer International Publishing, Cham (2018). https://doi.org/10.1007/978-3-319-74412-4_7
23. Chikhi, N.F.: Calcul de centralité et identification de structures de communautés dans les graphes de documents. http://thesesups.ups-tlse.fr/1364/ (2010)
24. Girvan, M., Newman, M.E.J.: Community structure in social and biological networks. Proc. Natl. Acad. Sci. **99**, 7821–7826 (2002). https://doi.org/10.1073/pnas.122653799
25. Community structure discovery in facebook: Ferrara. E. Int. J. Soc. Netw. Min. **1**, 67–90 (2012)

26. Rhouma, D., Romdhane, L.B.: An efficient algorithm for community mining with overlap in social networks. Expert Syst. Appl. **41**, 4309–4321 (2014). https://doi.org/10.1016/j.eswa.2014.01.002
27. Shen, H., Cheng, X., Cai, K., Hu, M.-B.: Detect overlapping and hierarchical community structure in networks. Phys. Stat. Mech. Appl. **388**, 1706–1712 (2009). https://doi.org/10.1016/j.physa.2008.12.021
28. Souam, F., Aïtelhadj, A., Baba-Ali, R.: Dual modularity optimization for detecting overlapping communities in bipartite networks. Knowl. Inf. Syst. **40**, 455–488 (2014). https://doi.org/10.1007/s10115-013-0644-8
29. Márquez, R.: Overlapping community detection in static and dynamic networks. In: Proceedings of the 13th International Conference on Web Search and Data Mining, pp. 925–926. Association for Computing Machinery, New York, USA (2020). https://doi.org/10.1145/3336191.3372185
30. Costa, L.D.F., Rodrigues, F.A., Travieso, G., Villas Boas, P.R.: Characterization of complex networks: a survey of measurements. Adv. Phys. **56**, 167–242 (2007). https://doi.org/10.1080/00018730601170527
31. Shah, D., Zaman, T.: Community Detection in Networks: The Leader-Follower Algorithm. arXiv:physics/1011.0774 (2010)
32. CFinder: locating cliques and overlapping modules in biological networks. Bioinformatics, Oxford Academic. https://academic.oup.com/bioinformatics/article/22/8/1021/227298. Last accessed 15 Dec 2020
33. Fu, J., Zhang, W., Wu, J.: Identification of leader and self-organizing communities in complex networks. Sci. Rep. **7**, 704 (2017). https://doi.org/10.1038/s41598-017-00718-3
34. Lu, H., Zhao, Q., Gan, Z.: A community detection algorithm based on the similarity sequence. In: Benatallah, B., Bestavros, A., Manolopoulos, Y., Vakali, A., Zhang, Y. (eds.) Web Information Systems Engineering—WISE 2014, pp. 63–78. Springer International Publishing (2014)
35. Li, Y., Luo, P., Wu, C.: A new network node similarity measure method and its applications. arXiv:physics/1403.4303 (2014)
36. Complex networks: node similarity network formation model. http://www.network-science.org/similaritymodel.html. Last accessed 05 Jan 2017
37. Pan, Y., Li, D.-H., Liu, J.-G., Liang, J.-Z.: Detecting community structure in complex networks via node similarity. Phys. Stat. Mech. Appl. **389**, 2849–2857 (2010). https://doi.org/10.1016/j.physa.2010.03.006
38. Cheng, Q., Liu, Z., Huang, J., Cheng, G.: Discovering mesoscopic-level structural patterns on social networks: a node-similarity perspective. Appl. Math. Inf. Sci. **9**, 317–335 (2015)
39. Scholz, M.: Node similarity as a basic principle behind connectivity in complex networks. J. Data Min. Digit. Humanit. (2015)
40. Newman, M.E.J.: Modularity and community structure in networks. Proc. Natl. Acad. Sci. **103**, 8577–8582 (2006). https://doi.org/10.1073/pnas.0601602103
41. Pons, P., Latapy, M.: Computing communities in large networks using random walks (long version). arXiv:physics/0512106 (2005)
42. Rosvall, M., Axelsson, D., Bergstrom, C.T.: The map equation. Eur. Phys. J. Spec. Top. **178**, 13–23 (2009). https://doi.org/10.1140/epjst/e2010-01179-1
43. Cai, B., Wang, H., Zheng, H., Wang, H.: An improved random walk based clustering algorithm for community detection in complex networks. In: 2011 IEEE International Conference on Systems, Man, and Cybernetics, pp. 2162–2167 (2011). https://doi.org/10.1109/ICSMC.2011.6083997
44. Kernighan, B.W., Lin, S.: An efficient heuristic procedure for partitioning graphs. Bell Syst. Tech. J. **49**, 291–307 (1970). https://doi.org/10.1002/j.1538-7305.1970.tb01770.x
45. Zahoránszky, L.A., Katona, G.Y., Hári, P., Málnási-Csizmadia, A., Zweig, K.A., Zahoránszky-Köhalmi, G.: Breaking the hierarchy—a new cluster selection mechanism for hierarchical clustering methods. Algorithms Mol. Biol. AMB **4**, 12 (2009). https://doi.org/10.1186/1748-7188-4-12

46. Ganesan, P., Garcia-Molina, H.: Exploiting hierarchical domain structure to compute similarity. Widom. J. ACM Trans. Inf. Syst. **21**, 64–93 (2003). https://doi.org/10.1145/635484.635487
47. Priyadarshan, S.: Community Detection in Social Networks. http://ethesis.nitrkl.ac.in/4756/ (2013)
48. MacQueen, J.: Some methods for classification and analysis of multivariate observations. Multivar. Obs. 17
49. Jiang, Y., Jia, C., Yu, J.: An efficient community detection method based on rank centrality. Phys. Stat. Mech. Appl. **392**, 2182–2194 (2013). https://doi.org/10.1016/j.physa.2012.12.013
50. Wang, Y., Di, Z., Fan, Y.: Identifying and characterizing nodes important to community structure using the spectrum of the graph. PLoS ONE **6**, e27418 (2011). https://doi.org/10.1371/journal.pone.0027418
51. Wang, Y., Di, Z., Fan, Y.: Detecting important nodes to community structure using the spectrum of the graph. arXiv:physics/1101.1703 (2011)
52. Gaisser, T.K., Hillas, A.M.: Reliability of the method of constant intensity cuts for reconstructing the average development of vertical showers. Int. Cosm. Ray Conf. **8**, 353–357 (1977)
53. Hu, F., Liu, J., Li, L., Liang, J.: Community detection in complex networks using Node2vec with spectral clustering. Phys. Stat. Mech. Appl. **545**, 123633 (2020). https://doi.org/10.1016/j.physa.2019.123633
54. Newman, M.E.J.: Analysis of weighted networks. Phys. Rev. E. **70**, (2004). https://doi.org/10.1103/PhysRevE.70.056131
55. Blondel, V.D., Guillaume, J.-L., Lambiotte, R., Lefebvre, E.: Fast unfolding of communities in large networks. J. Stat. Mech. Theory Exp. **2008**, 10008 (2008). https://doi.org/10.1088/1742-5468/2008/10/P10008
56. Gach, O., Hao, J.-K.: A memetic algorithm for community detection in complex networks. In: Parallel Problem Solving from Nature—PPSN XII, pp. 327–336. Springer, Berlin (2012). https://doi.org/10.1007/978-3-642-32964-7_33
57. Chen, X., Li, J.: Community detection in complex networks using edge-deleting with restrictions. Phys. Stat. Mech. Appl. **519**, 181–194 (2019). https://doi.org/10.1016/j.physa.2018.12.023
58. Jiang, H., Liu, Z., Liu, C., Su, Y., Zhang, X.: Community detection in complex networks with an ambiguous structure using central node based link prediction. Knowl. Based Syst. **195**, 105626 (2020). https://doi.org/10.1016/j.knosys.2020.105626
59. Jiang, S., Wu, M.: Detecting communities and corresponding central nodes in large social networks. In: Zhang, Y., Li, K., Xiao, Z. (eds.) High Performance Computing, pp. 55–65. Springer, Berlin (2013)
60. Javadi, S.H.S., Gharani, P., Khadivi, S.: Detecting Community Structure in Dynamic Social Networks Using the Concept of Leadership, vol. 145, pp. 97–118 (2018). https://doi.org/10.1007/978-3-319-74412-4-7. arXiv:physics/1711.02053
61. Chen, Y., Zhao, P., Li, P., Zhang, K., Zhang, J.: Finding communities by their centers. Sci. Rep. **6**, 24017 (2016). https://doi.org/10.1038/srep24017
62. Lei, J., Juan, W.X., Yong, Z.: The independence of the centrality for community detection. Int. J. Mod. Phys. C **29**, 1850060 (2018). https://doi.org/10.1142/S0129183118500602
63. Rusinowska, A., Berghammer, R., Swart, H.D., Grabisch, M.: Social networks: prestige, centrality, and influence. In: de Swart, H. (ed.) Relational and Algebraic Methods in Computer Science, pp. 22–39. Springer, Berlin Heidelberg (2011)
64. Lawyer, G.: Understanding the influence of all nodes in a network. Sci. Rep. **5**, (2015). https://doi.org/10.1038/srep08665
65. de Arruda, G.F., Barbieri, A.L., Rodríguez, P.M., Rodrigues, F.A., Moreno, Y., da Costa, L.F.: Role of centrality for the identification of influential spreaders in complex networks. Phys. Rev. E **90**, 032812 (2014). https://doi.org/10.1103/PhysRevE.90.032812
66. Freeman, L.C.: Centrality in social networks conceptual clarification. Soc. Netw. **1**, 215–239 (1978). https://doi.org/10.1016/0378-8733(78)90021-7
67. Page, L., Brin, S., Motwani, R., Winograd, T.: The PageRank Citation Ranking: Bringing Order to the Web (1999)

68. Li, Q., Zhou, T., Lv, L., Chen, D.: Identifying influential spreaders by weighted LeaderRank. Phys. Stat. Mech. Appl. **404**, 47–55 (2014). https://doi.org/10.1016/j.physa.2014.02.041
69. Lö, L., Zhang, Y.-C., Yeung, C.H., Zhou, T.: Leaders in social networks, the delicious case. PLoS One **6**, e21202 (2011). https://doi.org/10.1371/journal.pone.0021202
70. Beni, H.A., Bouyer, A.: TI-SC: top-k influential nodes selection based on community detection and scoring criteria in social networks. J. Ambient. Intell. Humaniz. Comput. (2020). https://doi.org/10.1007/s12652-020-01760-2
71. Ahajjam, S., Badir, H.: Identification of influential spreaders in complex networks using HybridRank algorithm. Sci. Rep. **8**, 11932 (2018). https://doi.org/10.1038/s41598-018-30310-2
72. Ahajjam, S., Haddad, M.E., Badir, H.: LeadersRank: Towards a new approach for community detection in social networks. In: 2015 IEEE/ACS 12th International Conference of Computer Systems and Applications (AICCSA), pp. 1–8 (2015). https://doi.org/10.1109/AICCSA.2015.7507215
73. Ahajjam, S., El Haddad, M., Badir, H.: A new scalable leader-community detection approach for community detection in social networks. Soc. Netw. **54**, 41–49 (2018). https://doi.org/10.1016/j.socnet.2017.11.004
74. Khorasgani, R.R., Chen, J., Zaiane, O.R.: Top leaders community detection approach in information networks. In: Proceedings of the 4th Workshop on Social Network Mining and Analysis, 2010, p. 228 (2013). ISSN: 2319–7323
75. Wu, Q., Qi, X., Fuller, E., Zhang, C.-Q.: Follow the leader: a centrality guided clustering and its application to social network analysis. Sci. World J. **2013**, e368568 (2013). https://doi.org/10.1155/2013/368568
76. Krawczyk, M.J., Kułakowski, K., Hołyst, J.A.: Hierarchical partitions of social networks between rivaling leaders. PLoS One **13**, e0193715 (2018). https://doi.org/10.1371/journal.pone.0193715
77. Fang, C., Mu, D., Deng, Z., Hu, J., Yi, C.-H.: Fast detection of the fuzzy communities based on leader-driven algorithm. Int. J. Mod. Phys. B **32**, 1850058 (2017). https://doi.org/10.1142/S0217979218500583
78. Lu, H., Zhao, Q., Sang, X., Lu, J.: Community detection in complex networks using non-negative matrix factorization and density-based clustering algorithm. Neural Process. Lett. **51**, 1731–1748 (2020). https://doi.org/10.1007/s11063-019-10170-1
79. Sun, H., Du, H., Huang, J., Li, Y., Sun, Z., He, L., Jia, X., Zhao, Z.: Leader-aware community detection in complex networks. Knowl. Inf. Syst. **62**, 639–668 (2020). https://doi.org/10.1007/s10115-019-01362-1

Chapter 6
On the Vulnerability of Community Structure in Complex Networks

Viraj Parimi, Arindam Pal, Sushmita Ruj, Ponnurangam Kumaraguru, and Tanmoy Chakraborty

Abstract In this paper, we study the role of nodes and edges in a complex network in dictating the robustness of a community structure toward structural perturbations. Specifically, we attempt to identify all vital nodes, which, when removed, would lead to a large change in the underlying community structure of the network. This problem is critical because the community structure of a network allows us to explore deep underlying insights into how the function and topology of the network affect each other. Moreover, it even provides a way to condense large networks into smaller modules where each community acts as a meta node and aids in more straightforward network analysis. If the community structure were to be compromised by either accidental or intentional perturbations to the network that would make such analysis difficult. Since identifying such vital nodes is computationally intractable, we propose a suite of heuristics that allow to find solutions close to the optimality. To show the effectiveness of our approach, we first test these heuristics on small networks and

The work was done when Viraj was an undergraduate student at IIIT-Delhi, India.

V. Parimi (✉)
Carnegie Mellon University, Pittsburgh, USA
e-mail: vparimi@cs.cmu.edu

P. Kumaraguru · T. Chakraborty
IIIT-Delhi, New Delhi, India
e-mail: pk@iiitd.ac.in

T. Chakraborty
e-mail: tanmoy@iiitd.ac.in

A. Pal · S. Ruj
Data61, CSIRO, Sydney, Australia

A. Pal
Cyber Security CRC, Canberra, Australia

S. Ruj
Indian Statistical Institute, Kolkata, India

A. Pal
University of New South Wales, Sydney, Australia

A. Biswas et al. (eds.), *Principles of Social Networking*,
Smart Innovation, Systems and Technologies 246,
https://doi.org/10.1007/978-981-16-3398-0_6

then move to more extensive networks to show that we achieve similar results. Further analysis reveals that the proposed approaches are useful to analyze the vulnerability of communities in networks irrespective of their size and scale. Additionally, we show the performance through an extrinsic evaluation framework—we employ two tasks, i.e., link prediction and information diffusion, and show that the effect of our algorithms on these tasks is higher than the other baselines.

6.1 Introduction

A large body of research in complex networks involves the study and effects of community structure as it is one of the salient structural characteristics of real-world networks. A network is said to have a community structure if it can be grouped easily into sets of nodes. Each set of nodes is densely connected internally and sparsely linked externally. Research in this field is broadly classified into two categories—first, where one detects the community structure within a given network and the other where one studies the properties of a community structure to infer more details about the network. A variety of methods have been proposed that target the former issue as described [1, 2]. The advantage of such algorithms is that it provides us with an efficient and approximate clustering of nodes that allows us to condense large networks to smaller ones owing to their mesoscopic structure. Within the second paradigm, the ability to detect vital nodes is of significant practical importance. It provides insight into how a network functions and how the network topology change affects the interactions between the nodes within the network. Exploring this structural vulnerability of the network allows us to prepare beforehand if the network is affected by undesired perturbations and adversarial attacks. A significant factor in understanding this is to analyze the network and comprehend the effect of these vital nodes' failure on the community structure of the network.

In this paper, we attempt to identify and investigate some vital nodes in a network, whose removal highly affects the network's community structure. Formally, given a network $G(V, E)$ and a positive integer k, we intend to find a set $S \in V$ consisting of k nodes whose removal leads to the maximum damage of the community structure. The change in the community structure is quantified using different measures such as Modularity [3], Normalized Mutual Information [4], Adjusted Rand Index [5], etc.

There are many real-world applications of this problem. Consider a power grid network where a power outage is a frequently occurring event. Most power networks have a regional hub that caters to the needs of nearby power stations. In such a scenario, the vendor of this power grid needs to make quick decisions about how the failure of some nodes in the network would affect the customers. The solution would be to ensure that crucial nodes in this network have enough backup so that the restoration process can move effortlessly. Another application would be the railway networks, where inadvertent cutting of routes to certain stations can cause significant problems for the city residents. Hence, the government needs to ensure that routes to certain critical stations have redundancies so that if one route gets cut off, then the trains can utilize other routes. This problem also has applications in Online Social Networks such as the worm containment problem [6]. This knowledge would provide helpful insights into protecting sensitive nodes once worms spread out into the network. In all of the issues mentioned above, it is evident that one needs to study the structural integrity of the communities underlying in the network. Note that a minor structural change that can be as small as removing a node in the network can lead to the community's breakdown that the node was a part of, given that the removed node had a considerable influence on the network. If the removed node were of less significance that would have less impact on the network's community structure.

Additionally, understanding the network vulnerability from the standpoint of the community structure is essential in real-world settings. The networks that are dealt with here have tremendous size, which adds to the computational overhead and, most importantly, shed light on some latent characteristics shared by different nodes. Since communities can act as meta-nodes, they allow for a more comfortable study of large networks. This reduces the computational overhead and provides useful insights based on the properties shared by the community's nodes that can be exploited to understand the network's vulnerability.

We propose a hierarchical greedy approach that selects communities based on the community-centric properties in phase 1 and then, within that community, selects the most vulnerable nodes in phase 2. We test this algorithm on six real-world datasets of varying sizes. Our empirical results indicate that the algorithm can identify properties that contribute most toward community structures' vulnerabilities in a network. The past work in this domain [7, 8] is restricted to smaller networks, but our work extends the scope toward even more extensive networks with the number of nodes in the order of millions.

In summary, our contributions in this paper are as follows:

– We study the structural vulnerability of communities in networks and assess the impact of nodes' failure on the underlying community structure.
– We suggest few heuristics, including a hierarchical greedy approach that allows for identifying such critical nodes in the network that profoundly impact the community structure.
– We conduct experiments on real-world datasets and show the effectiveness of the heuristics that we propose.
– We propose a novel task-based strategy to evaluate the extent of correctness of the algorithm extrinsically. This allows us to estimate the performance of our algorithm in a real-world context.

The remaining part of this paper is as follows. We discuss the literature review on community detection and vulnerability assessment in Sect. 6.2. We formalize our problem in Sect. 6.3. We then discuss some preliminaries in Sect. 6.4. We present our proposed methodology in Sect. 6.5. Section 6.6 describes the datasets used to evaluate the proposed approach. In Sect. 6.7 we provide the results of our method when applied to these datasets and briefly discuss the evaluation strategy how we go about validating our proposed method on larger datasets. We put forward our conclusion in Sect. 6.8.

6.2 Related Work

This section first presents the literature on community detection algorithms and then discusses community vulnerability analysis.

6.2.1 Community Detection

Community detection, a task of grouping similar nodes together, is a significant problem. A lot of work has been done in the past to come up with a solution effectively. Numerous approaches have been developed and applied to detect community structure. For instance, a hierarchical agglomerative algorithm was proposed by Newman et al. [9]. An extensive literature survey can be found in [2]. Here we briefly mention some of the popular approaches.

Initial efforts at community detection assumed that the nodes are densely connected within a community and sparsely connected across communities. Under this assumption, the algorithms proposed were targeted toward community detection in static networks. Such efforts involved several approaches such as modularity optimization [3, 10–13], clique percolation [14, 15], information-theoretic approaches [16, 17], and label propagation [18–20]. Furthermore, spectral partitioning [21, 22], local expansion [23, 24], random-walk based approaches [25, 26], diffusion-based approaches [18] and significance-based approaches [27] were explored to help in identifying the community instances within a network. Several pre-processing methods [28, 29] were also introduced to improve upon these algorithms. Such methods involved generating a preliminary community structure on a set of nodes and modifying iteratively until all the nodes are covered. Apart from these, several other algorithms were proposed to detect communities in dynamically evolving networks [30, 31].

Another set of community detection algorithms allows a vertex to be part of multiple communities simultaneously. Such overlapping community detection algorithms used ideas based on local expansion and optimization. These include RankRemoval [32] which uses local density function, LFM [24], and MONC [33] which iteratively maximize a fitness function, and GCE [34] which makes use of an agglomerative pipeline to detect overlapping community instances. Other approaches also looked into the idea of partitioning links instead of nodes to discover the network's underlying community structure. The clique percolation method was also explored in CFinder [35], but since many real-world networks are sparse, these methods generally produced low-quality outputs. Recently, several new ideas were presented, such as [36] which solved a constrained optimization problem using simulated annealing techniques, and [37–40] which used mixture models to solve the problem. Even a game-theoretic approach [41] was proposed in which a community is equated to a Nash local equilibrium. Non-negative Matrix Factorization [42, 43] framework has also been utilized to identify fuzzy or overlapping community structures. Chakraborty et al. proposed MaxPerm and GenPerm, two greedy approaches which maximize a node-centric metric, called "permanence" to detect disjoint [9] and overlapping communities [44]. They also proposed a post-processing technique based on permanence to detect overlaps from a disjoint community structure [45]. Several ensemble-based approaches were also proposed by leveraging the output of disjoint community detection methods [46–48].

6.2.2 Community Vulnerability Analysis

Assessing the structural network vulnerability has received increasing attention. For example, Nguyen et al. [49] have proposed a Community Vulnerability Assessment (CVA) problem and suggested multiple heuristic-based algorithms based on the modularity measure of communities in the network. These approaches are restricted to the scope of online social networks and do not cater to general network structures. Another work by Nguyen et al. [50] explored the number of *connected triplets* in a network as they capture the strong connection of communities in social networks. They proposed an efficient approximation algorithm to identify triangle breaking points like nodes or links within a network.

Additionally, different measures and metrics have been proposed to measure the robustness of a network. Such efforts include the average size of a cluster, relative size of the largest components, diameter, and network connectivity. One approach dealt with this problem using the weighted count of loops in a network. Chan et al. [51] addressed this problem in both deterministic and probabilistic settings where they suggested solutions based on minimum node cutset. Frank et al. [52] outlined a solution that uses the second smallest eigenvalue of a Laplacian matrix of a network and termed it as the algebraic connectivity of that network. Fiedler [53] proposed four basic attack strategies, namely, ID removal, IB removal, RD removal, and RB removal. ID and RD removal deal with the degree distribution of the network. The only difference is that the second approach changes the removal strategy based on the degree distribution change. IB and RB removal are also similar constructs, but they are based on the betweenness distribution. Holme et al. [54] used an algorithm adapted from Google's PageRank providing a sequence of losses that add to the collapse of the network. Allesina et al. [55] evaluated the network characteristics like cyclomatic number and gamma index. They mentioned that such global graph-theoretic indices are not sufficient to measure a network's vulnerability, but they showcase the hierarchy of nodes in the system.

Ramirez et al. [56] proposed an approach where the community structure's resilience is quantified by introducing disruption in the original network and measuring the change in the community structure temporally, i.e., after the disconnection and during the restoration process. Geubesic et al. [57] provided a review of various approaches that use the *facility importance* concept to understand the system-wide vulnerability. These concepts include alpha index, beta index, etc. They concluded that simple graph-theoretic measures were not sufficient to measure the vulnerability of a network. It also required many local efforts, such as the degree of node. They mentioned that global indicators measure network accessibility, path availability, and local measures to provide better information about node criticality. Sankaran et al. [8] proposed a new vulnerability metric where they considered a combination of external and internal factors such as connection density. They proposed a non-linear weighted function to combine these factors. However, the proposed method was not proved feasible in practice as the weights of all the elements were assumed to be equal and not self-adjusting to the network. These methods allow us to quantify a community's vulnerability but do not provide us with a set of nodes that contribute to the community's vulnerability.

The information of critical nodes that contribute to the communities' vulnerability would provide far more insights than just discovering the vulnerable community. As a result, a more comprehensive study is required to assess the vulnerability of general network structures.

6.3 Problem Statement

Let $G(V, E)$ be an input graph and let k be the number of nodes that we want to select. Let $\mathcal{A}$ be a community detection algorithm. For a vertex set $S \in V$, let $G[S]$ be the subnetwork induced by S and $f(\mathcal{A}(G[V]), \mathcal{A}(G[V \setminus S]))$ is a value function that computes some measure of the difference between the community structures of $G[V]$ and $G[V \setminus S]$ obtained from $\mathcal{A}$. We need to identify a set $S \in V$ of size k which

$$\textbf{maximize} \quad f(\mathcal{A}(G[V]), \mathcal{A}(G[V \setminus S])) \tag{6.1}$$

This problem is computationally intractable as shown by Alim et al. [49] and hence requires the use of greedy heuristics to approach the optimal answer.

6.4 Preliminaries

We used the Louvain algorithm for detecting the underlying community structure. It is a greedy optimization algorithm proposed by Blondel et al. [10] that tries to optimize the modularity metric of a network and extracts communities from large networks using heuristics. This approach, however, can easily be modified to use with other community detection algorithms as well.

To quantify the difference between the community structures of $G[V]$ and $G[V \setminus S]$, we use the following measures:

- **Modularity:** It is a measure to quantify the strength of the division of the network into communities. Networks with high modularity have denser connections within a community and sparse connections across communities. It is defined as follows

$$Q = \frac{1}{(2m)} \sum_{vw} \left[A_{vw} - \frac{k_v k_w}{(2m)} \right] \delta(c_v, c_w), \tag{6.2}$$

where m = number of edges, A = adjacency matrix, k_v = degree of node v, c_v = community label of node v, $\delta(c_v, c_w)$ = 1 if $c_i = c_j$ and 0 otherwise.

- **Normalized Mutual Information:** It is a measure that quantifies the similarity between two community structures. It produces 1 if two community structures are exactly the same and 0 otherwise. It is defined as follows

$$N = \frac{2 \sum_{i=1}^{c_X} \sum_{j=1}^{c_Y} \left[n_{ij} \log\left(\frac{n_{ij} n}{x_i y_j}\right) \right]}{(n-k)H(X) + \bar{y} \log(n-k) - \sum_{j=1}^{c_Y} (y_j \log(y_j))}, \tag{6.3}$$

where c_X = number of communities in community structure X, c_Y = number of communities in community structure Y, $n_{ij} = \left| X_i \cap Y_j \right|$, n = number of nodes in the network, $x_i = |X_i|$, $y_i = |Y_i|$, $\bar{y}$ = total size of communities in Y, $H(X)$ = entropy of X.

- **Adjusted Rand Index:** It is another measure of similarity between two data clusterings. It represents the frequency of occurrence of agreements over the total pairs. Its maximum value is 1 which indicates perfect similarity between two clusterings. It is defined as follows

$$R = \frac{\sum_{ij} \binom{n_{ij}}{2} - \left[\sum_i \binom{a_i}{2} \sum_j \binom{b_j}{2} \right] / \binom{n}{2}}{\frac{1}{2} \left[\sum_i \binom{a_i}{2} + \sum_j \binom{b_j}{2} \right] - \left[\sum_i \binom{a_i}{2} \sum_j \binom{b_j}{2} \right] / \binom{n}{2}} \tag{6.4}$$

where L = contingency Table, $n_{ij} = L[i][j]$, a_i = sum of entries in ith row in L, b_i = Sum of entries in ith column in L, n = number of nodes in the network.

6.5 Proposed Methodology

Given the computation intractability of the problem statement, we first chunk our approach into two sections. We analyze the structural properties of a small network and generate ground truth data. This data provides us a way to compare our proposed heuristics, thereby quantifying the effectiveness of these heuristics.

Algorithm 1: Exhaustive Algorithm

Input : Network $G = (V, E)$, k, a community detection algorithm $\mathcal{A}$, a value function F
Output: Set of nodes whose size is k

$X \leftarrow$ Run community detection algorithm $\mathcal{A}$ on G.
$C \leftarrow$ Generate all the combination of nodes in V of size k.
foreach $C_i \in C$ **do**
    $G' \leftarrow$ Remove C_i from G.
    $Y \leftarrow$ Run community detection algorithm $\mathcal{A}$ on G'.
    Compute F by comparing Y and X.
    Return a C_i which maximizes F.
end

The thorough approach to gather this information is described in Algorithm 1. This approach compares the networks' community structures before and after structural perturbations, where similarity scores for each combination of nodes are computed.

Yang et al. provide a comparative analysis of significant community detection algorithms, including Edge Betweenness, Fastgreedy, and Infomap. In Algorithm 1, we generate all possible combinations of nodes of size k and then analyze the effect of each such combination to see which minimized the target value function more. Let's consider the computational complexity of the community detection algorithm to be D. This means that the complexity of this algorithm is $O(\max(C(n, k), D))$ where $n = |V|$ and k is the budget. Note that D is generally defined in terms of $|V|$ and $|E|$, so this term becomes more dominant for larger networks, thereby increasing the algorithm's computational complexity.

Algorithm 2: Network-Based Greedy Approach

Input : Network $G = (V, E)$, k, a structural property to rank the nodes P, a community detection algorithm $\mathcal{A}$, a value function F
Output: Set of nodes whose size is k, score

$X \leftarrow$ Run community detection algorithm $\mathcal{A}$ on G.
$R \leftarrow$ Rank the nodes in G based on the structural property P.
$G' \leftarrow$ Remove top k nodes from G based on R.
$Y \leftarrow$ Run community detection algorithm $\mathcal{A}$ on G'.
Compute the value function F by comparing X and Y.
Return the set of top k nodes in R along with the score of the value function F.

Next, we propose a naive network-based greedy approach defined in Algorithm 2. This algorithm takes in a property as an input and ranks the nodes in the input network based on the property specified. It greedily removes the top k nodes based on their ranks and then evaluates the underlying community structure using a community detection algorithm. The output of this algorithm computes the value function and returns the set of nodes removed along with the evaluated value function score. The structural properties which were used are as follows

- **Clustering Coefficient**—We use the global coefficient, which is defined as the number of closed triplets over the total number of triplets where a triplet is a set of three nodes that are connected by either two or three undirected edges. The complexity of calculating this property for a node is $O(|V|^3)$.
- **Degree Centrality**—It is defined as the number of edges that are incident upon a node. The time complexity of this metric is $O(|V| + |E|)$.
- **Betweenness Centrality**—We use the betweenness centrality estimate defined by Freeman. [58] as the number of times a node acts as a bridge along a shortest path route between two other nodes. Time complexity of this metric is $O(|V||E| + |V|^2)$.
- **Eigenvector Centrality**—It is a measure of the influence of a particular node in the network [59]. This centrality estimate is based on the intuition that a node is more central when there are more connections within its local network. The time complexity of calculating this metric for a node is $O(|V|^3)$.
- **Closeness Centrality**—It measures how easily other vertices can be reached from a particular vertex [60, 61]. Time complexity of this metric is $O(|V||E| + |V|^2)$.
- **Coreness**—The coreness of a node is k if it is a member of a k-core but not a member of a $k+1$-core where a k-core is a maximal subnetwork in which each vertex has at least degree k [62]. The time complexity of this metric is $O(|E|)$.
- **Diversity**—The diversity index of a vertex is estimated by the normalized Shannon entropy of the weights of the edges incident on a vertex [63]. The time complexity of calculating this metric is $O(|V| + |E|)$.
- **Eccentricity**—It is defined as the shortest maximum distance from the vertex to all the other vertices in a network. The time complexity of this metric is $O(|V|^2 + |V||E|)$.
- **Constraint**—Introduced by Burt [64], this measure estimates the time and energy that are concentrated on a single cluster. This measure would be higher for a node that belongs to a small network, and also, all the contacts are highly connected. The time complexity of calculating this metric is $O(|V| + |E| + |V|d^2)$ where d is the average degree.
- **Closeness Vitality**—It is defined as the change in the distance between all node pairs when the node in focus is removed. It is based on the Wiener Index, which is defined as the sum of distances between all node pairs [65]. The metric's time complexity is $O(|E| \log |V|)$.

This algorithm takes in a community detection algorithm and a structural property of a node as inputs. If the computational complexity of the community detection algorithm is D and for calculating the structural property for all nodes is S as discussed above, then the total computational complexity of Algorithm 2 is $O(\max(S, D))$.

The downside of this algorithm is that it does not consider the nodes' effect on the community structure of the networks itself. This is addressed in Algorithm 3, which also considers the underlying community structure. Here, we propose a hierarchical approach where we choose a community based on some community-centric metric in the first phase. Then in the second phase, we select a node greedily based on its node-centric properties. The community-centric properties used are as follows:

– **Link density**: $D(G) = \frac{2E}{V(V-1)}$, where E is the number of edges in the network and V is the number of vertices in the network.
– **Conductance**: Given a graph $G(V, E)$, $\lambda(G) = \frac{s}{v}$, where s is number of vertices with one endpoint in G and another in $\bar{G}$, v is the sum of degree of nodes in G. This measure calculates how well-knit a graph is.
– **Compactness**: $C(G)$ is defined as the average shortest path lengths within the network G.

This algorithm takes a community detection algorithm, a node's structural property, and a community-centric property as inputs. The computational complexity of calculating the community-centric property will be constant in time as they are defined in terms of fixed formulas; hence, this would not contribute much to this algorithm's overall complexity. If the computational complexity of the community detection algorithm is D and for calculating the structural property for all nodes is S as discussed above, then the total computational complexity of Algorithm 2 is $O(\max(S, D))$.

Algorithm 3: Community-Based Greedy Approach

Input : Network $G = (V, E)$, k, a global community-centric property P_c, a node-centric property P_n, a community detection algorithm $\mathcal{A}$, a value function F
Output: Set of nodes whose size if k, score

Function `best_community(`*network G, community structure X*`)`**:**
  foreach $X_i \in X$ **do**
    $G' \leftarrow$ Create a subnetwork from G with only the vertices from X_i.
    $R_g \leftarrow$ Rank each such G' based on the community-centric property P_c.
    Return X_i whose induced subnetwork G' ranked above others based on R_g.
  end
Function `best_node(`*network G, community structure X*`)`**:**
  $G' \leftarrow$ Create the subnetwork G' from G which is induced from X.
  $R_n \leftarrow$ Rank the nodes in G' based on the node-centric property P_n.
  Return the top node that ranks above others based on R_n.
$X \leftarrow$ Run community detection algorithm $\mathcal{A}$ on G
$Y \leftarrow$ Run community detection algorithm $\mathcal{A}$ on G
while *k nodes are not selected* **do**
  $X' \leftarrow$ best_community(G, Y)
  $node \leftarrow$ best_node(G, X')
  $G' \leftarrow$ Remove *node* from G and add this node to the output set
  $Y \leftarrow$ Run community detection algorithm $\mathcal{A}$ on G'
end
Compute the value function F by comparing Y and X.
Return the output set of nodes and the score evaluated by the value function F.

The algorithms proposed above are sufficient for smaller networks. We can evaluate them with Algorithm 1; however, real-world networks exhibit a much more extensive and complex structure.

The reason is the inefficiency of Algorithm 1 as it is a brute force method. This won't allow for the extraction of the ground truth, which we use to estimate the performance of Algorithms 2 and 3. To counter this, we propose a new task-based approach. Here, the intuition is as follows: if the performance of an extrinsic task, based on the network structure is ϕ, then after removing the nodes based on the outputs of Algorithms 2 and 3, the task would perform $\chi \leq \phi$ on the new network structure, thereby validating the selection of nodes.

Specifically, suppose that a user wants to select vulnerable nodes in an extensive network such that the resulting value function score is maximized. To do so, a straightforward way is to use Algorithms 2 and 3 to select the nodes whose effectiveness can be validated by the results of Algorithm 1. But since the network is extensive, it is quite evident that it is not feasible to use Algorithm 1. To counter this, one would use Algorithm 4 to validate the results based on the network's performance drop on the tasks. Since we are using the same algorithms used for small networks, it is evident that the actual problem at hand of maximizing the value function is still of prime focus. Only the way to validate those same results has been changed for more extensive networks.

Algorithm 4: Task-Based Approach

Input : Network $G = (V, E)$, k, a task T, a community detection algorithm $\mathcal{A}$, a value function F

Output: Set of nodes whose size if k, score

Function `compute_task_performance` (*Task* T*, network* G_1*, network* G_2*, community structure of* G_1 X*, community structure of* G_2 Y) **:**

    if T *is link prediction* **then**

        Create a test and train edge list based on the edge set of G_1.

        $G'_1 \leftarrow$ Create a subnetwork induced by the training set

        Apply the link prediction task using X to decide on the edge probabilities on G'_1

        Compute the $F1$ score for the predicted edges

        Repeat the same process for network G_2

        Compare the $F1$ scores for both the networks

    end

    else

        Select a random set of seed nodes that are active by default

        With $p_i = 0.7$ and $p_o = 0.3$ apply the information diffusion task on G_1 using the independent cascade model for 200 iterations. This will give the number of active nodes at the end of the iterations

        Repeat the process with G_2

        Compare the number of active nodes at the end for both G_1 and G_2

    end

$X \leftarrow$ Run community detection algorithm $\mathcal{A}$ on G

$S \leftarrow$ Output from Algorithm 2 or Algorithm 3 which return the target set of nodes

$G' \leftarrow$ Remove nodes in S from G

$Y \leftarrow$ Run community detection algorithm $\mathcal{A}$ on G'

$score \leftarrow$ compute_task_performance(T, G, G', X, Y)

In Algorithm 4, we consider two different tasks, which are described as follows:

1. **Link Prediction:** We predict the likelihood of a future association between two nodes knowing that there is no association between those nodes in the current state of the network. Hence, the problem asks to what extent the evolution of a complex network can be modeled using features intrinsic to the network topology itself. Generally, in literature, people use few metrics to assign probabilities to a set of non-edges in a network such as Within-Inter-Cluster defined by Rebaza et al. [66], Modified Common Neighbors and Modified Resource Allocation defined by Soundarajan and Hopcroft [67].
2. **Information Diffusion**: It is defined as the process by which a piece of information is spread and reaches individuals through interactions. We empirically study the behavioral characteristics of information diffusion models, specifically IC (Independent Cascade), on different community structures. We incorporate the community information in this task by assigning p_i probability to edges inside a community and p_o probability to edges that connect separate communities. We keep $p_i \geq p_o$ as information is more likely to spread among nodes within the same neighborhood as observed by Lin et al. [68].

6.6 Datasets

To run our experiments extensively, we select six real-world networks of diverse sizes. The datasets used are as follows:

1. **Karate Club:** The data was collected from the members of a karate club [69, 70]. Each node represents a club member, and each undirected edge represents a tie between two members of the club. The network has two communities, one formed by "John A" and another by "Mr Hi".
2. **Football Network:** Girvan and Newman [9] collected this network. It contains American football games between division IA colleges during the regular-season Fall of 2000. The nodes represent teams identified by names, and edges represent regular-season games between two teams that they connect. The network has twelve communities where each community is signified by the conferences that each college belongs to.
3. **Indian Railway Network:** This network was used in [71], which consists of nodes that represent stations where two stations are connected by an edge if there exists at least one train route between them such that these stations are scheduled halts. The states act as communities, and hence there are 21 communities.
4. **Co-authorship Network:** This network was collected by Chakraborty et al. [71]. This dataset comprises nodes representing an author, and an undirected edge between two authors is drawn if and only if they were co-authors at least once. Each author is tagged with one research field on which he/she has written most papers on. There are 24 such fields, and they act as communities.

Table 6.1 Properties of the real-world networks used in our experiments. We chose 3 small and 3 large networks to extensively show the effects of each algorithm in terms of efficiently computing the vulnerable communities

Dataset	#Nodes	#Edges	#Communities
Karate Club Network	34	78	2
Football Network	115	613	12
Indian Railway Network	301	1,224	21
Co-authorship Network	103,667	352,183	24
Amazon Product Co-purchasing Network	334,863	925,872	75,149
Live Journal Network	3,997,962	34,681,189	287,512

5. **Amazon Product Co-purchasing Network:** This was collected by crawling the Amazon site [72]. The nodes represent products, and an undirected edge between two nodes represents a frequently co-purchased product. There are 75,149 communities, and only groups containing more than three users are considered.
6. **Live Journal:** This is a free online blogging community where users declare friendship with each other [72]. Therefore, each node is a user, and an edge between two users represents a friendship. Users are allowed to form groups, and such user-defined groups form communities. There are 287,512 communities, and only groups containing more than three users are considered (Table 6.1).

6.7 Experiments

We divide this section into three subsections to cover all the value functions discussed in Sect. 6.4. We first present the results of Algorithm 1 for smaller networks, which will be used as a benchmark to compare the results of Algorithms 2 and 3 whose results will follow. Using the inferences from these results, we build on our argument and present the results of Algorithm 4 to establish similar results even on more extensive networks.

6.7.1 Modularity

Exhaustive Approach: Table 6.2 shows the results of Algorithm 1 on three small networks when using the modularity as the target value function. We perform the analysis by fixing $k = 5$.[1] For the Karate network, we observe that nodes (0, 1, 3, 5,

[1] We choose the value of k to be five because of the following reason. Since we intend to compare our approach with the ground truth data, we first need to generate this ground truth data. For smaller k values, the number of nodes' combinations is more diminutive and keeps increasing exponentially

Table 6.2 Effect of the exhaustive algorithm on the small networks. The nodes here indicate the ID of the most vulnerable points in the network when the modularity is utilized as the value function

Network	Nodes	Modularity
Karate	(0, 1, 3, 5, 6)	0.13436
Football	(23, 33, 24, 32, 45)	0.10492
Railway	(105, 76, 203, 123, 97)	0.14723

Since the networks are smaller in size, the budget k is fixed at 5 which is why the algorithm detects only 5 vulnerable nodes. The corresponding modularity score reported is the maximum across all possible combinations of the nodes

6) are the most vulnerable as their removal maximizes the difference of modularity scores between the original and the perturbed networks. Similarly, the most susceptible nodes identified for the other two networks are mentioned in Table 6.2.

Network-Based Greedy Approach: This section presents the analysis results on all the datasets of Algorithm 2. We performed this analysis on all the datasets irrespective of their scale as the algorithm applied was greedy and did not need much time to execute. Moreover, we fix $k = 5$ for smaller networks, but such a removal strategy won't showcase significant effects for more extensive networks. This is because removing just five nodes in more extensive networks won't affect the underlying community structure enough to cause substantial structural perturbations. So, to handle such cases, we instead remove 5% of the total nodes. From Fig. 6.1, we infer that the clustering coefficient as a network-based greedy metric performs better than other greedy metrics when we remove the target five nodes.

Moreover, when we compare the maximum values attained in the smaller networks, we see that this algorithm cannot achieve the optimal answer indicated in Table 6.2. For example, in the Karate network, the maximum score obtained by Algorithm 2 is around 0.05, whereas the optimal answer is 0.13. This indicates that there is a lot of scope for improvement.

Community-Based Greedy Approach: In this section, we evaluate the performance of Algorithm 3 over all the datasets. As mentioned previously, we fix $k = 5$ for smaller networks and 5% for more extensive networks. We compare the different community-centric properties in Table 6.3. Here we present the best modularity scores obtained after applying this algorithm on all the datasets. As this algorithm is also inherently greedy, it also is computationally efficient. Based on Table 6.3, we observe that Link Density performed better than the other community-centric properties as the scores over all the datasets were maximum. Now that we have established that the best community-centric property in a modularity difference maximization setting is link density, we present the node-centric properties' results in Fig. 6.2. Overall, we ran

as we increase the value of k. To limit the computational time, we restricted k to be five, and beyond that, the number of combination of nodes was too large. Simultaneously, we did not want to choose a smaller k, as removing a smaller number of nodes would not have that much impact on the underlying community structure than removing more nodes.

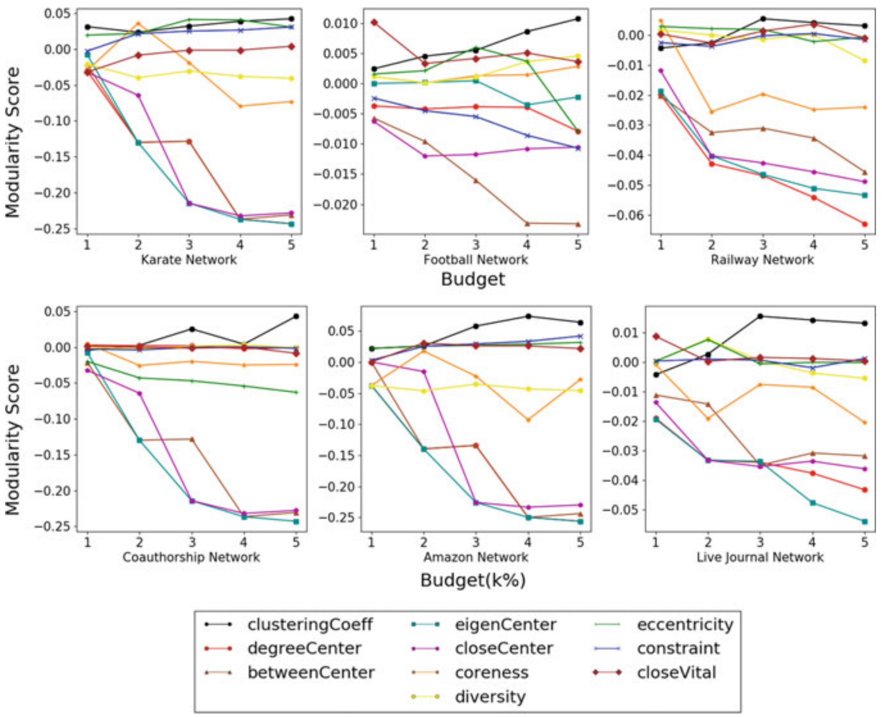


Fig. 6.1 Outcome of the network-based approach over all the networks with modularity being the target value function. The legend indicates all the structural properties of a network that were used to greedily select nodes. For smaller networks we used $k = 5$ nodes whereas for larger networks we used 5% of the nodes in the corresponding network. If we compare the smaller datasets' results with Table 6.2, we observe that the maximum values obtained could not attain the optimal values. Across the networks, for higher budget we observe that clustering coefficient turned out to be the best indicator for vulnerability in terms of modularity

experiments on the datasets; we found that eigenvector centrality performs better than other greedy metrics.

Additionally, when we compare this algorithm's results with the ground truth data presented in Table 6.2, we observe that this solution comes close to the optimal solution. For example, in the Railway network, we follow that the best modularity score obtained to be around 0.06, which is close to the ground truth score of 0.14 compared to the 0.01 score obtained from Algorithm 1. So it is evident from this data that the difference between the optimal solution and the current solution has decreased, thereby establishing the superiority of Algorithm 3 over 2.

Table 6.3 Outcome of the community-based approach using modularity as the target value function. It shows the effects of different community-based metrics have when used to greedily select nodes

Network	Link Density	Conductance	Compactness
Karate	0.04194	0.00116	0.02219
Football	0.04202	0.02193	0.00490
Railway	0.06422	0.03174	0.03749
Co-authorship	0.13037	0.03609	0.00285
Amazon	0.13052	0.00550	0.01783
Live Journal	0.05289	0.00016	0.03749

Based on this table we observe that Link Density performs better to indicate the vulnerability of nodes in terms of difference between modularity of the resultant and the original network across all the datasets

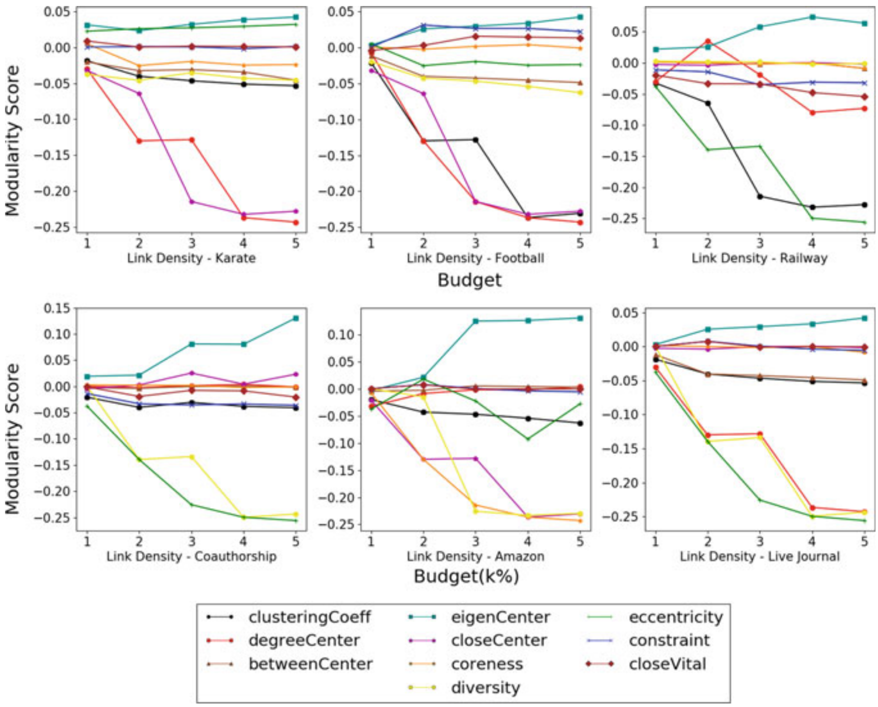


Fig. 6.2 Results of the community-based approach over several datasets with modularity being used as the target value function. These results are reported only for Link Density as it outperformed the other community-based greedy metrics as described in Table 6.3. The plots indicate that across the datasets for larger budgets, eigenvector centrality performs better in comparison to other greedy metrics. The values indicated are close to the ground truth as reported in Table 6.1

6.7.2 Normalized Mutual Information

Exhaustive Approach: Table 6.4 presents the results of Algorithm 1 on three small scale datasets where the value function that we are trying to minimize is NMI. Note that we would want to minimize NMI as this metric gives a value of 1 for two similar community structures and 0 otherwise as mentioned in Sect. 6.4. For this experiment, we fix the number of target nodes, i.e., $k = 5$. For the football network, we observe that nodes (32, 33, 5, 6, 1) are identified as the most vulnerable as they minimize the NMI score between the original and the structurally perturbed one to 0.38. This value represents the ground truth as no other combination of the five-tuple nodes will further decrease the NMI score between the two partitions. Similarly, the other small datasets' ground truth values can be found in Table 6.4.

Network-Based Greedy Approach: This section presents the analysis results on all the datasets of Algorithm 2. Moreover, we fix $k = 5$ for smaller networks, as mentioned previously. Still, for more extensive networks, such removal strategy won't showcase significant effects, and hence we remove till 5% of the total nodes in such cases. Based on Fig. 6.3, we infer that eccentricity as a network-based greedy metric performs better than other greedy metrics when we remove the target five nodes. As we evaluate the NMI measure, we compare the minimum values attained in the ground truth data to the minimum values obtained with Algorithm 2. This is because NMI's value is small when two clusterings are not the same as mentioned previously in Sect. 6.4. Based on this comparison for smaller networks, we see that this algorithm could not attain the optimal answer indicated by Table 6.4. For example, in the Karate network, the minimum score obtained by Algorithm 2 is 0.55, whereas the optimal answer is 0.36. This indicates that there is a lot of scope for improvement.

Community-Based Greedy Approach: In this section, we evaluate the performance of Algorithm 3 over all the datasets. As mentioned previously, we fix $k = 5$ for smaller networks and 5% for more extensive networks. We compare the different community-centric properties in Table 6.5. Here we present the best NMI scores obtained after applying this algorithm on all the datasets. Based on Table 6.5, we observe that Link Density performed better than the other community-centric properties as the scores over all the datasets were minimal. With link density as the best community-

Table 6.4 Effect of the exhaustive algorithm on the smaller networks. The nodes here indicate the ID of the most vulnerable points in the network when NMI is utilized as the value function

Network	Nodes	NMI
Karate	(33, 10, 32, 6, 23)	0.36762
Football	(32, 33, 5, 6, 1)	0.38580
Railway	(51, 143, 2, 89, 287)	0.38723

Since the networks are smaller in size the budget k was fixed at 5 which is why there the algorithm detected 5 vulnerable nodes. The corresponding NMI score reported was the minimum across all possible combinations of the nodes

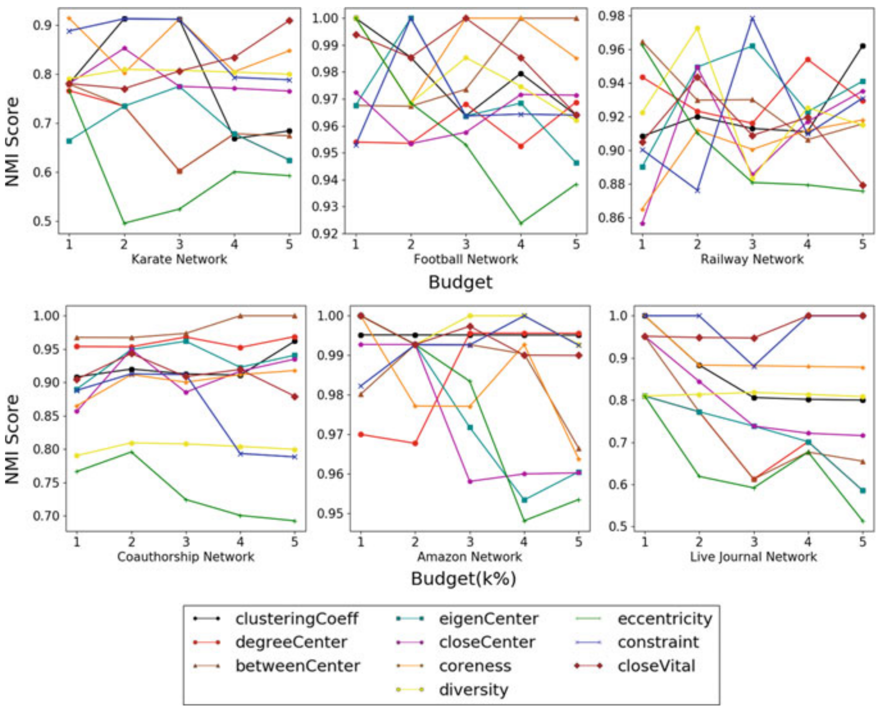


Fig. 6.3 Results of the network-based approach over all the datasets with NMI being the target value function. For smaller networks we use $k = 5$ and for larger ones we use 5% of the total nodes within the network. The plots indicate that for larger budgets, eccentricity performs well in identifying vulnerable nodes when the vulnerability of a community is quantified using NMI where lower values are better indicators of disjointness. However, upon closer inspection one can observe that these values when compared to the ground truth values reported in Table 6.4 are still pretty far from optimal

Table 6.5 Results of the community-based approach using NMI as the target value function. It shows the effects of different community-based metrics that are used to greedily select nodes

Network	Link Density	Conductance	Compactness
Karate	0.62484	0.68425	0.79993
Football	0.75558	0.96877	0.91794
Railway	0.51372	0.80825	0.62484
Co-authorship	0.59279	0.71568	0.79993
Amazon	0.58566	0.76560	0.65510
Live Journal	0.58566	0.62484	0.78850

We observe that Link Density performs better to indicate the vulnerability of nodes in terms of the NMI between the resultant and the original network across all the datasets

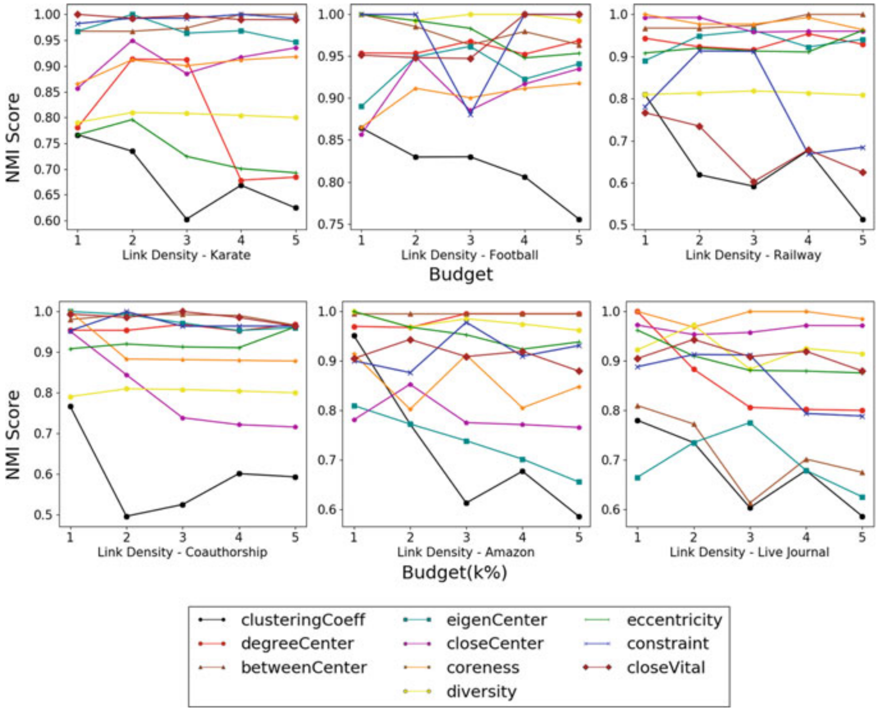


Fig. 6.4 Outcome of the community-based approach over all the datasets with NMI being the target value function. Based on Table 6.5 we observed that Link Density performs better in comparison to other greedy metrics. The plots reported in this figure present the results of the node-centric properties with Link Density as the community-centric method. They indicate that across all the datasets clustering coefficient performed better compared to other greedy metrics

centric method, we present the node-centric properties' results in Fig. 6.4. Overall the datasets we ran experiments on, we found that the clustering coefficient performs better than other greedy metrics.

Additionally, when we compare this algorithm's results with the ground truth data presented in Table 6.4, we observe that this solution comes close to the optimal solution. For example, in the Railway network, we follow that the best NMI score obtained to be around 0.5 is close to the ground truth score of 0.38 compared to the 0.88 score obtained from Algorithm 1. So it is evident from this data that the difference between the optimal solution and the current solution has decreased, thereby establishing the superiority of Algorithm 3 over 2.

6.7.3 Adjusted Rand Index

Exhaustive Approach: Table 6.6 shows the results of Algorithm 1 on three small scale datasets when using the ARI as the target value function. We performed the analysis by fixing $k = 5$. We observe that nodes (61, 85, 16, 99, 7) are the most vulnerable for the football network as their removal minimized the ARI scores between the original and the perturbed network's vertex clusterings. Similarly, the most susceptible nodes identified for the other two datasets have been tabulated in Table 6.6.

Network-Based Greedy Approach: This section presents the analysis results on all the datasets of Algorithm 2. We fix $k = 5$ for smaller networks as mentioned previously, but for more extensive networks, we remove till 5% of the total nodes. Based on Fig. 6.5, we infer that closeness vitality as a network-based greedy metric performs better than other greedy metrics when we remove the target five nodes. As we evaluate the ARI measure, we compare the minimum values attained in the ground truth data to the minimum values obtained with Algorithm 2. This is because ARI's value is small when two clusterings do not agree with each other, as mentioned previously in Sect. 6.4. Based on this comparison for smaller networks, we see that this algorithm cannot attain the optimal answer mentioned in Table 6.6. For example, in the Railway network, the minimum score obtained by Algorithm 2 is 0.65, whereas the optimal answer is -0.28. This indicates that there is a lot of scope for improvement.

Community-Based Greedy Approach: In this section, we evaluate the performance of Algorithm 3 over all the datasets. As mentioned previously, we fix $k = 5$ for smaller networks and 5% for larger networks. We compare different community-centric properties in Table 6.7. Here we present the best ARI scores obtained after applying this algorithm on all the datasets. We observe that conductance performs better than the other community-centric properties as the scores over all the datasets are minimum. With conductance as the best community-centric method, we present the node-centric properties' results in Fig. 6.6. Overall the datasets we run experiments on, we find that coreness performs better compared to other metrics.

Additionally, when we compare this algorithm's results with the ground truth data presented in Table 6.6, we observe that this solution comes close to the optimal

Table 6.6 Effect of the exhaustive algorithm on the smaller networks. The nodes here indicate the ID of the most vulnerable points in the network when ARI is utilized as the value function

Network	Nodes	ARI
Karate	(32, 7, 12, 18, 2)	−0.46342
Football	(61, 85, 16, 99, 7)	0.36342
Railway	(171, 229, 236, 75, 204)	−0.28694

Since the networks are smaller in size the budget k was fixed at 5 which is why there the algorithm detected 5 vulnerable nodes. The corresponding ARI score reported was the minimum across all possible combinations of the nodes

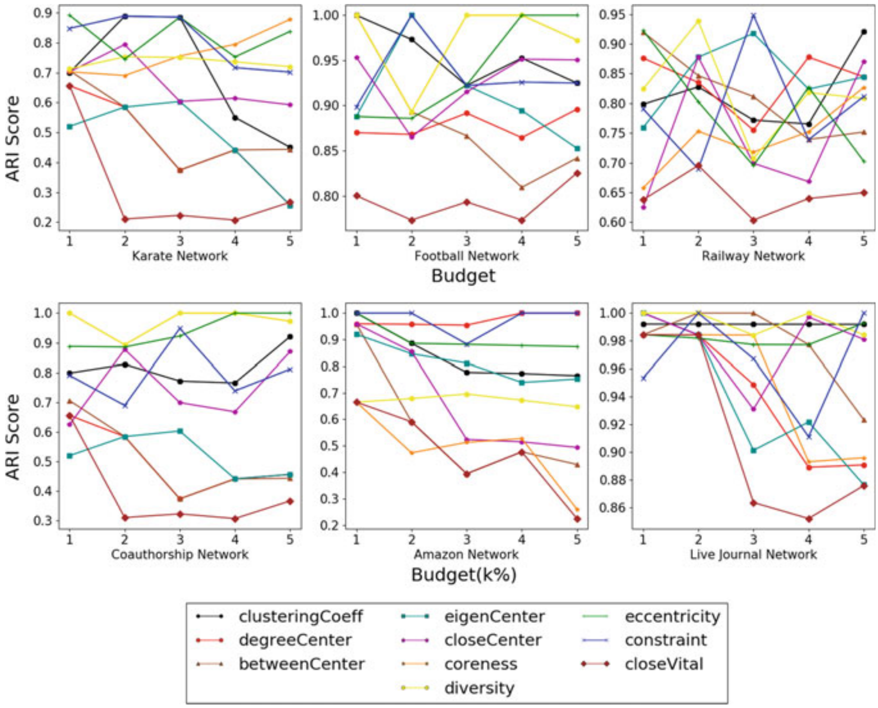


Fig. 6.5 Results of the network-based approach applied on several datasets with ARI being the target value function. We chose $k = 5$ for smaller networks and for larger networks we chose upto 5% of the total nodes within the network. The plots reported in this figure show that closeness vitality performed better than the other node-centric properties as for larger budgets the ARI between the resultant and the original network was low. When compared to the exhaustive results as reported in Table 6.6, we observe that the ARI values with closeness vitality as the greedy metric does not come close to the optimal answer

solution. For example, in the Railway network, we follow that the best ARI score obtained to be around 0.26 is close to the ground truth score of −0.28 compared to the 0.65 score obtained from Algorithm 1. So it is evident from this data that the difference between the optimal solution and the current solution has decreased, thereby establishing the superiority of Algorithms 3 over 2.

6.7.4 Task-Based Approach

Based on the results that we observed in the previous sections for the smaller networks, we perform similar tests on more extensive networks using Algorithm 4. To quantify this algorithm's performance, we use the widely use F1 score for the link prediction task. We evaluate the fraction of active nodes at the end of the few cas-

Table 6.7 Results of the community-based approach using ARI as the target value function. It shows the effects of different community-based metrics used to greedily select nodes

Network	Link Density	Conductance	Compactness
Karate	0.45034	0.25670	0.82691
Football	0.82530	0.64736	0.89587
Railway	0.44367	0.26693	0.27997
Co-authorship	0.71958	0.45670	0.75187
Amazon	0.69230	0.64979	0.76453
Live Journal	0.44367	0.25670	0.26693

We observe that conductance performs better to indicate the vulnerability of nodes in terms of the ARI between the resultant and the original community structure across all the datasets

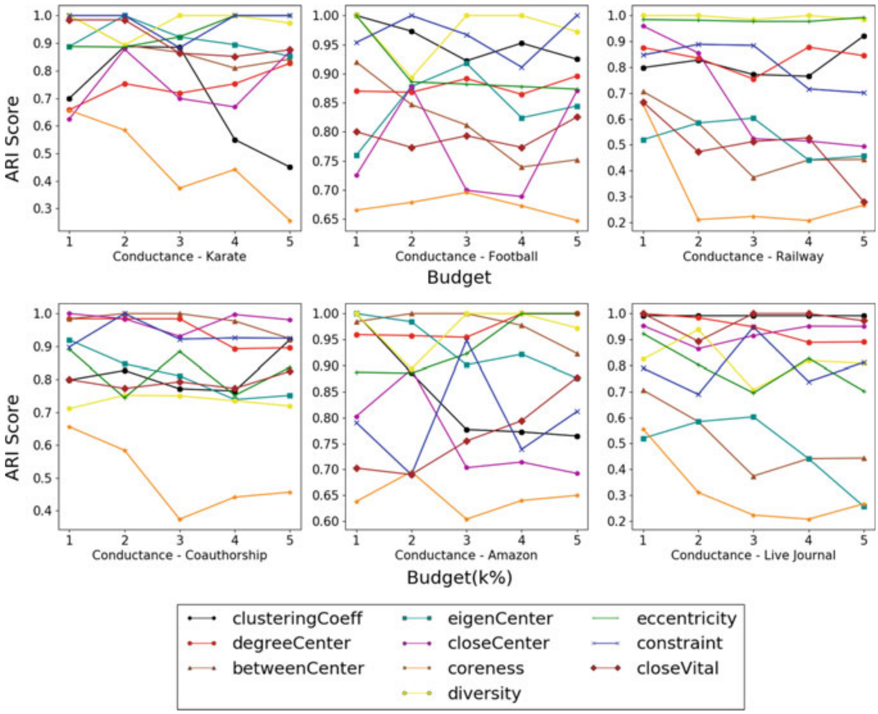


Fig. 6.6 Outcome of the community-based approach over all the datasets with ARI being the target value function. Based on Table 6.7, we observe that conductance performs better compared to other community-based methods to quantify the vulnerability of communities using ARI. The plots show the effects of different node-centric properties with conductance in Algorithm 3. The results show that across all the datasets, coreness outperforms other metrics. Upon comparing these results with the ground truth data in Table 6.6, we observe that the values are close to the optimal answers

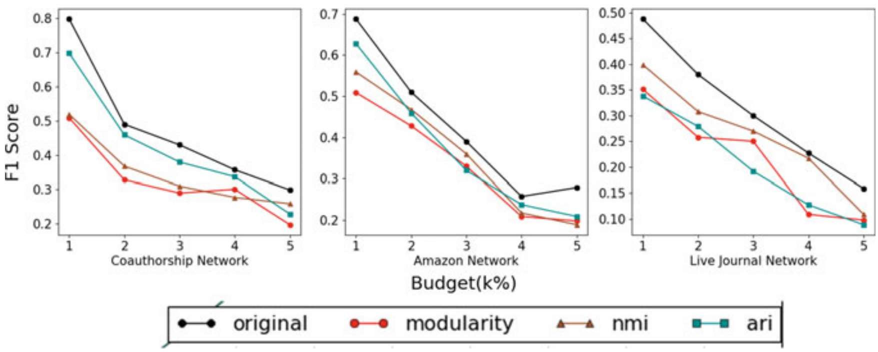


Fig. 6.7 Results of link prediction task over larger datasets with all the value functions. For modularity, we use Link Density combined with eigenvector centrality. For NMI, we use Link Density combined with eccentricity, and for ARI, we use conductance and closeness vitality. Original here represents when we use the original community structure for the nodes rather than the community structure after perturbing it. The plots indicate that when using these combinations for different value functions, the link prediction task's performance quantified with the F1 score decreases

cades for the information diffusion task. For each experiment, we consider k to be the percentage of nodes removed as otherwise the change in the community structure would not be enough to have significant effects. We have divided this section into two subsections to cover both the tasks that were described before.

Link Prediction: We test this task by assigning probabilities to the edges using three metrics separately: Within-Inter Cluster, Modified Common Neighbors, and Modified Resource Allocation. We find that Within-Inter Cluster produces better results compared to the other alternatives.

Based on Fig. 6.7, we observe that overall value functions the network's performance in the link prediction task has decreased, which is evident from the lower F1 scores. For each value function, we show the best combination as identified in the previous sections. The performance drop can be attributed to significant changes introduced into the system by removing vulnerable nodes. Their removal triggers significant structural perturbations in the underlying community structure, which causes the within-inter cluster method to assign lower probabilities to the edges due to fewer connections within the community and more connections across other communities. This decreased the likelihood of the test edge being classified as a valid link, thereby reducing the performance.

Information Diffusion: In Fig. 6.8, we observe that overall value functions the performance in the information diffusion task has decreased, which is evident from the lower fraction of active nodes. For this set of experiments, we set $p_i \geq p_o$ and let the cascade model run for 200 iterations. With a higher probability for the initial set and the subsequent set of active nodes to affect the nodes within their community, it is trivial to see that the fraction of nodes that will be active at the end of all the iterations

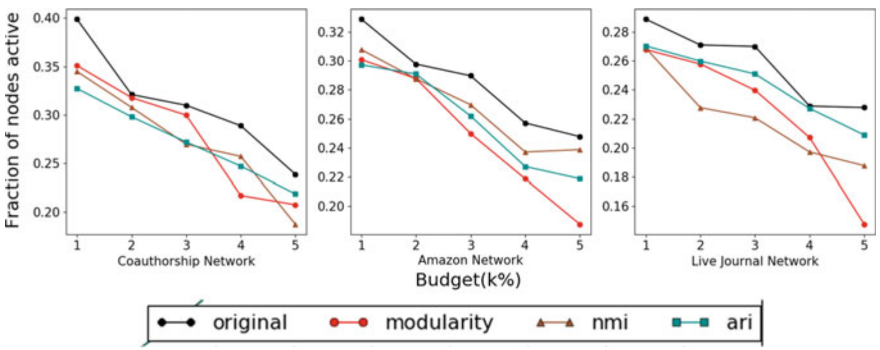


Fig. 6.8 Outcome of information diffusion task over larger networks with all the value functions. For modularity we use Link Density along with eigenvector centrality, for NMI we use Link Prediction combined with eccentricity and for ARI we utilize conductance combined with closeness vitality. Original here represents when we use the original community structure for the nodes rather than the community structure after perturbing it. The plots show that for all the target value functions the performance of the information diffusion task decreases. This performance was quantified using the fraction of *active* nodes after all the iterations

would be low. This is true if the underlying community structure was significantly perturbed and the network was highly disconnected, whereas it would be the opposite for the other case. For each value function, we show the best combination as identified in the previous sections.

This shows that the best combination of community-centric and network-centric nodes that we get from Algorithm 3 when applied to the more extensive networks using Algorithm 4 results in the decrease in the performance of the networks over both tasks that they are employed on, thereby validating our initial hypothesis. This establishes that Algorithm 3 can be applied to any general network irrespective of the size.

6.8 Conclusion

In this paper, we proposed a hierarchical greedy-based approach that efficiently identified critical nodes in the network, which significantly impacted the underlying community structure. Additionally, we also proposed a novel task-based strategy to apply the results of the hierarchical greedy-based approach on more extensive networks and quantify its effectiveness, which would enable us to estimate the performance of the algorithm in a real-world context.

Due to the extensive size of our experiments, we show our best results only. Since Algorithm 1 is exhaustive and hence was applied only to small networks such as Karate, Football, and Railway Network. The results of this algorithm provided us with the benchmark to compare with our other algorithms. We further saw that

Algorithm 2 was not that promising and were far from the gold standard in comparison to Algorithm 3 which came close to the gold standard. This comparison showed that Algorithm 3 works best for small networks. As mentioned previously, we used Algorithm 4 to compare the performance of Algorithms 2 and 3 for large networks such as Co-authorship, Amazon, and Live Journal Networks. Based on these results, we established that when we use Algorithm 3, we get a performance drop over both the tasks, namely, link prediction and information diffusion, compared to the original network. This establishes the generalizability of Algorithm 3.

To conclude, this work has provided a hierarchical approach that allowed for identifying the vulnerable nodes in a network efficiently. The proposed method was used to analyze the community vulnerability of several networks whose validity was established using both exhaustive and task-based approaches depending on the network's size.

Acknowledgements T. Chakraborty would like to acknowledge the support of SERB (ECR/2017/001691) and the Infosys Centre for AI, IIIT-Delhi.

References

1. Chakraborty, T., Dalmia, A., Mukherjee, A., Ganguly, N.: Metrics for community analysis: a survey. ACM Comput. Surv. (CSUR) **50**(4), 54 (2017)
2. Lancichinetti, A., Fortunato, S.: Community detection algorithms: a comparative analysis. Phys. Rev. **80**(5) (2009). https://doi.org/10.1103/PhysRevE.80.056117
3. Newman, M.E.J.: Modularity and community structure in networks. Proc. Natl. Acad. Sci. **103**(23), 8577–8582 (2006). https://doi.org/10.1073/pnas.0601602103
4. Danon, L., Díaz-Guilera, A., Duch, J., Arenas, A.: Comparing community structure identification. J. Stat. Mech. Theory Exp. **2005**(09), P09008 (2005). http://stacks.iop.org/1742-5468/2005/i=09/a=P09008
5. Hubert, L., Arabie, P.: Comparing partitions. J. Classif. **2**, 193–218 (1985). https://doi.org/10.1007/BF01908075
6. Li, P., Salour, M., Su, X.: A survey of internet worm detection and containment. IEEE Commun. Surv. Tutor. **10**(1), 20–35 (2008). https://doi.org/10.1109/COMST.2008.4483668
7. Alim, M.A., Li, X., Nguyen, N.P., Thai, M.T., Helal, A.: Structural vulnerability assessment of community-based routing in opportunistic networks. IEEE Trans. Mob. Comput. **15**(12), 3156–3170 (2016). https://doi.org/10.1109/TMC.2016.2524571
8. Wei, D., Zhang, X., Mahadevan, S.: Measuring the vulnerability of community structure in complex networks. Reliab. Eng. Syst. Saf. **174**, 41–52 (2018). https://doi.org/10.1016/j.ress.2018.02.001
9. Girvan, M., Newman, M.E.J.: Community structure in social and biological networks. Proc. Natl. Acad. Sci. **99**(12), 7821–7826 (2002). https://doi.org/10.1073/pnas.122653799
10. Blondel, V.D., Guillaume, J.L., Lambiotte, R., Lefebvre, E.: Fast unfolding of communities in large networks. J. Stat. Mech. Theory Exp. **2008**(10), P10008 (2008). http://stacks.iop.org/1742-5468/2008/i=10/a=P10008
11. Clauset, A., Newman, M.E.J., , Moore, C.: Finding community structure in very large networks. Phys. Rev. E, 1– 6 (2004). https://doi.org/10.1103/PhysRevE.70.066111. www.ece.unm.edu/ifis/papers/community-moore.pdf
12. Guimera, R., Amaral, L.A.N.: Functional cartography of complex metabolic networks. Nature **433**(7028), 895–900 (2005). http://dx.doi.org/10.1038/nature03288

13. Newman, M.: Fast algorithm for detecting community structure in networks. Phys. Rev. E, **69** (2003). http://arxiv.org/abs/cond-mat/0309508
14. Farkas, I., Ábel, D., Palla, G., Vicsek, T.: Weighted network modules. New J. Phys. **9**(6), 180–180 (2007). https://doi.org/10.1088/1367-2630/9/6/180
15. Palla, G., Derenyi, I., Farkas, I., Vicsek, T.: Uncovering the overlapping community structure of complex networks in nature and society. Nature **435**(7043), 814–818 (2005). http://dx.doi.org/10.1038/nature03607
16. Rosvall, M., Bergstrom, C.T.: An information-theoretic framework for resolving community structure in complex networks. Proc. Natl. Acad. Sci. **104**(18), 7327–7331 (2007). https://doi.org/10.1073/pnas.0611034104
17. Rosvall, M., Bergstrom, C.T.: Maps of random walks on complex networks reveal community structure. Proc. Natl. Acad. Sci. **105**(4), 1118–1123 (2008). https://doi.org/10.1073/pnas.0706851105
18. Raghavan, U., Albert, R., Tirupatikumara, S.: Near linear time algorithm to detect community structures in large-scale networks. Phys. Rev. E Stat. Nonlinear Soft Matter Phys. **76**(3), (2007). https://doi.org/10.1103/PhysRevE.76.036106
19. Xie, J., Szymanski, B.K.: Community detection using a neighborhood strength driven label propagation algorithm. In: Proceedings of the 2011 IEEE Network Science Workshop, NSW '11, pp. 188–195. IEEE Computer Society, Washington, DC, USA (2011). https://doi.org/10.1109/nsw.2011.6004645
20. Xie, J., Szymanski, B.K.: Towards linear time overlapping community detection in social networks. In: Proceedings of the 16th Pacific-Asia Conference on Advances in Knowledge Discovery and Data Mining—Volume Part II, PAKDD'12, pp. 25–36. Springer-Verlag, Berlin (2012). http://dx.doi.org/10.1007/978-3-642-30220-6_3
21. Newman, M.E.J.: Community detection and graph partitioning. EPL (Eur. Lett.) **103**(2), 28003 (2013). https://doi.org/10.1209/0295-5075/103/28003
22. Richardson, T., Mucha, P.J., Porter, M.A.: Spectral tripartitioning of networks. Phys. Rev. E **80**(3), (2009). https://doi.org/10.1103/PhysRevE.80.036111
23. Baumes, J., Goldberg, M., Magdon-Ismail, M.: Efficient identification of overlapping communities. In: Kantor, P., Muresan, G., Roberts, F., Zeng, D.D., Wang, F.Y., Chen, H., Merkle, R.C. (eds.) Intelligence and Security Informatics, pp. 27–36. Springer, Berlin (2005)
24. Lancichinetti, A., Fortunato, S., Kertész, J.: Detecting the overlapping and hierarchical community structure in complex networks. New J. Phys. **11**(3), 033015 (2009). https://doi.org/10.1088/1367-2630/11/3/033015
25. De Meo, P., Ferrara, E., Fiumara, G., Provetti, A.: Enhancing community detection using a network weighting strategy. Inf. Sci. **222**, 648–668 (2013). https://doi.org/10.1016/j.ins.2012.08.001
26. Kanawati, R.: YASCA: an ensemble-based approach for community detection in complex networks. In: International Computing and Combinatorics Conference, pp. 657–666. Springer, Cham (2014)
27. Lancichinetti, A., Radicchi, F., Ramasco, J.J., Fortunato, S.: Finding statistically significant communities in networks. PLoS One **6**(4), 1–18 (2011). https://doi.org/10.1371/journal.pone.0018961
28. Bader, D.A., Meyerhenke, H., Sanders, P., Wagner, D. (eds.): Graph Partitioning and Graph Clustering-10th DIMACS Implementation Challenge Workshop, Georgia Institute of Technology, Atlanta, GA, USA, February 13–14, 2012. Proceedings, Contemporary Mathematics, vol. 588. American Mathematical Society (2013). https://doi.org/10.1090/conm/588
29. Riedy, J., Bader, D.A., Jiang, K., Pande, P., Sharma, R.: Detecting Communities From Given Seeds in Social Networks. Georgia Institute of Technology (2011)
30. Agarwal, P., Verma, R., Agarwal, A., Chakraborty, T.: Dyperm: maximizing permanence for dynamic community detection. In: Pacific-Asia Conference on Knowledge Discovery and Data Mining, pp. 437–449. Springer, Berlin (2018)
31. Rossetti, G., Cazabet, R.: Community discovery in dynamic networks: a survey. ACM Comput. Surv. (CSUR) **51**(2), 35 (2018)

32. Baumes, J., Goldberg, M.K., Krishnamoorthy, M.S., Magdon-Ismail, M., Preston, N.: Finding communities by clustering a graph into overlapping subgraphs. IADIS AC **5**, 97–104 (2005)
33. Havemann, F., Heinz, M., Struck, A., Gläser, J.: Identification of overlapping communities and their hierarchy by locally calculating community-changing resolution levels. J. Stat. Mech. Theory Exp. **2011**(01), P01023 (2011). https://doi.org/10.1088/1742-5468/2011/01/p01023
34. Lee, C., Reid, F., McDaid, A., Hurley, N.: Detecting highly overlapping community structure by greedy clique expansion. KDD SNA 2010 (2010)
35. Adamcsek, B., Palla, G., Farkas, I.J., Derenyi, I., Vicsek, T.: CFinder: locating cliques and overlapping modules in biological networks. Bioinformatics **22**(8), 1021–1023 (2006). https://doi.org/10.1093/bioinformatics/btl039
36. Nepusz, T., Petróczi, A., Négyessy, L., Bazsó, F.: Fuzzy communities and the concept of bridgeness in complex networks. Phys. Rev. E Stat. Nonlinear Soft Matter Phys **77**(1), 016107 (2008)
37. Newman, M.E.J., Leicht, E.A.: Mixture models and exploratory analysis in networks. Proc. Natl. Acad. Sci. **104**(23), 9564–9569 (2007). https://doi.org/10.1073/pnas.0610537104
38. Nowicki, K., Snijders, T.A.B.: Estimation and prediction for stochastic blockstructures. J. Am. Stat. Assoc. **96**(455), 1077–1087 (2001). https://doi.org/10.1198/016214501753208735
39. Ren, W., Yan, G., Liao, X., Xiao, L.: Simple probabilistic algorithm for detecting community structure. Phys. Rev. E **79**, (2009). https://doi.org/10.1103/PhysRevE.79.036111
40. Zarei, M., Izadi, D., Samani, K.A.: Detecting overlapping community structure of networks based on vertex–vertex correlations. J. Stat. Mech. Theory Exp. **2009**(11), P11013 (2009). https://doi.org/10.1088/1742-5468/2009/11/p11013
41. Chen, W., Liu, Z., Sun, X., Wang, Y.: A game-theoretic framework to identify overlapping communities in social networks. Data Min. Knowl. Discov. **21**, 224–240 (2010). https://doi.org/10.1007/s10618-010-0186-6
42. Yang, J., Leskovec, J.: Overlapping community detection at scale: a nonnegative matrix factorization approach. In: Proceedings of the Sixth ACM International Conference on Web Search and Data Mining, pp. 587–596. WSDM '13, ACM, New York, USA (2013). https://doi.org/10.1145/2433396.2433471
43. Zhang, S., Wang, R.S., Zhang, X.: Identification of overlapping community structure in complex networks using fuzzy c-means clustering. Phys. Stat. Mech. Appl. **374**, 483–490 (2007). https://doi.org/10.1016/j.physa.2006.07.023
44. Chakraborty, T., Kumar, S., Ganguly, N., Mukherjee, A., Bhowmick, S.: Genperm: a unified method for detecting non-overlapping and overlapping communities. IEEE Trans. Knowl. Data Eng. **28**(8), 2101–2114 (2016)
45. Chakraborty, T.: Leveraging disjoint communities for detecting overlapping community structure. J. Stat. Mech. Theory Exp. **2015**(5), P05017 (2015)
46. Chakraborty, T., Ghosh, S., Park, N.: Ensemble-based overlapping community detection using disjoint community structures. Knowl. Based Syst. **163**, 241–251 (2019)
47. Chakraborty, T., Park, N.: Ensemble-based discovery of disjoint, overlapping and fuzzy community structures in networks. arXiv:1712.02370 (2017)
48. Chakraborty, T., Park, N., Subrahmanian, V.: Ensemble-based algorithms to detect disjoint and overlapping communities in networks. In: 2016 IEEE/ACM International Conference on Advances in Social Networks Analysis and Mining (ASONAM), pp. 73–80. IEEE (2016)
49. Nguyen, N.P., Alim, A., Shen, Y., Thai, M.T.: Assessing network vulnerability in a community structure point of view. In: 2013 IEEE/ACM International Conference on Advances in Social Networks Analysis and Mining (ASONAM 2013), pp. 231–235 (2013). https://doi.org/10.1145/2492517.2492644
50. Nguyen, H.T., Nguyen, N.P., Vu, T., Hoang, H.X., Dinh, T.N.: Transitivity demolition and the fall of social networks. IEEE Access **5**, 15913–15926 (2017). https://doi.org/10.1109/ACCESS.2017.2672666
51. Chan, H., Akoglu, L., Tong, H.: Make it or break it: manipulating robustness in large networks. In: Proceedings of the 2014 SIAM International Conference on Data Mining, pp. 325–333. SIAM (2014)

52. Frank, H., Frisch, I.: Analysis and design of survivable networks. IEEE Trans. Commun. Technol. **18**(5), 501–519 (1970). https://doi.org/10.1109/TCOM.1970.1090419
53. Fiedler, M.: Algebraic connectivity of graphs. Czechoslov. Math. J. **23**(2), 298–305 (1973). http://eudml.org/doc/12723
54. Holme, P., Kim, B.J., Yoon, C.N., Han, S.K.: Attack vulnerability of complex networks. Phys. Rev. E **65**, (2002). https://doi.org/10.1103/PhysRevE.65.056109
55. Allesina, S., Pascual, M.: Googling food webs: can an eigenvector measure species' importance for coextinctions? PLoS Comput. Biol. **5**(9), 1–6 (2009). https://doi.org/10.1371/journal.pcbi.1000494
56. Ramirez-Marquez, J.E., Rocco, C.M., Barker, K., Moronta, J.: Quantifying the resilience of community structures in networks. Reliab. Eng. Syst. Saf. **169**, 466–474 (2018). https://doi.org/10.1016/j.ress.2017.09.019
57. Grubesic, T.H., Matisziw, T.C., Murray, A.T., Snediker, D.: Comparative approaches for assessing network vulnerability. Int. Reg. Sci. Rev. **31**(1), 88–112 (2008). https://doi.org/10.1177/0160017607308679
58. Freeman, L.C.: A set of measures of centrality based on betweenness. Sociometry **40**(1), 35–41 (1977). http://www.jstor.org/stable/3033543
59. Bonacich, P.: Factoring and weighting approaches to status scores and clique identification. J. Math. Sociol. **2**(1), 113–120 (1972). https://doi.org/10.1080/0022250X.1972.9989806
60. Bavelas, A.: Communication patterns in task-oriented groups. Acoust. Soc. Am. J. **22**, 725 (1950). https://doi.org/10.1121/1.1906679
61. Sabidussi, G.: The centrality index of a graph. Psychometrika **31**, 581–603 (1966). https://doi.org/10.1007/BF02289527
62. Batagelj, V., Zaversnik, M.: An O (m) algorithm for cores decomposition of networks. CoRR. arXiv:cs.DS/0310049 (2003). http://arxiv.org/abs/cs.DS/0310049
63. Eagle, N., Macy, M., Claxton, R.: Network diversity and economic development. Science **328**(5981), 1029–31 (2010)
64. Burt, R.: Structural holes and good ideas. Am. J. Sociol. **110**(2), 349–399 (2004). http://www.jstor.org/stable/10.1086/421787
65. Brandes, U.: Network Analysis: Methodological Foundations, vol. 3418. Springer Science & Business Media (2005)
66. Valverde-Rebaza, J., Lopes, A.: Structural link prediction using community information on twitter. In: Proceedings of the 2012 4th International Conference on Computational Aspects of Social Networks, CASoN '12 (2012). https://doi.org/10.1109/CASoN.2012.6412391
67. Soundarajan, S., Hopcroft, J.: Using community information to improve the precision of link prediction methods. In: Companion Proceedings of the 21st International Conference on World Wide Web, WWW '12, pp. 607–608 (2012). https://doi.org/10.1145/2187980.2188150
68. Lin, S., Hu, Q., Wang, G., Yu, P.S.: Understanding community effects on information diffusion. In: Cao, T., Lim, E.P., Zhou, Z.H., Ho, T.B., Cheung, D., Motoda, H. (eds.) Advances in Knowledge Discovery and Data Mining, pp. 82–95. Springer International Publishing, Cham (2015)
69. Zachary, W.: An information flow model for conflict and fission in small groups. J. Anthropol. Res. **33**, 452–473 (1977)
70. Zachary, W.W.: Zachary karate club network dataset–KONECT (2017). http://konect.uni-koblenz.de/networks/ucidata-zachary
71. Chakraborty, T., Srinivasan, S., Ganguly, N., Mukherjee, A., Bhowmick, S.: On the permanence of vertices in network communities. In: Proceedings of the 20th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, pp. 1396–1405. KDD '14, ACM, New York, USA (2014). https://doi.org/10.1145/2623330.2623707
72. Yang, J., Leskovec, J.: Defining and evaluating network communities based on ground-truth. CoRR. arXiv:abs/1205.6233 (2012). http://arxiv.org/abs/1205.6233

Chapter 7
Community Detection in Multidimensional and Multilayer Networks

Soumita Das and Anupam Biswas

Abstract The use of social networking platforms has seen rapid growth recently. People connect to each other using varied platforms but their actions differ with respect to the platforms. This results in heterogeneous relations such as functional relations, spatial relations, and temporal relations. In order to look the interactions from different perspectives and ensemble people with similar activities, an extensive study on community detection in heterogeneous networks is highly recommendable. Particularly, recent research activities have gained a lot of attention for community detection in multidimensional networks and multilayer networks. To this end, we open on various important features of these heterogeneous networks considering their role in the community detection process. Successively, we reviewed several current strategies to further process the earlier heterogeneous community detection approaches based on these features followed by some of the evaluation strategies. Our study aims to guide our attention toward the numerous approaches to detect shared and unshared communities across multiple graph layers, the challenges to the previous approaches, and the developments in the recent approaches.

Keywords Multidimensional networks · Multilayer networks · Community detection

7.1 Introduction

Community detection is an inseparable part of complex networks. Tremendous amount of research attention has been guided toward complex networks because recent studies have witnessed that real-world networks are complicated in nature. In order to understand such complex networks, we have selectively picked Social Networks (SN) to shed subsequent light on the heterogeneous activities and behavior

S. Das (✉) · A. Biswas
Department of Computer Science and Engineering, National Institute of Technology, Silchar 788010, Assam, India
e-mail: anupam@cse.nits.ac.in

A. Biswas et al. (eds.), *Principles of Social Networking*,
Smart Innovation, Systems and Technologies 246,
https://doi.org/10.1007/978-981-16-3398-0_7

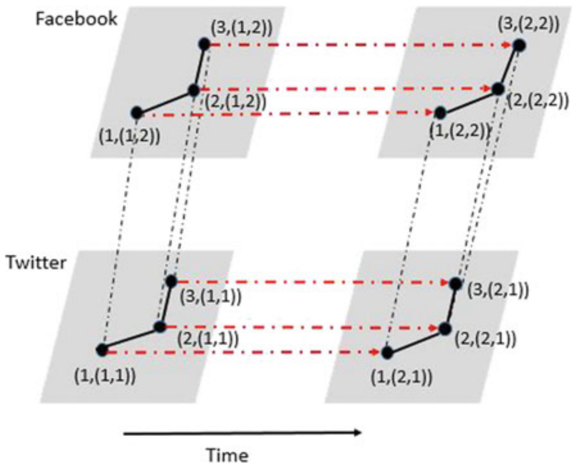


Fig. 7.1 Multilayer network example

of people and community detection is very significant in our discussion particularly to comprehend the dynamics of these SN [1]. However, the definition of community is defined differently by various set of users such as [2–4],

1. It is a set of nodes that are densely connected across all network dimensions.
2. It is a high-density region that exists across some dimensions.
3. It also signifies a set of users having more similar properties as compared to users outside the group. Nodes that exhibit similar properties share connections with each other and hence, set of similar nodes are densely connected.
4. It is also defined as set of users interacting more frequently.

The objective of community detection algorithms is to partition the densely connected components inherent in the network. We consider SN here because it deals with multiple types of interactions. Considering heterogeneous interactions leads to better community detection because it provides better information about the community membership of nodes [5]. It is relatable in our discussion because in real world, people interact with each other using different platforms such as phone, SN, or face-to-face and thus, heterogeneity is inherent in human interactions. Hence, in this chapter, we opted for heterogeneous networks. We shall mainly discuss the multidimensional and multilayer type of heterogeneous network. An example of *multilayer network* have been shown diagrammatically in Fig. 7.1. It considers two aspects such as time slot and social media platform respectively where the first aspect is ordered and the second aspect is unordered. Here, the figure shows two time slots represented by $\mathcal{L}_1 = 1, 2$ and two social media platforms represented by $\mathcal{L}_2 = 1, 2$ where *Twitter* is labeled as 1 and *Facebook* is labeled as 2. The undirected intralayer edges have been indicated by black solid lines, directed intralayer edges across different time slots are represented by red dotted lines with arrow, and undirected intralayer edges across same time slot have been represented by black dotted lines. Here, a state node is represented as $(i, (\alpha_1, \alpha_2))$ where $i \in 1, 2, 3$ and $\alpha_1 = 1, 2$ and $\alpha_2 = 1, 2$. The total number of layers is $l = \mathcal{L}_1 \times \mathcal{L}_2$.

The heterogeneous networks have gained enormous popularity in the recent time because of their ability to consider various dynamical nodes concurrently. These heterogeneous networks deal with multiple types of interactions or objects or relationships that exist between entities [6–8]. The heterogeneity is represented by numerous layers in a multilayer network which comprises varied edges between the same set of entities, e.g., nodes representing a set of friends may have lunch together or share friendship link on facebook or it may be comprised of diverse entities, e.g., the relationship of authorship that employees and research papers shares among themselves [9]. The varied relationships where each of the relationship plays a unique role for a particular task are taken care by the *multilayer networks* and *multidimensional networks* by capturing the connections/interactions derived from activity data [10]. It is important to consider all the vital information hidden in the existing relationships in order to work on a community detection problem [11]. It is also to be mentioned here that there are different terms for defining *multilayer networks* such as *multiplex networks, multilevel networks, multidimensional networks, multirelational networks, edge-colored networks, node-colored networks, multifaceted networks, and independent networks* [12].

Modeling and analyzing such large networks is not feasible and boundary specification may lead to loss of significant data. Hence, there arises the need for network simplification which targets to simplify the structure of the network by replacing some objects (i.e., nodes and edges). Network simplification can be extensively classified into three broad categories such as selection, aggregation, and transformation. Selection is a method to contract the size of heterogeneous networks by incorporation of filtering or sampling subsets of nodes, edges and/or layers. Here, the objects removed are not replaced by any other objects. Whereas aggregation refers to a more condensed approach of network simplification which incorporates partitional or hierarchical grouping mechanisms. This approach replaces the removed objects with some different types of objects. While transformation is another approach of network compaction where the removed objects (i.e., nodes and edges) are replaced, i.e., transformed into different types of objects. Transformation can be classified into projection and graph embedding methods.

Recent research activities focus toward finding the hidden structures in heterogeneous networks. Earlier multidimensional community detection approaches had a lot of challenges such as requirement of tuning of user-supplied parameters, consideration of only the topological features or attribute information, and ignoring semantic information. Hence, the previous approaches were not efficient enough to detect heterogeneous communities. In this study, our aim has been to exploit the varied representations and important features of these heterogeneous networks with the objective of detecting better communities. Our discussion also focuses on the previous challenges, developments in the recent approaches, type of community detected, and evaluation strategies to predict the quality and accuracy of detected communities.

Table 7.1 Notations and terminology used in this paper

Symbol	Name
G	Graph
$\mathcal{M}$	Multilayer network
$\mathfrak{P}$	Binary relation
α, β	Layers
n, m	Set of nodes
$A = \{a_{ij}\}$	Adjacency matrix
$\mathscr{P}$	Participation matrix
A^{α}	Layer Adjacency matrix
$\mathscr{C}$	Adjacency coupling matrix
$\mathcal{N}$	Set of nodes $\{n_1, n_2, \ldots, n_{\|N\|}\}$
L; $\mathcal{L}$	Layer; set of layers
R	Set of relations
d	Dimension
v_0	Seed node

7.2 Representation of Multidimensional Networks and Multilayer Networks

Heterogeneity in Online Social Networks (OSNs) indicates nodes and links of varied types. In these types of networks, each node may be connected to several different categories of nodes by different types of links. To understand the structure and properties of this heterogeneous behavior in OSNs, existing network representations of heterogeneous networks with several constraints to the general framework have been discussed here. Each of these network representations is particularly helpful in certain specific situation. The terminologies associated with various network representations have been shown in Table 7.1. Some of these representations along with their advantages have been discussed as follows:

1. **Adjacency Matrix**: These types of representations are particularly used to provide an edge-oriented view of the graphs. With these types of representations, we analyze the edge-similarity. Although the adjacency matrix representation is useful to represent single layer networks, it is not efficient to cover the multiple aspects of a multilayer network. In order to represent the information possessed by a multilayer network $\mathcal{M}$, it is represented using multiple adjacency matrices $A^{[\alpha]}$ for each of the layers G_α [13–15].

   (a) **Adjacency matrix of a layer graph**: The adjacency matrix for a layer graph G_α is represented by a symmetric matrix $n_\alpha \times n_\alpha$ where $A^\alpha = a^\alpha_{ij}$ with $a_{ij} =$ 1 only if an edge exists between (i, α) and (j, α) in G_α.

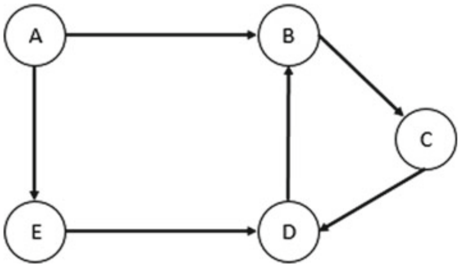


Fig. 7.2 A simple graph G

(b) **Participation matrix**: The adjacency matrix of $G_{\mathfrak{P}}$ is represented by an $n \times m$ matrix $\mathscr{P} = p_{i\alpha}$ and $p_{i\alpha} = 1$ if and only if node i participates in layer α. Such a matrix is called participation matrix.

(c) **Adjacency matrix of a coupling Graph**: Adjacency matrix for a coupling graph $G_{\mathfrak{C}}$ is indicated by an $N \times N$ matrix where $mathscrC = \{c_{ij}\}$ with $c_{ij} = 1$ only when node-layer pair i and j in $G_{\mathfrak{C}}$ are connected and they represent the same node in different layers.

2. **Adjacency list**: In graph representation, edges or arcs are represented as a list using adjacency list representation. In an undirected graph, each entry represents a set of two nodes comprising of two ends of an edge. Whereas, in a directed graph, each entry comprises a source node and a destination node. It is also to be mentioned here that there are many advantages of an adjacency list representation over adjacency matrix representation such as flexibility with addition and deletion of nodes in graph G, the nodes need not be ordered, maximum space utilization. Here, maximum space utilization means that in contrast to adjacency matrix representation, where if the number of edges equals to the total number of nodes in the network, then the resulting adjacency matrix is a sparse matrix, which wastes a huge space [16]. In order to overcome these issues led to the outbreak of adjacency list representation. An adjacency list representation of the graph in Fig. 7.2 is shown in Table 7.2.
3. **Adjacency Tensor**: Tensors are utilized to represent complicated set of relationships that are dynamic in nature. An example of rank-2 tensors is matrices.

Table 7.2 Adjacency list representation of Graph G

Node	Adjacency list
A	B, E
B	C
C	D
D	B
E	D

However, complicated relationships are represented by using tensors of higher order. Adjacency tensor $W_{\beta}\alpha$ as a linear combination of tensors is represented as

$$W_{\beta}^{\alpha} = \sum_{i,j=1}^{N} w_{ij} e^{\alpha}(i) e_{\beta}(j) = \sum_{i,j=1}^{N} w_{ij} E_{\beta}^{\alpha}(i, j), \tag{7.1}$$

In Eq. 7.1, $E_{\beta}^{\alpha}(i, j) \in \mathbb{R}^{N \times N}$ indicates the tensor product of the canonical vectors assigned to nodes n_i and n_j. The rank of such tensors can be reduced by fattening method. Then, the tensors can be represented using super-adjacency matrices. An advantage of super-adjacency matrix representation is that missing nodes are represented in a convenient way. It is also to be added here that Tensor representation is useful for tracing the influence throughout the network [15, 17].

4. **Supra-adjacency Matrix**: The components of the coupling graph $G_{\mathfrak{C}}$ are called supra-nodes. A supra-graph denoted by $G_{\mathcal{M}}$ is the combination of the layer-graph $G_{\mathfrak{l}}$ and coupling graph $G_{\mathfrak{C}}$. As the name implies, supra-adjacency matrix indicates adjacency matrix of a supra-graph $G_{\mathcal{M}}$.

$$\bar{\mathscr{A}} = \bigoplus_{\alpha} A^{\alpha} + \mathscr{C} \tag{7.2}$$

In Eq. 7.2, $A = \bigoplus A^{\alpha}$ which indicates intralayer adjacency matrices. These types of representations are particularly helpful to calculate several basic metrics such as degree of a node-layer, layer degree of a node-layer, and coupling degree of a node-layer which have been further illustrated in [13]. These network representation approach has the advantage over tensors because of the convenience of representing missing nodes [17].

5. **Multigraph**: This is an edge-colored/edge-labeled-based network representation used to represent various types of edges in a multilayer/multidimensional network. It is defined by a triple $G = (V, E, C)$ where V indicates the node set, $E \subset V \times V \times C$ is the edge labeled set where C is the color set/label set. In multidimensional network, "label" indicates dimension. The belongingness of a node to a certain dimension say d is based on the existence of at least one edge with dimension d that is adjacent to the given node. Whereas the belongingness of an edge to a certain dimension d is based on whether it's label is d. It is to be mentioned here that in edge-colored representation of multigraph, edges incident on the same node are represented by the same color. As been mentioned earlier that this network representation comprises multiple edges where different edges may represent different kinds of relations such as friendship, arguments, and horseplay possessed by a multilayer network which can be represented graphically, algebraically or using sociometric notations and is represented using multigraph [18–23].

6. **Exponential Random Graph Models (ERGMs:)** ERGMs considers networks to be endogenic and it focuses on the statistical analysis of the network under consideration. It is particularly applied in bipartite and multipartite networks.

ERGMs for multilevel nodes comprising of two levels have been discussed in. The interdependence of varied networks in a multivariate network has also been studied. It requires modeling of within-level one-mode networks and the cross-level bipartite networks together. It has been found that highly structured within-level networks are created by simple cross-level effects [24, 25].

7.3 Important Features to Be Considered

There are several important features associated with a heterogeneous network which would assist in further understanding of the structural properties of the network and hence, in detecting communities inherent in a network. In this section, we present various features of the problem representation which are as follows:

1. **Node-aligned/Fully aligned**: When all the nodes that are present in a multilayer network participate in all the layers, the network is said to be fully aligned or node-aligned. Hence, a node-aligned multiplex network is defined as
   Definition: *Node-aligned Multiplex network* : A node-aligned multiplex network is a multiplex network $(\mathcal{N}, \mathcal{L}, V, E)$ where $\forall n \in \mathcal{N}, l \in \mathcal{L} : (n, l) \in V$.
   It is obvious that when the nodes are fully aligned, finding overlapping nodes considering all the layers present in the network is redundant. These feature of the heterogeneous network also affects many other measures such as checking the presence of anti-correlated layers, edge- and triangle-based overlapping measures have the same overlapping measures in the node-aligned and non-aligned networks, correlation between different measurements is highly evident in node-aligned networks [13, 14].
2. **Weak ties**: The significance of weak ties lies in its ability to consider holes present in the network. The presence of weak ties in a network assists in the information flow through the remote parts of a network. So, it is evident that entities that do not have weak ties are deprived of remote information and get only the information from their close friends. Thus, weak ties may act as bridges between network fragments and hence, it helps to integrate social systems [26–28].
3. **Eigen vector centality**: The quantification of the importance of nodes in a network is performed using eigenvector centrality. Like, a node says n_i will have a high centrality value when it's neighboring nodes have a high centrality value. To illustrate properly, let us consider an adjacency matrix A for an undirected network and let v be the solution of the equation $Av = \lambda_1 v$. Here, suppose λ_1 be the largest eigenvalue of A, then v is the leading eigenvector of A and eigenvalue centrality are given by the components of v. The eigenvector centrality of node n_i is given by v_i [15].
4. **Overlapping Communities**: In the context of multilayer network or multidimensional network, *overlapping communities* are those communities that are common between various layers/dimensions and is shown in Fig. 7.3 where C_1, C_2, C_3 represents overlapping clusters. Basically, it allows us to compress the redundant

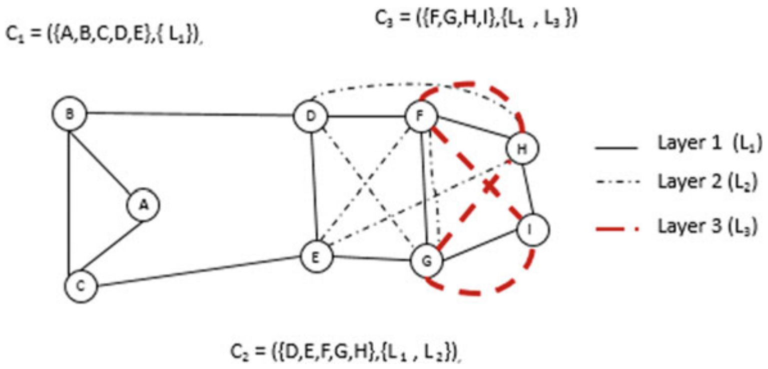


Fig. 7.3 An example of overlapping clusters

information inherent in the layers/dimensions of a heterogeneous network. It deals with all types of network behaviors like assortative, disassortative, directed, etc. Whereas, in directed networks, nodes might play different roles and hence may belong to numerous groups [29].

5. **Dynamic**: There are some edges and nodes in a network that can appear and disappear with respect to time. Similarly, communities in a multidimensional/multilayer network also evolve with times which leads to dynamic communities [30].
6. **Non-overlapping/Disjoint communities**: These communities comprises a set of nodes that are densely connected inside a group and loosely connected outside the group and these set of nodes do not share any relation with nodes in other layers/dimensions [31, 32].

7.4 Community Detection in Heterogeneous Networks

In this section, we discuss the approaches adopted to detect communities in heterogeneous networks which have been summarized in Table 7.3. Primarily for our discussion, we considered the *multilayer network* and *multidimensional network*. But a multidimensional network is also a variant of multilayer network and the motivation behind this discussion is that communities contribute to the structure and function of the system. So, the organization of network into communities stands out to be very significant. But, community detection remains to be a complex issue till date because there is no standard definition of community. Different authors describe a community from varied perspectives. Whatever be the approaches undertaken by the community detection algorithm, it is true that the result expected in the community detection problem is a list of a set of densely connected entities. In addition to this, the challenges to previous approaches, developments in current approach, type of

Table 7.3 Summary of recent multidimensional and multilayer community detection approaches

References	Network representation	Algorithm/Model	Type of community
Tagarelli et al. [33]	Adjacency list	M-EMCD	Non-overlapping
Liu et al. [34]	Set of layers	lcd	Overlapping
Li et al. [35]	Multilayer tensor	Multilayer network community detection (IMLC) based on influence measurement	–
Shahmoradi et al. [36]	Adjacency matrix	Multilayer community detection (MLCD)	Overlapping
Chouchane et al. [19]	Multigraph	MCDA	Disjoint
Chen et al. [37]	Conventional graph model	MSH-LPA	Overlapping
Contisciani et al. [38]	Adjacency tensor	MTCOV-EM	Overlapping
Naderipour et al. [26]	Two-layer graph	PCMTL	Overlapping
Guesmi et al. [8]	Relational Context Family (RCF)	CoMRCA	Disjoint community
Pizzuti and Socievole [39]	Adjacency list	ML@NetDE	–

community detected, the network representation, assisting features, and evaluation metrics utilized by various approaches have been covered in this discussion.

7.4.1 Community Detection in Multilayer Networks

The problem of community detection in *multilayer networks* is to identify the underlying community structures by considering the relationships within and between layers and this is represented using a mathematical model which captures the complex interactions or relationships that may exist among the same set of individuals. It consists of intralayer edges and interlayer edges. Here, intralayer edges indicate edges that connect nodes belonging to the same layer. Whereas interlayer edges connect nodes belonging to different layers.

As real-world networks possess varied relationships and interactions across numerous platforms, recent research activities have seen huge attention in community detection in *multilayer networks*. Here, we summarize the recent community detection approaches in *multilayer networks* with regard to the challenges in earlier multilayer community detection approaches, features considered for detecting communities, the developments in the recent approaches, the type of communities detected and the evaluation strategies adopted to analyze the quality and accuracy of communities in *multilayer networks* have been also been discussed in this section [31, 37, 40].

1. **Modularity-based Ensemble Multilayer Community Detection** ($M - EMCD$): It is a modularity-optimization-driven ensemble-based approach which incorporates *aggregation* of the community structures separately generated for each layer to multilayer community detection with the objective to detect consensus community structures. Some of the issues which the earlier approaches on consensus community detection had to deal with are internal connectivity, redundancy in terms of multilayer edges connecting different communities, etc. To deal with these issues, $M - EMCD$ had been introduced to incrementally refine the modularity of the consensus solution provided by the topological-lower-bounded baseline, until no further improvements are possible. The consensus community structure obtained after applying the $M - EMCD$ approach have undergone the following developments:

   – The prototypical community membership of nodes has been captured.
   – It preserves the multilayer topology information.
   – The edge-connectivity has been optimized via modularity analysis.
   – Even in the presence of disconnected components in the *multilayer graph*, same robustness has been achieved.
   – It scales well with the size of *multilayer network*.

   It is also to be mentioned here that $M - EMCD$ approach uses a set of layers for modeling multilayer graphs and it results in $non - overlapping$ communities. To satisfy the quality and accuracy of the detected communities, various evaluation metrics such as $Redundancy$, $MultilayerSilhoutte$, $Modularity$, NMI have been incorporated. In order to find the detailed information about the $M - EMCD$ approach, [33] may be referred.
2. **lcd**: This approach considers a *multilayer network* as a single relationship shared by a set of nodes. The previous multilayer community detection approaches ignored interplay between the layers or unique topological structure in a particular layer and most of them could only detect non-overlapping communities. Recently, a new algorithm, namely, *lcd* have been introduced which solves the community detection problem by exploiting the topological structure and interplay between layers in a *multilayer network* to detect overlapping communities. The *lcd* method have been summarized as follows [34]:

   – All layers are merged into a single layer using Eq. 7.3.

$$a_{ij} = \begin{cases} 1, \; \forall l \in L, \; \exists a_{ij}^{l} = 1 \\ 0, \quad else \end{cases} \tag{7.3}$$

   In Eq. 7.3, l refers to a layer, a_{ij} signifies adjacency matrix for unified network, a_{ij}^{l} refers to a adjacency matrix for layer l.
   – Edge-pairs (a node shared by two edges) that exist within or across layers are extracted.

– Then, the similarity of all edge-pairs is calculated based on *Jaccard similarity*.
– The edge-pair similarities are utilized to obtain hierarchical structures. Thereafter, maximizing the community density (evaluates the strength of connections between nodes in all layers) results in overlapping communities.

To evaluate the accuracy of the *lcd* approach, *recall rate*, *precision rate* have been used.

3. **MSH-LPA**: This is an *aggregation*-based community detection approach in *multilayer networks* based on improved label propagation algorithm (LPA). Earlier LPA-based community detection approaches had a lot of shortcomings like instability, high time complexity etc. The *MSH-LPA* algorithm have reduced the instability and time complexity issues by considering the *MSH-index* which captures the influence of nodes and it is a combination of *SH-index* of the node and weight of the node. Here, the influence of the node is based on the influence that comes from the neighbors of different layers. The *MSH-index* is defined as

$$MSH(i) = SH(i) + \sum_{j \in N(i)} \frac{w(i, j)}{\mid N(i) \mid'} \tag{7.4}$$

In Eq. 7.4, $SH(i)$ indicates the *SH-index* of node i, $w(i, j)$ indicates the influence of neighboring nodes from different layers, $\mid N(i) \mid'$ refers to degree of node i. Here, the second part of Eq. 7.4 is used to distinguish nodes with same $SH - index$. Thereafter, the order of the nodes are updated based on $SH - index$ and the order of the node-labels is updated based on the following formula:

$$l(i) = argmax_l \sum_{j \in N(i)} MSH(j) \tag{7.5}$$

This approach basically uses a graph model to model a *multilayer graph* and a key feature of this approach is the inclusion of both same-layer edge and latent-edge in the detection of overlapping communities. Eventually, this approach has been evaluated based on *modularity* metrics. For further details regarding the $MSH - LPA$ algorithm you can refer [37].

4. **MTCOV-EM**: This approach utilizes the topology of interactions and *node attributes* for detecting communities in *multilayer networks*. Earlier community detection approaches had to assume some priori information to detect communities. But the $MTCOV - EM$ algorithm detects communities without any priori correlation structure between attributes and communities but detects it from data. It shows the impact of node attributes in uncovering meaningful patterns. It is also to be added here that the inclusion of node information assists in the prediction of missing links or attributes which further leads to more interpretable community structures. Moreover, it is flexible in exploiting the attributes that are more correlated with the network communities. It also exploits the sparsity of datasets to several inference tasks. Here, we utilize EM algorithm for inference purposes. It assumes conditional independence between network and attribute variables and

hence, the total log-likelihood is decomposed into a sum of two terms as

$$\mathcal{L}(U, V, W, \beta) = \mathcal{L}_G(U, V, W) + \mathcal{L}_X(U, V, \beta) \tag{7.6}$$

In Eq. 7.6, the first term indicates the log-likelihood for structural dimension and the second term signifies log-likelihood for attribute dimension. However, the parameters' performance can be improved by balancing the contribution of the structure and node attributes based on the magnitude of each term. For representation of the network structure, the $MTCOV - EM$ algorithm uses adjacency tensor representation and ultimately the incorporation of this approach leads to overlapping communities. Evaluation metrics based on similarity score such as *F1-score, Jaccard similarity* have been used to analyze the correctness of this approach. For more details regarding this approach please refer [38].

5. **Possibilistic c-means clustering model considering two-layer graphs (PCMTL)**: This is a type-2 fuzzy community detection model in two-layer graphs that detects overlapping communities based on structural and attribute similarities. With the increasing popularity of social networks, it is evident that a node is attached with multiple attributes. The attributes that resembles closely with the interactions between nodes and related attributes to the new interaction are taken. Earlier type-1 fuzzy approach had to deal with uncertainty of parameters, degree of belonging were exact values. The *PCMTL* approach utilized the type-2 fuzzy sets to overcome the issues related with type-1 fuzzy approach. Here, the objective function is defined by

$$min J_m(\tilde{U}, v) = \sum_{i=1}^{c}\sum_{k=1}^{n}(\underline{u}_{ik}^{m})d_{ik}^{max} + \sum_{i=1}^{c}\sum_{k=1}^{n}(\bar{u}_{ik})^m d_{ik}^{min} \tag{7.7}$$

In Eq. 7.7, $\tilde{U}$ indicates type-2 fuzzy sets and $\underline{u}_{ik}$ indicates lower membership degree and $\bar{u}_k$ indicates upper membership degree value of node k, the term $m \in [1, \infty)$ indicates the fuzzy weighting exponent, considering the new interaction c indicates the number of clusters, d_{ik}^{min} and d_{ik}^{max} indicates the least and greatest distance between node k and center of community i. Here, the belongingness of a node to a community as regard to varied aspects of interactions in a two-layer graph is determined based on suggested values of interval type-2 membership.

This approach is based on two perspective. Firstly, more number of common attributes shared by two nodes implies higher probability for them to belong to the same cluster and secondly, if two-node belongs to the same community in higher number of layers, higher is the probability for those pair of nodes to belong to the same community in community detection with different interactions between those nodes. It shows the contribution of strong ties and weak ties in the belongingness of a node to a community. *This community detection approach is particularly beneficial when there are many uncertainties in the system.* Ultimately, the evaluation is performed using a new metric, namely, *validity index*. An elaborate illustration of this approach can be found at [26].

6. **ML@NetDE**: This is a differential evolution-based community detection approach which aims to find a partition of attributed *multilayer networks* based on a fitness function. Earlier community detection approaches in *multilayer networks* considered either the multiple level aspect or the attribute information. However, the *ML@NetDE* differential evolution-based community detection approach in *multilayer networks* is a combination of structural connectivity and node similarity. As this is a differential evolution-based approach, so the mutation and crossover operators, representation of the problem and fitness function to optimize are to be choosen. The fitness value between the mutant vector and target vector and is defined by

$$x_i^{t+1} = \begin{cases} u_i^t \ if \ cumd(u_i^t) \leq cumd(x_i^t) \\ x_i^t \qquad otherwise \end{cases} \tag{7.8}$$

In Eq. 7.8, u_i^t indicates the mutant vector and x_i^t indicates the target vector. The fitness function illustrates the importance of attributes and edges by optimizing simultaneously for all the layers, the connectivity between nodes belonging to the same community and the homogeneity of their features. But, if the expected fitness result is not obtained, then this approach continues with incrementing the upper bound progressively to allow more variation. Then, the obtained value is checked for merging which results in detection of accurate community structures in attributed *multilayer networks*. This approach utilizes the adjacency list representation to model *multilayer networks* and incorporates *NMI, Density, Entropy* for evaluation purpose. A detailed illustration of this approach is given in [39].
7. **Multilayer network local community detection (IMLC) based on influence maximization**: As the name implies, this is a multilayer network local community detection algorithm which exploits *influence relations* to detect communities in *multilayer networks* comprising of two or more layers. As mutual influence increases the complexity of the network, so the earlier approaches could not be successful in considering this aspect to detect multilayer communities. Recently, a local community detection approach, namely, $IMLC$ which is a combination of direct influence relation and indirect influence relation has been introduced to cover these complex issues and detect multilayer communities. Basically, local community detection begins with a selected seed node say v_0 and based on the incorporation of a certain classification algorithm, community structures are detected depending on the local information. Here, the selection of seed nodes holds a very important role because the speed of the spread of influence throughout the network depends on the influence of seed nodes.

   Seed node selection is a very difficult problem in multilayer complex network. Hence, this model has come up with a new idea to select nodes based on high node influence which indicates the strong ability of a node to influence the nodes that lies in its vicinity and is defined by

$$C(v) = \sum_{uin\tau(v)} \delta(u) \tag{7.9}$$

In Eq. 7.9, $\tau(v)$ refers to set of nodes v. After seed node selection, this approach goes through a merging process of seed node through impact measurement. Consecutively, local communities are obtained through influence measurement. The idea behind this algorithm is that high impact between nodes and their neighbors enhances their probability to belong to the same community. Ultimately, the accuracy of this approach is determined by *influence*. To gather more knowledge about this algorithm, please refer [35].

8. **Multilayer community detection (MLCD)**: Community detection in *multilayer networks* should be carried with the layers that have congested connections among the inherent nodes. It is assumed that a multilayer community comprises a set of closely connected nodes in a subset of layers. It can detect both disjoint and overlapping communities. This approach basically comprises two steps such as

   – **Multi-objective mathematical model**: A multi-objective mathematical model is proposed which considers the different aspects associated with a community and incorporates a genetic algorithm-based multi-objective optimization method to obtain the best partition of nodes with respect to the given layers in a *multilayer network*.
   – **Multilayer community detection** (*MLCD*): A novel algorithm, namely, *MLCD* have been utilized to detect the best multilayer overlapping communities using all the Pareto front optimal solutions.

   Finally, for evaluating the detected communities, evaluation metrics such as *NMI*, *multilayer modularity* have been incorporated. To get an in-depth understanding of the *MLCD* approach, please refer [36].

7.4.2 Community Detection in Multidimensional Networks

In a multidimensional network, the entities may be connected by multiple types of links. Here, the labels indicate the dimension of the network and the edges indicate the interactions or relationship between entities. As *multidimensional network* is a variant of multilayer network, and several approaches related to community detection in *multilayer networks* have already been discussed earlier, so we would detail in brief a few of the recent approaches of community detection in *multidimensional networks* due to space limitations. The criteria that were covered in illustrating the multilayer community detection approaches have been used to demonstrate the multidimensional community detection approaches

1. **CoMRCA**: This is an integration-based approach which incorporates three concepts such as object context, relation context, and the concept lattice family (CLF) to view the community detection problem across multiple network dimensions. In this scenerio, object context refers to a set of objects or entities of the same type, relation context refers to the set of interactions that takes place between the object contexts. Earlier approaches considered only on the topological properties

of the network and ignored the embedded semantic information. The *CoMRCA* approach overcomes the above-mentioned issues and exploits the topological properties and embedded semantic information (users' attributes) to detect multidimensional disjoint communities.

Here, multidimensionality indicates varied types of objects and relationships present in a heterogeneous network which is represented using Concept Lattice Family (CLF). At this point, CLF is constructed based on relational concept analysis (RCA) techniques. In the RCA technique, entities are defined by their respective attributes and their relationships with other entities. Hence, RCA is used for processing multirelational datasets.

CoMRCA is incorporated to explore the CLF and extract inherent multidimensional communities. To extract the varied relations and entities that are embedded in the structure of social networks across different network dimensions, a new algorithm, namely, *SearchCommunity* based on the RCA technique has been suggested. The input of the *SearchCommunity* algorithm is the combination of a CLF and a set of relations R which outputs a set of communities and the corresponding set of labels. Thereafter, the detected communities are evaluated using *Concept-Gain (CG), Recall, Precision, $F\beta$-score*. To find further details regarding this approach, please refer [8].

2. **Multidimensional community detection algorithm (MCDA)**: This is a parameterless approach which considers both the density and interaction between the nodes in order to detect multidimensional disjoint communities. Earlier approaches to multidimensional community detection were parameter-laden, lacked outlier detection mechanism, and failed to select dimensions associated with the detected communities in a systematic way. The *MCDA* approach has been introduced recently which overcomes the above-mentioned issues and effectively detects outliers, i.e., those nodes that deviate significantly from the densely connected nodes.

   Community in a multidimensional network may exist in different combinations of dimensions. Hence, predicting the relevant dimension and separating the irrelevant dimensions with respect to the detected communities is significant for better community detection. This approach is basically composed of two steps such as

   – *Outlier detection*: The outliers that lies within a community need to be systematically discriminated using a probabilistic approach. Firstly, we compute the strength of connection between two nodes say node u and node v using Eq. 7.10,

$$f(v,u) = \frac{|\ \eta(u, D(v,u)) \cap \eta(v, D(v,u))\ |}{|\ \eta(u) \cap \eta(v)\ |} \tag{7.10}$$

   In Eq. 7.10, $D(v,u) \sqsubseteq d$ refers to the subset of dimensions that connects nodes u and v and the term $\eta(u, D(v,u))$ signifies the set of neighbors of node u with respect to all the network dimensions d. Thereafter, an outlier score is estimated followed by automatic outlier detection function.

– *Mining multidimensional communities*: After the selection of outliers, MCDA adopts a propagation strategy that emphasizes upon the most frequently used interaction dimensions among neighbors as an additional constraint for membership function. Here, the propagation strategy is implemented based on the importance of dimensions to extract the best partition. After the communities have been recovered, this approach selects the relevant dimensions and eliminates the irrelevant dimensions.

After community detection, *NMI* have been incorporated to evaluate the accuracy of the *MCDA* algorithm. To get an elaborate description of the *MCDA* approach, please refer [19].

7.5 Evaluation Strategies

This section has been dedicated to various evaluation attributes to determine the performance of community detection algorithms in *multilayer networks* where each layer signifies a unique meaning. Two main attributes such as datasets and metrics have been discussed. Considering the above brief insights, various widely used evaluation metrics pertaining to measure both the accuracy and quality of identified communities are explained with their internal details [41, 42]. To measure accuracy, ground-truth communities are required but quality does not require it. Thus, specific datasets are necessary to evaluate depending on what perspective communities are to be evaluated and hence, we discuss two types of datasets such as synthetic networks and real-world networks.

7.5.1 Datasets

Community detection algorithms are evaluated on different datasets based on our requirement. In recent time, availability of rich datasets has increased broadly. We basically discuss synthetic multilayer networks and real-world multilayer datasets. A list of various multilayer datasets are available at http://dm.kaist.ac.kr/datasets/multi-layer-network/ and http://multilayer.it.uu.se/datasets.html.

7.5.1.1 Synthetic Multilayer Networks

The role of synthetic networks in the evaluation of different community detection approaches holds a very high importance because synthetic networks are generated with different adjustable parameters and hence, with a slight modification of certain parameters, we obtain a range of different parameters with different characteristics. This can assist in our understanding of the behavior of different algorithms on different networks. For example, the LFR network is a synthetic network which follows power law in degree distribution and community size. An advantage of these types of networks is its flexibility to allow overlapping communities which assists to model the LFR networks in terms of structural parameters with respect to real networks. Lancichinetti introduced a new method which runs the LFR benchmark algorithm on several times on the original graph and segments it in multiple layers which yields a multilayer synthetic network [43, 44]. Another type of synthetic network is the mLFR network [36].

7.5.1.2 Real-World Multilayer Datasets

Real-world multilayer networks can be broadly classified into three categories such as transportation networks, biological networks, and social network. These datasets inhibit different types of relationships and interactions between the same set of individuals. It contains various types of inherent communities across the layers and within the layers. To get a brief overview of the real-world multilayer datasets, we list below some of the popularly used datasets [13].

Biological General Repository for Interaction Datasets (BioGRID) 3.2.108: This is a multiplex network dataset which considers the different protein-protein interactions among 13 organisms and each of the layers correspond to a unique interaction such as, physical, co-localization, additive, synthetic genetic, direct, or association interaction. In total, there are 3–7 layers identified for each of the organisms [20].

ObamaInIsrael2013: President Obama's visit to Israel in the year 2013 has been modeled from Twitter as a three-layer graph comprising retweets, mentions, and replies. Here, "retweet" indicates that, if the information tweeted by a user is supported by its followers, then they can forward the tweet to its followers. Whereas, the term "mention" indicates that if a user has willingness to share a message with another user, they can mention the user in their tweet and the term "reply" indicates the exchange of messages between users in response to the tweet of a user [35].

Airline data set: This is a thirty-seven-layer airline dataset which represents a multilayer network. It is basically an airline facility operating in Europe where each layer corresponds to a different airline. It is composed of 450 nodes and 3,588 edges [45].

DBLP data set: This is a multilayer bibliographic dataset and each layer corresponds to the top 50 computer science conferences. This dataset comprises 1,08,030 nodes and 2,76,658 edges where a node indicates an author and an edge indicates the relationship shared between two authors if they co-authored at least one research paper or they share some research similar interest [35, 46–49].

Mobile Phone: This is a set of 200 mobile users dataset collected from Lausanne, Switzerland by Nokia Research Center(NRC) Lausnne representing a multilayer network. Each layer represents relationships based on physical locations, phone calls, and Bluetooth scans, respectively [50].

Citeseer: This is a multilayer dataset representing citation network of computer science publications. It comprises 3,312 nodes and 4,536 edges. Each layer represents the relationships that are formed due to citation or content similarity [51–53].

Flickr: This is a multilayer network representing social network dataset with tagged photos. There are 16,710 nodes and 716,063 edges. Nodes indicate user and an edge between a pair of nodes indicates the existence of a user in another users' contact list or their liking for the same images [52, 54].

Arxiv Publication Database: This bibliographic dataset represents a multilayer network where each layer corresponds to relationships that are formed due to citation with different research topics. Thus, the number of layers in this dataset is equal to the number of research topics [49].

Cora: This is another bibliographic multilayer dataset where each layer represent research fields such as natural language processing, data mining, and robotics, respectively [55].

7.5.2 Evaluation Metrics

As has been mentioned in Sect. 7.4, there does not exist any shared or universal accepted definition of a community. Hence, to evaluate the detected communities, a wide range of quality metrics and accuracy metrics have been introduced to solve the community detection problem from different perspectives. We discuss some representative metrics which have been classified into the following.

7.5.2.1 Accuracy Metrics

This metric is used to evaluate and compare the result of community detection algorithms on different datasets with given ground-truth communities. Suppose we have the ground-truth classes as $C = c_1, c_2, c_3, \ldots, c_k$ and the computed clusters as $\Omega = \{\omega_1, \omega_2, \ldots, \omega_k\}$. We list below some of the accuracy metrics as follows:

– *Purity*: This accuracy metric yields the percentage of the total number of nodes that have been categorized correctly and is defined as [56],

$$Purity(\Omega, C) = \frac{1}{N}\sum_k \max_j |\omega_k \cap c_j| \tag{7.11}$$

In Eq. 7.11, N refers to the total number of nodes inherent in the network and $\omega_k \cap c_j$ gives the number of nodes shared by ω_k and c_j.

– *Standard Mutual Information (NMI)*: This evaluation metric is particularly applicable for when we have the ground-truth information. The similarity between the two sets of nodes is calculated by NMI as [35, 57],

$$I(P_1, P_2) = \frac{-2\sum_{i=1}^{C_{S_1}}\sum_{j=1}^{C_{S_2}} N_{ij} log(\frac{N_{ij}n}{N_i N_j})}{\sum_{i=1}^{C_{S_1}} N_i log(\frac{N_i}{n}) + \sum_{j=1}^{C_{S_2}} N_j log(\frac{N_j}{n})} \tag{7.12}$$

In Eq. 7.12, symbols P_1, P_2 indicates partitions. N_{ij} signifies nodes shared by community i of partition P_1 and community j of partition P_2. $N = [N_{ij}]$ is the confusion matrix. The sum of row i and column j of the confusion matrix is indicated by N_i and N_j respectively. Whereas C_{S_1} and C_{S_2} indicate the number of partitions of P_1 and P_1 and number of nodes is indicated by n.

– *Precision*: It is a measure of accuracy for community detection algorithms which is defined as [58]

$$Precision = \frac{TP}{TP + FP} \tag{7.13}$$

In Eq. 7.13, TP=true positive, FP=false positive.

– *Recall*: It is defined as [58],

$$Recall = \frac{TP}{TP + FN} \tag{7.14}$$

The symbols TP and FN indicate true positive and false negative, respectively.

– *F1-score*: This metric requires prior information about partition labels. It is a combination of *precision* and *recall* and is defined as [58],

$$F1 - score = \frac{2 \times (Precision \times Recall)}{Precision + Recall} \tag{7.15}$$

7.5.2.2 Quality Metrics

In contrast with the accuracy metrics, quality metrics does need the ground-truth information. This metric is mainly used to check the structural feasibility of the communities within the network. Communities that are detected by various community detection algorithms are structurally feasible if nodes within communities are densely connected in comparison to nodes outside communities. We have discussed some of the quality metrics in this section.

Modularity: It is widely used to evaluate the goodness of detected community by considering that the number of edges internal to a community should be greater than that in a random graph with similar degree distribution [59, 60]. Girvan and Newman's modularity function is used to evaluate the quality of community partitions which is defined as [61]

$$Modularity, Q = \frac{1}{2m}\sum_{ij}\left(A_{ij} - \frac{k_i k_j}{2m}\right)\delta(C_i, C_j) \tag{7.16}$$

In Eq. 7.16, m signifies the number of edges, The element of the adjacency matrix is denoted by A_{ij}, if i and j belongs to the same community $\delta(C_i, C_j{=}1)$, else 0.

Modularity density: This community evaluation metrics is used to eliminate the community evaluation problem which is defined as [62]

$$D = \sum_{i=1}^{c} \frac{L(V_i, V_i) - L(V_i, \bar{V_i})}{\mid V_i \mid} \tag{7.17}$$

In Eq. 7.17, symbol c indicates the number of communities,$\mid V_i \mid$ indicates node in the ith community, $L(V_i, \bar{V_i}) = \sum_{j \in V_i, k \in \bar{V_i}} A_{jk}$ signifies the number of connections shared between ith community and other communities. Whereas, $L(V_i, V_i) = \sum_{j \in V_i, k \in V_i} A_{jk}$ indicates the number of edges among ith community.

7.6 Conclusion

Complex networks have managed to gather extensive attention from the scientific community. Our study suggests that in recent times, a plethora of promising approaches have been introduced to detect communities in *multilayer networks*. Our motivation to study the different community detection approaches has been due to the vital role that communities play in characterizing the structure and function of the multilayer system under consideration. Further studies have triggered the need of *multilayer networks* to model real-world interactions and relationships. A key feature of the *multilayer networks* lies with its flexibility to incorporate different types of data in a single structure and thus, the different aspects of relationships/interactions are covered by these networks. It is also to be mentioned here that in a *multilayer network*, edges may exist in different layers/dimensions signifying different types of relationships/interactions but they may be potentially related and hence, considering these networks helps to detect better communities. Also, *multilayer network* are of different types such as directed, undirected, weighted, unweighted, static, dynamic and there are several other unique features and different systematic methods of representing networks. Basically, our objective to discuss different approaches has been to summarize the latent interlayer and intralayer community structures. After incorporation of the community detection algorithm, the result generated is evaluated and compared for accuracy and quality of detected communities. To support these accuracy and quality evaluation, several datasets and evaluation metrics for *multilayer networks* have been discussed in this chapter.

References

1. Luo, L., Liu, K., Guo, B., Ma, J.: User interaction-oriented community detection based on cascading analysis. Inf. Sci. **510**, 70–88 (2020). Elsevier
2. Amelio, A., Pizzuti, C.: A cooperative evolutionary approach to learn communities in multilayer networks. In: Parallel Problem Solving from Nature –PPSN XIII, pp. 222–232. Springer, Cham (2014)
3. Tang, L.,Wang, X., Liu, H.: Uncovering groups via heterogeneous interaction analysis. In: 9th IEEE International Conference on Data Mining (ICDM), pp. 503–512 (2009)
4. Li, Z., Pan, Z., Zhang, Y., Li, G., Hu, G.: Efficient community detection in heterogeneous social networks. Math. Probl. Eng. (2016)
5. Tang, L., Wang, X., Liu, H.: Community detection via heterogeneous interaction analysis. Data Min. Knowl. Discov. **25**(1), 1–33 (2012)
6. Fortunato, S.: Community detection in graphs. Phys. Rep. **486**(3–5), 75–174 (2010)
7. Tang, L., Liu, H.: Community detection and mining in social media. Synth. Lect. Data Min. Knowl. Discov. **2**(1), 1–137 (2010)
8. Guesmi, S., Trabelsi, C., Latiri, C.: Multidimensional community discovering in heterogeneous social networks. Concurr. Comput.: Pract. Exp. e5809 (2020)
9. Interdonato, R., Magnani, M., Perna, D., Tagarelli, A., Vega, D.: Multilayer network simplification: approaches, models and methods. Comput. Sci. Rev. **36**, 100246 (2020)
10. Kazienko, P., Musial, K., Kukla, E., Kajdanowicz, T., Bródka, P.: Multidimensional social network: model and analysis. In International Conference on Computational Collective Intelligence, pp. 378–387. Springer, Berlin (2011)
11. Cai, D., Shao, Z., He, X., Yan, X., Han, J.: Mining hidden community in heterogeneous social networks. In: Proceedings of the 3rd International Workshop on Link Discovery, pp. 58–65 (2005)
12. Hammoud, Z., Kramer, F.: Multilayer networks: aspects, implementations, and application in biomedicine. Big Data Anal. **5**(1), 1–18 (2020)
13. Cozzo, E., de Arruda, G.F., Rodrigues, F.A., Moreno, Y.: Multilayer networks: metrics and spectral properties. In: Interconnected Networks, pp. 17–35. Springer, Cham (2016)
14. Bródka, P., Chmiel, A., Magnani, M., Ragozini, G.: Quantifying layer similarity in multiplex networks: a systematic study. R. Soc. Open Sci. **5**(8), 171747 (2018)
15. De Domenico, M., Solé-Ribalta, A., Cozzo, E., Kivelä, M., Moreno, Y., Porter, M.A., Gómez, S., Arenas, A.: Mathematical formulation of multilayer networks. Phys. Rev. X **3**(4), 041022 (2013)
16. Singh, H., Sharma, R.: Role of adjacency matrix and adjacency list in graph theory. Int. J. Comput. Technol. **3**(1), 179–183 (2012)
17. Kivelä, M., Arenas, A., Barthelemy, M., Gleeson, J.P., Moreno, Y., Porter, M.A.: Multilayer networks. J. Complex Netw. **2**(3), 203–271 (2014)
18. Wasserman, S., Faust, K.: Social Network Analysis: Methods and Applications, vol. 8. Cambridge University Press (1994)
19. Chouchane, A., Boutemine, O., Bouguessa, M.: On detecting multidimensional communities. In: IEEE/ACM International Conference on Advances in Social Networks Analysis and Mining, pp. 45–78. Springer, Cham (2019)
20. De Domenico, M., Nicosia, V., Arenas, A., Latora, V.: Layer aggregation and reducibility of multilayer interconnected networks (2014). arXiv:1405.0425
21. Porter, M.A.: What is... a multilayer network. Not. AMS **65**(11) (2018)
22. Berlingerio, M., Coscia, M., Giannotti, F., Monreale, A., Pedreschi, D.: Foundations of multidimensional network analysis. In: 2011 International Conference on Advances in Social Networks Analysis and Mining, July 2011, pp. 485–489. IEEE (2011)
23. Dunlavy, D.M., Kolda, T.G., Kegelmeyer, W.P.: Multilinear algebra for analyzing data with multiple linkages. In: Kepner, J., Gilbert, J. (eds.) Graph Algorithms in the Language of Linear Algebra. Fundamentals of Algorithms, pp. 85–114. SIAM, Philadelphia (2011)

24. Wang, P., Robins, G., Pattison, P., Lazega, E.: Exponential random graph models for multilevel networks. Soc. Netw. **35**(1), 96–115 (2013)
25. Dimitrova, T., Kocarev, L.: Graphical models over heterogeneous domains and for multilevel networks. IEEE Access **6**, 69682–69701 (2018)
26. Naderipour, M., Zarandi, M.H.F., Bastani, S.: A type-2 fuzzy community detection model in large-scale social networks considering two-layer graphs. Eng. Appl. Artif. Intell. **90**, 103206 (2020)
27. Granovetter, M.: The strength of weak ties: a network theory revisited. Sociol. Theory **201–233** (1983)
28. Kadushin, C.: Understanding Social Networks: Theories, Concepts, and Findings. Oup Usa (2012)
29. De Bacco, C., Power, E.A., Larremore, D.B., Moore, C.: Community detection, link prediction, and layer interdependence in multilayer networks. Phys. Rev. E **95**(4), 042317 (2017)
30. Coscia, M., Giannotti, F., Pedreschi, D.: A classification for community discovery methods in complex networks. Stat. Anal. Data Min.: ASA Data Sci. J. **4**(5), 512–546 (2011)
31. Huang, X., Chen, D., Ren, T., Wang, D.: A survey of community detection methods in multilayer networks. Data Min. Knowl. Discov. **1–45** (2020)
32. Negara, E.S., Andryani, R.: A review on overlapping and non-overlapping community detection algorithms for social network analytics. Far East J. Electron. Commun. **18**(1), 1–27 (2018)
33. Tagarelli, A., Amelio, A., Gullo, F.: Ensemble-based community detection in multilayer networks. Data Min. Knowl. Disc. **31**(5), 1506–1543 (2017)
34. Liu, W., Suzumura, T., Ji, H., Hu, G.: Finding overlapping communities in multilayer networks. PloS One **13**(4), e0188747 (2018)
35. Li, X., Xu, G., Lian, W., Xian, H., Jiao, L., Huang, Y.: Multi-layer network local community detection based on influence relation. IEEE Access **7**, 89051–89062 (2019)
36. Shahmoradi, M.R., Ebrahimi, M., Heshmati, Z., Salehi, M.: Multilayer overlapping community detection using multi-objective optimization. Future Gener. Comput. Syst. **101**, 221–235 (2019)
37. Chen, D., Du, P., Jiang, Q., Huang, X., Wang, D.: A feasible community detection algorithm for multilayer networks. Symmetry **12**(2), 223 (2020)
38. Contisciani, M., Power, E., De Bacco, C.: Community detection with node attributes in multilayer networks (2020). arXiv:2004.09160
39. Pizzuti, C., Socievole, A.: A differential evolution-based approach for community detection in multilayer networks with attributes. In: International Conference on Database and Expert Systems Applications, September 2020, pp. 250–265. Springer, Cham (2020)
40. Vaiana, M., Muldoon, S.: Resolution limits for detecting community changes in multilayer networks (2018). arXiv:1803.03597
41. Biswas, A., Biswas, B.: Defining quality metrics for graph clustering evaluation. Expert Syst. Appl. **71**, 1–17 (2017)
42. Creusefond, J., Largillier, T., Peyronnet, S.: On the evaluation potential of quality functions in community detection for different contexts. In: International Conference and School on Network Science, January 2016, pp. 111–125. Springer, Cham (2016)
43. De Domenico, M., Lancichinetti, A., Arenas, A., Rosvall, M.: Identifying modular flows on multilayer networks reveals highly overlapping organization in interconnected systems. Phys. Rev. X **5**(1), 011027 (2015)
44. De Domenico, M., Porter, M.A., Arenas, A.: MuxViz: a tool for multilayer analysis and visualization of networks. J. Complex Netw. **3**(2), 159–176 (2015)
45. Cardillo, A., Gómez-Gardenes, J., Zanin, M., Romance, M., Papo, D., Del Pozo, F., Boccaletti, S.: Emergence of network features from multiplexity. Sci. Rep. **3**(1), 1–6 (2013)
46. Boden, B., Günnemann, S., Hoffmann, H., Seidl, T.: Mining coherent subgraphs in multi-layer graphs with edge labels. In: Proceedings of the 18th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, August 2012, pp. 1258–1266 (2012)
47. Silva, A., Meira Jr, W., Zaki, M. J.: Structural correlation pattern mining for large graphs. In: Proceedings of the Eighth Workshop on Mining and Learning with Graphs, July 2010, pp. 119–126 (2010)

48. Xu, Z., Ke, Y., Wang, Y., Cheng, H., Cheng, J.: A model-based approach to attributed graph clustering. In: Proceedings of the 2012 ACM SIGMOD International Conference on Management of Data, May 2012, pp. 505–516 (2012)
49. Zhou, Y., Cheng, H., Yu, J.X.: Graph clustering based on structural/attribute similarities. Proc. VLDB Endow. **2**(1), 718–729 (2009)
50. Kiukkonen, N., Blom, J., Dousse, O., Gatica-Perez, D., Laurila, J.: Towards rich mobile phone datasets: Lausanne data collection campaign. In: Proceedings of ICPS, Berlin, vol. 68 (2010)
51. Li, H., Nie, Z., Lee, W. C., Giles, L., Wen, J.R.: Scalable community discovery on textual data with relations. In: Proceedings of the 17th ACM Conference on Information and Knowledge Management, October 2008, pp. 1203–1212 (2008)
52. Ruan, Y., Fuhry, D., Parthasarathy, S.: Efficient community detection in large networks using content and links. In: Proceedings of the 22nd International Conference on World Wide Web, May 2013, pp. 1089–1098 (2013)
53. Silva, A., Meira, W., Jr., Zaki, M.J.: Mining attribute-structure correlated patterns in large attributed graphs (2012). arXiv:1201.6568
54. Qi, G.J., Aggarwal, C.C., Huang, T.: Community detection with edge content in social media networks. In: 2012 IEEE 28th International Conference on Data Engineering, April 2012, pp. 534–545. IEEE (2012)
55. Dong, X., Frossard, P., Vandergheynst, P., Nefedov, N.: Clustering with multi-layer graphs: a spectral perspective. IEEE Trans. Signal Process. **60**(11), 5820–5831 (2012)
56. Zhao, Y., Karypis, G.: Empirical and theoretical comparisons of selected criterion functions for document clustering. Mach. Learn. **55**(3), 311–331 (2004)
57. Danon, L., Diaz-Guilera, A., Duch, J., Arenas, A.: Comparing community structure identification. J. Stat. Mech: Theory Exp. **2005**(09), P09008 (2005)
58. Cazabet, R., Chawuthai, R., Takeda, H.: Using multiple-criteria methods to evaluate community partitions (2015). arXiv:1502.05149
59. Chakraborty, T., Dalmia, A., Mukherjee, A., Ganguly, N.: Metrics for community analysis: a survey. ACM Comput. Surv. (CSUR) **50**(4), 1–37 (2017)
60. Arasteh, M., Alizadeh, S.: A fast divisive community detection algorithm based on edge degree betweenness centrality. Appl. Intell. **49**(2), 689–702 (2019)
61. Newman, M.E., Girvan, M.: Finding and evaluating community structure in networks. Phys. Rev. E **69**(2), 026113 (2004)
62. Li, Z., Zhang, S., Wang, R.S., Zhang, X.S., Chen, L.: Quantitative function for community detection. Phys. Rev. E **77**(3), 036109 (2008)

Chapter 8
Viral Marketing: A New Horizon and Emerging Challenges

Chitra Krishnan, Mubashir Majid Baba, Gurinder Singh, and Jasmine Mariappan

Abstract While companies have long been using electronic processes, the Internet and other digital technologies are packed with new and innovative ways to give customers more value. This challenge not only illustrates the basic concepts of traditional marketing but also modern marketing practices. It is important that businesses connect with customers more likely to gain their attention. It is also important for an advertisement campaign to be effective in being able to engage, surprise and enjoy users in promoting a brand or products spontaneously and induce word of mouth. For this reason, companies use a variety of communication methods, in particular online communication and digital marketing. Presently, marketing environment was revolutionised by the introduction of the internet and modern technology. In a new age of communication technology, the emphasis is evolving on marketing and consumer behaviour. One of the many marketing concepts is the era of interruption marketing, defined by all ads that capture the attention of the consumer as he does something else that comes to an end when new trends appear. Viral marketing can be one of the most popular breakdown marketing. Viral marketing is a marketing form that contacts users, in particular, through word of mouth on the Internet. Viral marketing has become more and more relevant in recent decades as social media networks such as Facebook, YouTube and Twitter are growing. The nature of this study is theoretical and covers related theories and literature. In this respect, the aim of the paper is to discuss the opportunities and challenges for marketers faced in a current environment presented by viral marketing. In a new age of communication technology, the emphasis is evolving on marketing and consumer behaviour. Diverse factors that directly or indirectly influence viral marketing are also discussed. Following the principles of viral marketing and how people communicate, a comprehensive

C. Krishnan (✉) · G. Singh
Amity International Business School, Amity University, Noida, India

M. Majid Baba
Department of Management Studies, University of Kashmir, Srinagar, Jammu and Kashmir, India
e-mail: mubashirbaba@rediffmail.com

J. Mariappan
Department of Business Studies, University of Technology and Applied Sciences, Ibra, Sultanate of Oman

A. Biswas et al. (eds.), *Principles of Social Networking*,
Smart Innovation, Systems and Technologies 246,
https://doi.org/10.1007/978-981-16-3398-0_8

analysis of key issues in the nature of viral marketing strategies will be introduced. This will be followed by an evaluation of the relevant subjects and possible risks. Finally, there will be a brief answer to the question of whether viral marketing is a critical new dimension to current century marketing.

Keywords Word of mouth · Viral messages · Social network · Internet · Technology · Digital marketing

8.1 Introduction

In the twenty-first century, marketing landscape was revolutionised by the introduction of the internet and modern technology. Technological developments are attributed to the introduction of the Internet in the first place. After the initial stages of its growth, which generated exuberant excitement and exaggerated expectations among companies as well as between consumers, it has grown into a large number of companies' distribution and communication channels. As an important part of many advertising events, the Internet now plays a big part. As the majority of people's lives is dominated by information technology. Our life has been very straightforward thanks to modern networking and Internet technology. For our everyday work or home like official, entertaining, educational or personal purposes, we use technology. Customers are well-trained today everyone likes to know about the products before going to the shop. This information can be obtained from the Internet or its social network. It's also really easy to buy a product from anywhere online.

The digital marketers are facing a great challenge. In today's online and product-saturated environment, they must find ways to make their product or service stand out. However, it is much simpler than done to start a viral marketing campaign. Viral ads are the most unforgettable. Viral marketing can be defined as a short time distributing or transferring information about a brand or product amongst mass media. It has done a great deal and led to the development of its company through thousands of marketers. And in today's digital age, people are highly social media immune, making it quite convenient. What viral marketing sounds like is transmitted like a virus worldwide. This basic approach will lead to a lot of positive results in promotional products. It leads to enormous ads, exposure and traffic and to increased turnover. Viral marketing in all its types refers to word of mouth, while for consumers it is a big practice. In a book, Philip Kotler and Marketing Moves said markets shift more rapidly than marketing. The conventional marketing paradigm must be fitted for the future. Commercialisation must be redefined and expanded. Marketing would not succeed, unless it is only responsible for driving up current goods sales [1].

With the rapid development of the Internet and other communication networks, a new room for word-of-mouth (WoM) communication was opened [2]. Viral marketing is WOM, a word focused on which communication and dissemination can be viewed. The term viral describes a form of advertisement that contains a customer message that is sent like a rampant influenza virus from one consumer to the next [3].

A major problem in viral marketing is the lack of control of the company's message: the more marketing tasks are delegated to customers the more control the company loses over its own camp designs and execution. This is a risky choice that could have unintended adverse effects. The goal of marketers engaged in successful viral marketing programs is to produce viral messages that are targeted at individuals that are very likely to be viewed and transmitted within a short time by these persons and their competitors in their communications.

Much research in the area of viral marketing is being carried out. Awareness, access to marketing strategies, interest and expertise that settle on their final decision are the big drivers behind viral marketing. The customer now has taken an observable phase; a product or service purchase or inventions sustained acceptance. One additional factor in this chain of viral marketing drivers is to check whether daily internet access substantially affects users in order to obtain an experience in viral marketing.

Viral marketing is a tactic that allows individuals to exchange marketing messages on the internet. It is referred to as viral marketing as it spreads like a virus. Messages about the product and its brands or services are sent over the Internet to a prospective customer. This potential purchaser transfers this information to another potential purchaser in a way that generates a wide network. What has a marketing virus to do? Any technique that encourages people to pass on a marketing communication to others, creating the potential for exponential growth in message visibility and impact, is characterised by viral marketing. Such techniques, like viruses, use the rapid replication to burst the message into the millions.

Viral marketing was founded in 1997 and Hotmail, a Free E-mail Service, was set up by Draper Fisher and Jurvetson. Hotmail was founded the year before by Sabeer Bhatia and Jack Smith, offering a great product: free internet e-mail. Tim Draper has come up with a simple but compelling idea to find a cheap way to advertise the commodity. Hotmail has registered more than 12 million users in the first 1.5 years. However, from launch to 12 million subscribers, Hotmail spent less than 500,000 on advertisement, printing and promotion. Hotmail has expanded exponentially faster than ever before, on-line, online or in print, in world history than any other company. In order for viral marketing to succeed, the marketing professors Andreas Kaplan and Michael Haenlein said the right message to be delivered to the right messengers in the appropriate environment. Viral marketing has seen immense development in many ways both off-line and on-line since the introduction of Hotmail. This special marketing strategy has been given many slightly different meanings [4].

After the popular Burger King, Hotmail and Procter and Gambler viral marketing campaign, several companies jumped on the bandwagon. Instantaneous movement can be created by viral campaigns and effectively promote brands, products and service [5]. Social networking growth made a major contribution to viral marketing's performance [6]. Since 2009, at least two-thirds of the world's population on the Internet visit a social networking or blog website every week. Over 1 billion active users are on Facebook alone. Time spent on social media platforms in 2009 started to overrun e-mail. A 2010 study showed 52 of people who look at news online via social networks, e-mails or posts [7].

Viral marketing is one of the most important strategies businesses must concentrate on when selling their products and services. Viral marketing is important to maintain a consistency of its goods and services in times of crisis and viral marketing depends on the strength of the organisation's social relationships with clients [8]. The higher the public's confidence in the organisation, the better it succeeds in viral marketing.

Unlike conventional marketing that is limited to particular markets and that the marketer manages and promotes, the viral marketing is consumer-driven marketing. Viral Marketing is the marketer's initiative to involuntarily make you the advertiser. The customer will become more like an advertising agency or a creative agency that is experienced in creating and transmitting marketing messages through the Internet and, if so, an agent of the virus, and you will have the motivating qualities to enable the receiver to transmit his message [9].

Many elements are critical and can become viral for a successful marketing campaign. First and foremost, products or services should be 'free' and readily accessible to all. This helps the marketing team achieve broad user awareness around the globe. The second element is the message's interesting and smart positioning. The user has to be able to accept the message. If your message or product is endorsed by a well-known person, on their website, social media and network pages, it may become viral. The next big aspect is the transfer function, so it is simple to transfer or exchange the message through e-mail, WhatsApp, networking websites, etc. The message should be both short and easy to understand.

The most famous video clips reach millions of viewers with funny pet videos or other YouTube clips. Other viral techniques include e-books, social networking apps such as fun but zap productivity games on Facebook, and some minor features including signing e-mails. Anything you can do to get a post and merge it to make it 'viral', as it is often demonstrated by viral marketing examples. The content can help the company brand or simply generate the desired excitement of its customers. The Internet makes this approach easier, but still, fierce rivalry. In certain organisations, viral marketing models alone are used. Email provider Hotmail could give the best example, including other ideas to help you develop your own viral marketing model. In today's global e-commerce society, where everyone wants to know more quickly and to connect with others through technological means such as mobile phones, Internet, e-mails, sms services or other formats, the incorporation of technology into the marketing arena is crucial.

Although electronic processes have long been used by businesses, the Internet and other emerging technology have created a flood of new and exciting ways to provide consumers with more value and better. With a view to business growth, marketers should be mindful that technologies can quickly create new consumers as well as new means to serve these markets. In addition to providing new avenues for information delivery markets, advanced technology has made it possible for customers to disseminate information digitally or through other digital channels that challenge conventional marketing practice since marketers want as many people as possible to get their names and content.

At the same time, customers have become more smart with the use of the Internet; they understand that more knowledge is placing them in a substantially better buying position: "Buyers are today in a matter of seconds able to compare prices and product characteristics [10]. They are just a few clicks from comparing the prices of rivals and can even specify the price they want to pay for their hotel room, plane tickets or mortgages to see if a voluntary provider is answerable. According to Statistica, there are 2.5 billion users worldwide on social media, and an article in Forbes by Robert Wynne estimates that about 5 billion content is published on Facebook and over 500 thousand tweets are sent daily. Three specific types of messengers are needed, business mavens, social hubs and sellers, in order to turn the ordinary message into a viral message. Business mavens are individuals who are on a pulse all the time. They are also among those who announce the message and send it to an immediate social network. Social centres are people whose social networks are incredibly complex, many know hundreds and may function as ties or bridges through different sub-cultures. Vending personnel should expect the message from the maven market, reinforced and reassured and distributed to the social hub to be disseminated further. Market mavens cannot be particularly convincing when transmitting details [11].

8.2 Traditional Versus Viral Marketing

Marketing is an integral aspect of any organisation that cannot be overlooked. Business owners often ask themselves this burning question – what forms of marketing can I use to drive my customers and sales? This is the exact point for most business owners where the battle against traditional forms of marketing and Viral marketing begins. Traditional marketing means any form of promotion, advertisement or campaign which has been in use for years by businesses and which has an established rate of success. Traditional marketing is the traditional strategies used since the idea of advertising or marketing was implemented. Traditional marketing is a very large variety of commercials and marketing. Newspaper, posters, radio, television, advertisement by roads and highways as well as magazine advertisements may be the most common methods of traditional marketing.

The purpose of the viral marketing campaign is not just to post daily advertising online, as opposed to conventional Marketing. The aim is to construct an informative message that references your product even if it doesn't. This message is so interesting that the viewer will be forced to share it with their friends and to spread the word about your product [12].

The biggest difference is definitely how communication takes place. There is no doubt the world is now a highly digital. We conduct many of our daily practises, not just digital magazines, but banking, shopping and online networking. Investing in the digital marketing campaign also seems to be a rational thought because of the rise of the digital age. While conventional marketing still retains a role, it steadily declines in society today. Instead of taking a page in an ordinary document to discuss your business, it is necessary for the businesses of today to have a website and to use the Internet for contact with customers.

The difference between a viral campaign and a traditional marketing campaign is in particular, how the message is delivered. A traditional marketing strategy aims to figure out the target audience, ensuring their advertisements are shown there. Examples are fairly straightforward: a nearby lawn mowing service at the neighbourhood store with a flyer, Make up ads in fashion magazines etc.

Viral marketers also argue that the viral ads are primarily for branding and that their content needs to be drastically different from traditional ones. However, this distinction is becoming less and less apparent with the variety of approaches that traditional marketing provides. As several TV ads have been shown by YouTube hits, conventionally-designed marketing can be just as shareable as viral marketing. Furthermore, what becomes viral is very difficult to predict, making it risky to make an advertiser the only target in a specific campaign becoming viral[13].

Viral marketing is important to keep up to date in today's world. Contrary to a conventional marketing massage, viral commercialization allows the consumer to buy what you sell on-site just following a viral massage. This is achieved by simply placing an online shopping link with your viral massage somewhere. In traditional marketing this is never that easy.

The biggest difference is definitely how communication takes place. There is no doubt the world is now a highly digital. We conduct many of our daily practises, not just digital magazines, but banking, shopping and online networking. Investing in the digital marketing campaign also seems to be a rational thought because of the rise of the digital age. While conventional marketing still retains a role, it steadily declines in society today. Instead of taking a page in an ordinary document to discuss your business, it is necessary for the businesses of today to have a website and to use the Internet for contact with customers.

The difference between a viral campaign and a traditional marketing campaign is in particular, how the message is delivered. A traditional marketing strategy aims to figure out the target audience, ensuring their advertisements are shown there. Examples are fairly straightforward: a nearby lawn mowing service at the neighbourhood store with a flyer, Make up ads in fashion magazines etc.

Viral marketers also argue that the viral ads are primarily for branding and that their content needs to be drastically different from traditional ones. However, this distinction is becoming less and less apparent with the variety of approaches that traditional marketing provides. As several TV ads have been shown by YouTube hits, conventionally-designed marketing can be just as shareable as viral marketing. Furthermore, what becomes viral is very difficult to predict, making it risky to make an advertiser the only target in a specific campaign becoming viral.

Viral marketing is important to keep up to date in today's world. Contrary to a conventional marketing massage, viral commercialization allows the consumer to buy what you sell on-site just following a viral massage. This is achieved by simply placing an online shopping link with your viral massage somewhere. In traditional marketing this is never that easy. Traditional marketing and viral marketing have both their own benefits and drawbacks still viral marketing takes over traditional marketing strategies.

8.3 The Viral Marketing Strategy: Word of Mouth

Viral marketing is considered a significant electronic leeway of WoM communication, involving the idea that news, information or entertainment is passed on or referred to another user. This approach is in line with recent customer behaviour trends. Indeed, customers now tend to trust the words of their peers rather than the business. They are no longer convinced by the company's traditional, one-straight promotional campaign: they seek guidance, analysis, comment and ranking from their peers before taking the purchase decision. In this process, blogs, forums, review pages and social networks are critical and the key recipient for knowledge and confidence.

They expect the company to sell its product in a desperate attempt. They want to hear however the 'real' side of the storey from people like them who already have the service/product and who are able to give an unbiased opinion. One should trust the knowledgeable consumer, for he obviously has relevant information and his information is most definitely personalised and pertinent. WoM references are now more effective and common than ever, which contribute to a reduced risk of incorrect purchasing decisions and the likelihood of saving time and money.

WoM only means that informal and ad hoc contact between people in the field of goods and services is often regarded as the dominant force on the market where information is filtered out and shared with friends and families treated as free, unbiased sources of counsel. In a seminal systematic study, Katz and Lazars Feld found WoM to be the key source of effect on the purchase of household goods. It is seven times more profitable than journals and magazines, four times more effective than personal sales and two times more successful than radio advertising for evolving consumer products [14].

A single suggestion can have a much more significant effect today in the hyper-connected world—contributing to WoM marketing or WoM advertisement campaigns to take advantage of this potential. Many best practices and marketing strategies promote natural words, but campaigns, especially in social media, can specifically seek to foster the social exposure of an online company. Nielsen states that 92% of people trust some other kind of marketing suggestions made by friends and families. The effectiveness of even academic research on WOM has been proved.

8.4 Characteristics of Viral Marketing Headings

In addition, different characteristics of viral marketing can be extracted from the essential factors that contributed to the success of Hotmail. First of all, it is obvious that viral marketing is closely related to conventional references. The innovation, however, is that information is spread primarily by e-mail and the Internet and not face to face. The unbelievable speed and the number of users, potentially obtained via viral marketing, often called the 'word of the mouth. This not only demonstrates the correctness of the long-running marketing statement of 'content is king' but also people have an authentic cause to express the message because they themselves will benefit from an expanding user base.

Innovative, creative concepts and unique ways of introducing a business or product to the customer are some of the main characteristics of viral messages. Modern messages in marketing must be entertaining, amusing, provocative, even surprising to get customers to share them. As examples of effective viral campaigns show, viral messages must have some difference and quite contentious that increases consumers' level of contact.

There are three common things in all viral marketing examples—intentional or unintended— the message, the message and the environment. In order to generate an effective viral marketing campaign, every part should be exploited. Any size of organisation may build viral marketing campaigns and they may stand up alone or participate in a larger conventional campaign. The campaigns themselves may use a variety of resources, such as videos, games, photographs, email, text messaging, free products, which appeal to users' or viewers' emotions, raise awareness and encourage consumption, sharing and sharing of such products, ideas and media. Viral marketing also depends on the assistance of an influence with a wide network of followers.

8.5 Benefits and Risk of Viral Marketing

Some of the reasons why viral marketing is a really powerful tool (some of them are inexpensive, fast and self-sufficient) and are pushing it to the top preferred marketing strategies currently in use by most businesses [5]. In addition, web-based viral marketing is made even better by increasing its effects. Currently, a message can be spread more reliably and more rapidly on the Internet than ever before. At this time, both companies and the audience are very excited about this strategy. It has its dark side, however. Many of the threats involved and possible harm caused by a viral marketing campaign.

Each internet marketer dreams of go viral' and finds the sweet spot that causes the internet population to repeatedly hit the share button. But it's not especially easy and while many of you think it's predictable with preparation and a formulation, this only holds true up to a certain extent. The types of content that are likely to go viral can be created and some online outlets are very good at it but the collective imagination is never guaranteed. Although marketers should ask themselves why they want to become viral and why the content they create with the aim is compatible with business goals before they start creating posts. They can also take 100,000 views on YouTube, but without any share and without any form of strategy that will not make the contents of their audience relate to brand, all of which generate content very well that makes public enjoyable and fun.

The advantages of such marketing can seem obvious. You can easily and at a reasonably low cost meet a large number of people. Instead of being something you have to handle and sell themselves, much of the interaction is done by clicks and share of the audience. In a way that is very rarely seen with any other marketing process, a viral promotion actually starts may lead to rapid success. Even if you do not instantly see results, this form of marketing is worth investigating the amount of visibility you can gain compared to the cost. In general, if you hear a product or service from people nearby you will usually get through the stuff. You will end up using the product or service to add more people and get used to it to the list of valuable customers. Company should concentrate on the right platforms and balance the content to increase the popularity of brands and penetrate your target market.

Another advantage is that mass media attention is increasingly possible. Even if this is not necessary for a growing business, the impact this can have is undeniable. Media outlets build brand awareness that always gives the brand credibility. If you are interested in reporting on-trend topics you can also benefit from ads without having to pay. The prospects of growth in viral marketing are faster and fulfilled in minimum time. Companies need time to evolve and create, and if this technology is widely used then the sky is at its limit. Viral marketing ideas are also used in modern times. Companies will put the brand on a competition podium by incorporating strategies and a fantastic content for viral marketing. For other things, anything with positive effects always has an adverse impact. When it comes to viral marketing, it's not only rainbows and butterflies. It's hard for certain brands to go viral. It's not always possible for the marketing team to anticipate and avoid a viral trend for all the wrong reasons. This refers in particular to smaller firms that cannot afford to take away harm caused by negative ads.

All of us saw poor reviews that took off the social media and the uproar they could cause. Internet users' rage may seem overwhelming when they come down to a person or business they do not like. The disassociation behind keys means that people are more hostile and are less able, even though they have no direct experience with this issue. They are more likely than others to express their disagreement. More crucial is how negative press is handled. This can add to an already complicated situation if it is done poorly. This aspect is therefore important to be taken into account as early as possible in order for any future harm to be adequately minimised and controlled.

8.6 Planning for Viral Marketing

As the world is filled with more campaigns vying for exposure, advertisers are finding it increasingly difficult to achieve a hit. Careful preparation means that when the campaign begins, you are set. Careful preparation also ensures that you have the highest chance of success with the campaign.

A video created by your department and seen by millions of people across the Internet is undoubtedly a very appealing concept, but it's nice, yet pointless if it doesn't add anything to your business objectives. Will this channel fulfil your target market? Find out if your campaign aims at brand awareness, driving traffic to your website, increasingly forecasting a new product, creating SEO relations or ensuring that customers purchase a product immediately. Whether or not it is one of the main purposes of the campaign, you should always be interested in viral promotions. Much is happening on the Internet. In order to send a campaign forward, it must be separated from the clutter. Make something scarce, educational, very usual, interesting and unusual.

The simpler a message is the more likely it is to be transferred. Viral marketers may use current social media as a hosting and networking environment to convey their message. For instance, using a common video hosting website like YouTube makes it easy for users to put the video on their own pages for a viral video. People want to be viewed as resourceful, compassionate by their friends and colleagues, and the content you produce should be related to them. Your content will have even better results if you tailor for your target market. Track customer experiences with your brand to see development and reaction. Make sure you know the various ways users will communicate about you. If the campaign expands, but the message is not as planned, some changes may be needed. Careful preparation and excellent content can all contribute to a good campaign with an appealing motivation. For a campaign to fly a little luck is typically required.

8.7 Who Is Carrying Out Viral Campaigns for Marketing?

Viral marketing is useful either as an individual marketing tactic or in a broader campaign using different marketing forms. Smaller businesses or companies find this particularly appealing because viral marketing may offer a cheaper alternative to conventional marketing activities. For example, a new energy drinking company can produce an Internet video featuring a person who drinks the energy before attempting an almost impossible bike leap. It might inspire people to share the video with others if the video is made to look genuine. The business will disclose its real intent after it receives enough views, persuading its viewers to look for more drink information without any usual ads.

In tandem with other types of marketing, such as the Blair Witch Project, viral marketing is commonly used. Long before advertisements, trailers, posters and other types of conventional marketing were released, the viral element of the campaign created excitement. This prompted many to speak about the film before it was released officially. To be viral, the contents must not be covert. Political campaigns also make videos that show a candidate's sound clips, claiming that something is offensive to people. Politicians hope the video will be viral and cause others to grow a negative attitude towards the targeted opponents by pointing out an outrageous comment.

8.8 Implementing Viral Marketing

A common misconception is that the credibility of something well known is the foundation of viral marketing. In reality, it depends more on an effective viral marketing campaign to connect with this demographic through useful information. Viral marketing is a personalised strategy that allows marketers to review and evaluate population data to identify and value a product's demographic target. After an organisation understands what its purpose is and how it interacts, it continues to build content that people want to share. Viral marketing is an interactive and ongoing operation. A organisation running a viral campaign needs not simply let its path go unchanged. It has effectively gained interest in the product, like most viral campaigns.

8.9 Ways to Improve Your Chances of Going Viral

1. It is important to keep your target audience in mind before doing something. Get it as generic as you can. It is important to monitor demographics in order to know your audience. You should remember your audience age, sex, location and preferences before creating any content.
2. When marketing in social media, some brands make a mistake by not creating high-quality content and concentrate only on promotional campaigns. While successful brands definitely have great marketing campaigns, but they also provide quality content frequently to ensure consumers are engaged.
3. Some people eat when they're down. Some people watch a sports film and go to the gym. The emotions after seeing, hearing or reading something affecting your actions. Integrating different emotions into your content will make your message more powerful. Scarecrow campaign from Chipotle is an excellent example of fuel emotion.
4. It is a fantastic way to take advantage of a current momentum to submit a tweet or create a blog post at the perfect moment. The date and the time you post your content should be considered. Marketers take advantage of big holidays. In order to make your content viral, it is crucial that you post when more people will likely reach your content.

5. Be assured that you will only be able to make your content with the right type of social media platform. Every platform differs and varies the type of content it contains. Depending on the audience and content, you can choose the platform.
6. If you have clearly identified your target audience desires and social media activity, you can use it to partner with other brands and influences to improve your chances of going viral. Your post is viral if your business has been supported by more influencing friends and close associates. This enhances your credibility. Work with people not only in your work sector but also outside your business.

8.10 Examples of Viral Marketing

Viral content typically has a viral plan well-designed but virility is often due in part to good luck, imagination and planning. Social media really has evolved as a marketing medium into its maturity. The way people connect, share and engage with brands is currently dominating socially. Social is the first thing people do online and social media are the most significant platform for inspiring purchases, according to the PwC survey of 22,000 customers. Viral marketing is like coronavirus, you assume it's just the little flu, but it spreads across the world before long.

The only difference is people would love viral marketing, not like COVID-19. A few carefully selected examples are covered in this section to help explain why and how a viral marketing initiative can be a double-edged and profitable spur.

Max Lanman created an ad in November 2017 to announce the 1996 Honda Accord by his mate. The announcement showed Lanman's girlfriend drinking coffee and keeping her cat in the vehicle while driving included some really nice witty printing like 'Cat and Coffee Pot Not included' and '0% APR for eBay-qualified buyers in good standing'. The sales price of 499dollar was caused by the video, but before the site was pulled, the eBay car auction rapidly increased to up to 150,000 dollars. Later, Car Max created a commercial for the Honda, which the couple embraced, offering the couple US$ 20,000. The video of Car Max itself has over 400,000 views, making it more effective than all of its other posts.

TikTok is the biggest question for many marketers, particularly when they try to reach Gen Z users. In particular, they made a trail for this campaign and developed a hashtag that allows users to upload videos of their own eyes, lips and faces. With over 2.8 million users created videos, the viral campaign became the most popular TikTok campaign ever.

In 2012, Australia's metro trains decided to find a way to promote people to safer around trains. McCann Australia wanted to bring levity to Metro Train advertisements as an alternative to the traditional approach to terrifying and off-talk ads and to provide us with the instant hit Dumb Ways to Die. The video went around the world and was impressed in the media in 2013 with more than 60 million. Most notably, however, the video's message has made people more conscious and safer around trains, which decreases rail injuries by 20%.

Dietz Nuts—the first meat nut ever in the Super Bowl 2019—was introduced by Deli Brand Dietz and Watson. The Office actor, Craig Robinson, shared the Zingers of the latest product in the marketing campaign to support this launch. The video was shared on digital and social media by Dietz and Watson instead of investing in the high price tag of a Super Bowl TV commercial. It resulted in a campaign that played in the Super Bowl hysteria, but which also engaged users in social networking.

However, the movement did not stop socially. Dietz and Watson created pop-up shops, in-store products and goods to help the video virality. For example, a new energy drink company might create an internet video that would show anyone who uses the drink before they jump, which seems to be unavoidable. If you make the video look real, you are inspired to share it. The business would reveal its true intention to convince viewers to pursue more Drink Details without any traditional ads after the video obtained ample views.

What followed will still remain one of the most popular examples of viral marketing. Oreo's fast thinking and wit allowed them to steal the game momentarily and concentrate on the brand. Oreo tweeting 'Dunk in the Dark' in the 2013 Super Bowl. While it is difficult to prepare content such as this, this tweet is so renowned and so often quoted by quick thinking and witnesses from marketers.

Another example Burger King has developed a website for the promotion of its new Tender crisp sandwich that allows users to send commands to the 'subservient chicken'. During the time most are leaving websites within 8 s of their visit, several of the first 15 million visitors to Burger King's subordinate chicken page spent 6 min or more in the area.

In April 2013, Unilever wrote a campaign to inspire women to see what they are like, along with its Ogilvy and Mather Brazil advertising agency. Dove 'Real Beauty Sketches' was a campaign showing a woman's viral video walking through a revolving door and saying 'average' or 'beautiful'. The Dove research has shown that most women do not find themselves beautiful and the campaign is planned to prove that they are beautiful. The advertisement was viral because it could emotionally communicate with the audience.

The Ice Bucket Challenge going viral is as fantastic for non-profit organisations as conventional brands. With his Ice Bucket Challenge video ad campaign, the amyotrophic lateral sclerosis (ALS) association illustrated this. Participants were encouraged to take a video to pour ice water over their head and then challenge three people to do the same thing and donate. This raised 115 million dollars for the non-profit. More than 2,5 million people, including celebrities including Tim Cook, Cristiano Ronaldo, Rafael Nadal Bill Gates and Oprah Winfrey, engaged alone in the United States.

BlendTec's 'Will It Blend' campaign is another example of viral marketing. The mixer enterprise was very aware of the brand in 2006 and has produced a range of videos showing the CEO of the company putting random items in one if its mixers. Its mixer has damaged things including an iPhone, a rake bag, a video game and credit cards. The videos had over 6 million views, and within 5 d of being released on YouTube, BlendTec became world-famous as an obscure brand. Furthermore, after the start of the campaign, their sales rose eight times.

The creativeness and campaigns companies deliver lie at the core of every digital brand today, no matter how large or small, and brands are beginning this year. Some of the most creative and inspiring social media initiatives by 2020 have been seen. Because of the pandemic, Netflix has opted to use social media for more direct contact with its audience. The initiative led to many live sessions at Instagram, where users can directly ask questions and talk to mental health experts. Netflix has seen more interest of more than 100,000 users on their Instagram live.

In March 2020, Zoom launched a monthly virtual background contest, in which remote workers can share their images and videos using the virtual background features of the app. This is an ongoing contest and three entrants will receive exclusive awards per month. It would seem that this competition has accomplished both of the above objectives. More than 50,000 people signed up for the free trial only to participate

8.11 Conclusion

Viral marketing is a powerful tool because, without undue marketing efforts, it is able to increase brand recognition on a large scale in a very short period of time. The trick for a viral message is that the audience needs to be drawn and involved in exceptional and imperative in nature. As the right content is exposed to the right target audience, the customer does the remainder. Through your family, friends and colleagues, you can share and propose helpful ideas in your social networks. In other words, they become marketers and outlets for sale. When done correctly, viral marketing will create a high level of interest in your company and goods among customers. If you make amazing content or at least put a name in the mind of people who have to buy things you sell, you like it to your customers. You will provide people with the tools to create a strong WoM digitally.

In today's world where people are linked to the latest technology more than ever on the Internet, traditional one-way communication technologies seem anachronistic and viral marketing, a two-way mechanism is very fits in very well. The modern customer enjoys talking to his colleagues on brands and goods, writing notes, reviews and engaging in ratings and reviewing platforms. Most significantly, though, he values the words of others rather than corporations, since he sees them as more honest and dissatisfied as they do not consider commercial or publicity motives. Viral marketing can also lead to incredibly positive results that can be seen in the form of an increased exposure to products, recognition and ultimate sales.

Since viral marketing is unpredictable and means its own very existence. Marketers should not be fascinated by this tool's amazing powers and advantages but should bear in mind that this does not carry negligible risks and could hurt the company. Viral marketing can therefore be considered to be a double-edged sword and should be used with great caution.

References

1. Philip Kotler, D.C.: Marketing Moves: A New Approach to Profits, Growth, and Renewal. Harvard Business Press (2002)
2. Phelps, J.E., Lewis, R., Mobilio, L., Perry, D., Raman, N.: Viral marketing or electronic word-of-mouth advertising: examining consumer responses and motivations to pass along email. J. Advert. Res. **44**(4) (2004)
3. Wolfgang Palka, K.P.: Mobile word-of-mouth - a grounded theory of mobile viral marketing. J. Inf. Technol. **24**, 172–185 (2009)
4. Kaplan, A.M., Haenlein, M.: Two hearts in three-quarter time: how to waltz the social media/viral marketing dance. Bus. Horiz. **54**(3), 253–263 (2011)
5. Dobele, A.L.: Why pass on viral messages? Because they connect emotionally. Bus. Horiz. **50**(4) (2007)
6. Grifoni Patrizia, F.F.: An integrated framework for on-line viral marketing campaign planning. Int. J. Bus. Res. (Toronto) (2013)
7. Schroeder, S.: Social networks play a major part in how we get news (2010). Mashable.com
8. Bulkeley, W.: E-commerce: advertisers find a friend in viral marketing [Electronic version]. Wall Street J. Eur. 25 (2002)
9. Youssef, R.O.: Measurement of the impact of viral marketing on the decision to purchase. Zarqa Univ. J. College Econ. Adm. Sci. 1–45 (2009)
10. Wangenheim, F.: Postswitching negative word of mouth. J. Serv. Res. **8**(1), 67–78 (2005)
11. Williams, T., Slama, M.: Market mavens purchase decision evaluation criteria: implications for brand and store promotion efforts. J. Consum. Mark. **12**(3), 4–21 (1995)
12. Marketing, D.: Viral Marketing Vs Conventional Marketing. Onlinstartupbo.: http://www.onlinestartupbox.com/news/viral-marketing-vs-conventional-marketing/. Accessed 12 Nov 2016
13. Fraile, A.: What is viral marketing? Examples and advantages (2020). https://www.cyberclick.net/numericalblogen/what-is-viral-marketing-advantages-and-examples. Accessed 13 July 2020
14. Katz, E., Lazarsfeld, P.F.: Personal Influence, Glencoe, Ill. The FreePress (1955)
15. Botha, E., Reyneke, M.: To share or not to share: the role of content and emotion in viral marketing. J. Public Affairs **13**(2), 160–171 (2013)
16. Carl, W.: The role of disclosure in organized word-of-mouth marketing programs. J. Mark. Commun. **14**(3), 225–241 (2008)
17. Ferguson, R.: Word of mouth and viral marketing: taking the temperature of the hottest trends in marketing. J. Consum. Mark. **25**(3), 179–182 (2008)
18. Pretorius, B.: The future according to .... Brand Pretorius. Future **1**(5), 73–75 (2000)

Chapter 9
Evolving Models for Dynamic Weighted Complex Networks

Akrati Saxena

Abstract For decades, complex networks, such as biological networks, social networks, chemical networks, and technological networks, have been used to study the evolution and dynamics of different kinds of complex systems. These complex systems can be better described using weighted links as binary connections do not portray the complete information of a system. All these weighted networks evolve in a different environment by following different underlying mechanics. Researchers have worked on unraveling the evolving phenomenon of weighted networks to understand their structure and dynamics. In this chapter, we will cover the evolution of weighted complex networks and evolving models to generate different types of synthetic weighted networks, including undirected, directed, signed, multilayered, community, and core–periphery structured weighted networks. We will further discuss various properties held by generated synthetic networks and their similarity with real-world weighted networks.

9.1 Introduction

In real-world complex networks, each link carries a unique strength or weight [1, 2]. For example, in a friendship network, the edge weight denotes the intimacy of the relationship or frequency of the communication [2]. In a co-authorship network, edge weight denotes the number of publications co-authored by two researchers. In an airport network, the edge weight can represent the number of available seats, traffic flow [3], or the frequency of transport availability between two airports [4].

The origin of the concept of strength of ties dates back to 1973 when Granovetter [5] introduced the idea of weak ties. He emphasized the inequality of connections in a social network and broadly categorized them as strong or weak ties. A connection is referred to as a strong tie if these people frequently talk to each other, otherwise it

A. Saxena (✉)
Department of Mathematics and Computer Science, Eindhoven University of Technology, Eindhoven, The Netherlands
e-mail: a.saxena@tue.nl

A. Biswas et al. (eds.), *Principles of Social Networking*,
Smart Innovation, Systems and Technologies 246,
https://doi.org/10.1007/978-981-16-3398-0_9

will be referred to as a weak tie if the communication is not frequent. After this work, Lin et al. [6] emphasized the importance of strong ties in self-growth as well as of weak ties in job achievement. In a friendship network, links can also be categorized as close friendship, acquaintance, obligation, or alliances. Researchers have verified the importance of social tie strength in real-world social networks [7–9].

Weighted networks are characterized by a function $f(E) \rightarrow R$, where the function f maps each edge to a real number, which represents the weight of that edge. Depending on the applications, weighted networks can be categorized as directed or undirected, and an edge will be denoted by an ordered or unordered pair of nodes, respectively. For example, co-authorship networks are undirected weighted networks, whereas transportation networks are directed weighted networks. In real-world networks, edges can also carry negative weights. The examples of such networks are frenemy (friend–foes) networks [10, 11] or trust–distrust networks [12, 13]. The edges between two people in these networks can represent varying degrees of friendship/trust or enmity/distrust. The former kind of edges is represented by positive weights and the latter by negative weights. Akin to unweighted networks, weighted network also carry several properties similar to unweighted complex networks, such as scale-free degree distribution [14, 15], small world phenomenon [16], high local clustering coefficient [17], community structure [18], assortativity [19], and so on. Weighted networks have been used to study their evolution and analyze the dynamic phenomenon occurring on these networks, such as information diffusion, opinion formation, influence propagation, and so on. However, the data collection to create weighted social networks is a computationally time taking and costly procedure, as it requires the knowledge of the strength of all relationships in a network. Therefore, researchers have studied the evolution of the topological structure of dynamically growing complex networks and proposed several modeling frameworks to generate synthetic weighted networks having properties as observed in real-world weighted networks. The first evolving model to generate weighted networks was proposed by Yook et al. [20] in collaboration with Albert-László Barabasi. After this pioneering work, researchers have proposed several other models to explain various aspects of network structure based on their evolving phenomenon and corresponding environment.

In this chapter, we will cover evolving models to generate different types of weighted networks, including undirected and directed networks, signed weighted networks, multilayered networks, community structured, hierarchical structured, and specialized evolving models. In literature, most of the proposed evolving models are dynamic, i.e., the network starts with some seed nodes, and then the nodes and edges are added iteratively based on the proposed growing method. We will discuss evolving models for different kinds of networks in the coming sections.

9.2 Preliminaries

In this section, we discuss some required definitions in the context of weighted networks. The following notations will be useful to understand given definitions. A binary network is represented by a matrix A, where $A_{xy} = 1$, if x and y are connected else $A_{xy} = 0$. Similarly, a weighted complex network can be represented by a matrix, where w_{xy} is the weight of the edge between vertices x and y. Let $\Gamma(x)$ denote the set of neighbors of node x in the given network, and k_x denotes the degree of node x. In this chapter, the terms nodes, vertices, and users will be used interchangeably, and the terms links, edges, and connections will be used interchangeably.

9.2.1 Strength of a Node

In weighted networks, the strength of a node is computed by adding the weight of all links connected to that node. It is denote by s_x and

$$s_x = \sum_y w_{xy}.$$

In a directed weighted network, the in-strength of a node can be computed as the sum of the weight of incoming edges and out-strength as the sum of the weight of outgoing edges.

9.2.2 Power-Law Distribution

Barabasi and Albert observed that in real-world unweighted networks, the degree distribution follows a power law, and the probability of a node having degree k is observed as $P(k) \propto k^{-\gamma}$ [21].

Weighted networks also follow three power-law distribution: (i) power-law degree distribution, (ii) power-law strength distribution, and (iii) power-law edge-weight distribution.

9.2.3 Preferential Attachment Model

The preferential attachment model, also referred as BA model, is an evolutionary model proposed by Barabasi and Albert for the formation of unweighted scale-free complex networks [21]. In BA model, the probability $\prod(k_x)$ that a new coming node will connect with an existing node x is proportional to its degree. So,

$$\prod(k_x) = \frac{k_x}{\sum_y k_y}.$$

Analogously, in a weighted network, the nodes follow preferential attachment in accordance with the strength s_x of the nodes. So,

$$\prod(s_x) = \frac{s_x}{\sum_y s_y}.$$

The generated networks follow a "rich get richer" phenomenon and therefore have power-law degree and strength distributions.

9.3 Undirected and Directed Weighted Networks

The first work to propose evolving models for generating synthetic weighted networks dates back to 2001, when Yook et al. [20] proposed two evolving models, (i) Weighted Scale Free (WSF) model and (ii) Weighted Exponential (WE) model. In both models, the network starts with a seed network having n_0 vertices. In the WSF model, each new incoming vertex is connected with m vertices using the degree preferential attachment rule. Each new vertex x has fixed weight $w_x = 1$, and this weight is distributed among its all m edges. The weight distribution across each edge (x, y) is proportional to the degree of other endpoint as $w_{xy} \propto k_y$

$$w_{xy} = \frac{k_y}{\sum_{y'} k'_y},$$

where y' belongs to m chosen neighbors of the new node x.

In the WE model, a new node makes m connections with the existing nodes with equal probability, and the weight assignment for the edges is the same as in the WSF model. The authors show that the generated networks using both the models follow power-law degree, strength, and edge-weight distribution. After this work, several evolving models have been proposed, and we have categorized them as discussed in the following subsections.

9.3.1 Power-Law Minimal Model

In real-world weighted networks, the strength, degree, and edge weight follow a power-law distribution. In this section, we will discuss evolving models based on power-law properties that can be designed with minimal effort, therefore referred to as *minimal models*.

In 2005, Antal and Krapivsky [22] proposed an evolving model where the network starts with some seed nodes, and each new incoming node x makes an edge with an existing node y. The edge weight w_{xy} is chosen uniformly from the given edge-weight distribution $\rho(w)$ and remains fixed once assigned. They further introduced two generalizations of this model, (i) each new node makes m connections, and (ii) after adding a new node, some links are also added among already existing nodes. In this model, the probability of adding a link (x, y) is directly proportional to the strength of its endpoints $P(x, y) \propto s_x \cdot s_y$. The authors showed that in the generated networks, the strength distribution has a tail, i.e., independent of the edge-weight distribution.

Bianconi [23] studied the evolution of weighted networks and divided these networks into two main classes based on the ratio of the rate of strengthening of edge weights and the rate of making new connections. In traffic networks, new links are formed more frequently with new nodes than increasing the strength of the already existing links. This type of networks is called class 1 networks, and the strength and degree of a node follow a linear relationship, $s_x \propto k_x$. While in co-authorship networks, two existing collaborators publish more papers together than with a new collaborator. This type of networks is called class 2 networks and follow non-linear relation as $s_x \propto k_x^\theta$ where $\theta > 1$. This model uses two steps, at every timestamp, (i) a new node is added and makes m connections using degree preferential attachment, and (ii) choose m' already existing edges preferentially and increase their weights by w_0. To choose m' edges, the model first chooses a node using degree preferential and then selects one of its edges with the probability directly proportional to the edge weight. We can set different values of m and m' to generate class 1 and class 2 networks.

Mukherjee and Manna [24] studied the weighted model based on the self-organizing link weights dynamics, and the following two steps govern the growth of the network.

1. The network is started with a seed graph. At every timestamp, one edge is selected randomly, and a node is connected to one of its endpoints with equal probability.
2. After adding a new node, edge weights are updated using the following two steps:
   (a) $s_x = \sum_y w_{xy}$, and
   (b) $w_{xy} = (s_x s_y)^\alpha$, where α is a parameter and its value can vary for different applications.

These networks follow all three power-law distributions observed in weighted networks.

9.3.2 Fitness Model

In the above-discussed models, the number of connections that a node attracts is directly proportional to the strength or degree of that node. However, this phenomenon is not enough to explain the network evolution and its dynamics for many real-world networks. In 2001, Bianconi and Barabási [25] introduced the concept of fitness for unweighted networks where each node has a fitness value η and more fit nodes attract more links; this phenomenon is also called "fitter gets richer".

Wang and Zhang [26] presented the Weighted Competition Scale Free (WCSF) model based on fitness that is quite similar to the BA model. In this model, the network starts with a seed graph, and at every timestamp,

1. With the probability p, a new vertex x is added to the network with m edges connected preferentially. This new vertex has fitness value η_x that is chosen uniformly from a fixed power-law fitness distribution $\rho(\eta)$ in the range $[\eta_{min}, \eta_{max}]$ and $\eta_{min} > 0$. The probability of selecting an existing node y to make a new connection considers the fitness of the node. It is defined as

$$\prod_y = \frac{\eta_y k_y}{\sum_z \eta_z k_z}.$$

2. With probability $(1 - p)$, the network will be self-grown, as any new node will not join, and m edges will be added among the existing users. To add a new edge, two end nodes are selected using the fitness preferential attachment rule. If they are not connected, connect them with edge weight $w_0 = 1$, otherwise increase the edge weight by $w_0 = 1$.

Thus, in the fitness model, the nodes having high fitness attract more new and self-growing links.

Zheng et al. [27] presented a stochastic weighted model where each node has a fitness value η uniformly distributed over range [0, 1]. In this model, with probability p, the edge weights are assigned using the WSF model, and with probability $(1 - p)$, edge weights are assigned using the fitness parameter.

9.3.3 Stochastic Model

In [28], the authors proposed a simple stochastic model by incorporating weight dynamics in the BA model. The authors assume that the link weights are evolved as the geometric Brownian motion, also called Gilbrat's law of proportionate effects. To understand weight dynamics, they used the theoretical framework of the scaling distribution of fluctuations proposed by Yamasaki et al. [29]. The model will start with a seed network having n_0 vertices, and each vertex has a self-loop. At every timestamp t, a vertex x is chosen with probability $k_x(t-1)/2t$ to make a new connection. With probability α, a new vertex will join the network, and vertex x is connected to this. With the probability $(1 - \alpha)$, vertex x is connected to already existing vertex y, where vertex y is chosen with probability $k_y(t-1)/(2t - k_x(t-1))$ if $x \neq y$, otherwise probability is 0. The edge weight is assigned randomly from a fixed power-law distribution. After adding a new edge, the weight of each edge is reduced or increased using a random factor $\chi_{xy}(t)$, so edge weight at time $(t+1)$ is $w_{xy}(t+1) = w_{xy}(t)\chi_{xy}(t)$. Thus, the weight of edges grows randomly with time, and it is similar to geometric Brownian motion. The network evolves from random to exponential to scale-free graphs as the value of α increases.

9.3.4 Incremental Weight-Distribution Model

In traffic networks, the addition of a new edge to an existing vertex increases the traffic on that node. This increased traffic is distributed among the neighbors of the node; such a distribution is called the local rearrangement of the weights. In 2004, Barrat et al. [30] presented a weighted traffic evolving model that is based on this phenomenon, also called the BBV (Barrat, Barthelemy, and Vespignani) model. This model has the following two steps.

1. The model starts with completely connected n_0 nodes, and each edge has weight w_0.
2. At every timestamp, a new node x is added and connects with m nodes having weight w_0 using strength preferential attachment rule. Each new connection adds some extra traffic δ to its endpoint node, so

$$s_y = s_y + w_{xy} + \delta.$$

   This extra created traffic is distributed to all the neighbors of the node proportional to their edge weights. Now, the new weight of an edge (y, z) is computed as

$$w_{yz} = w_{yz} + \delta \tfrac{w_{yz}}{s_y}.$$

If $\delta = w_0 = 1$, it perfectly depicts the traffic model, where, all incoming traffic is mostly distributed on outgoing links. If $\delta < 1$, it is similar to collaboration network, where a new collaboration does not affect old collaborations of the node. $\delta > 1$ signifies the situation where incoming edge bursts more traffic on other neighbors. The authors further presented the evolution of the strength and degree of the nodes as well as of the edge weight. The strength of a node will be changed if the link is directly connected to this node or one of its neighbors. So, at time t, the average strength of a node x $(s_x(t))$ can be calculated using evolution equation for s_x,

$$\frac{\mathrm{d}s_x}{\mathrm{d}t} = m\frac{s_x}{\sum_y s_y}(1+\delta) + \sum_{y\epsilon\Gamma(x)} m\frac{s_y}{\sum_z s_z}\delta\frac{w_{xy}}{s_y}.$$

Similarly, the degree of a node x is evolved as

$$\frac{\mathrm{d}k_x}{\mathrm{d}t} = m\frac{s_x(t)}{\sum_y s_y(t)}.$$

Similar to the strength evolution, the weight of an edge changes when a node is directly connected to it or one of its neighbors. The rate of change of edge weight w_{xy} can be computed as

$$\frac{\mathrm{d}w_{xy}}{\mathrm{d}t} = m\frac{s_x}{\sum_y s_y}\delta\frac{w_{xy}}{s_x} + m\frac{s_y}{\sum_y s_y}\delta\frac{w_{xy}}{s_y}.$$

The networks generated using the BBV Model follow a power-law distribution for the degrees, strengths, and edge weights.

Hu et al. [31] extended the BBV model [30] and proposed two separate models for traffic and friendship weighted networks. In the traffic-driven model, they take a constant traffic increasing rate W for a network. This new traffic is distributed among all the edges proportional to their strength. The rest of the mechanism of this model is similar to that of the BBV model. At every timestamp, the extra added traffic on each node is computed as

$$\Delta W_x = W\frac{s_x}{\sum_y s_y}.$$

So, the time evolution equation of edge weight is given as

$$\frac{\mathrm{d}w_{xy}}{\mathrm{d}t} = \Delta W_x\frac{w_{xy}}{s_x} + \Delta W_y\frac{w_{xy}}{s_y}.$$

Time evolution equation of strength is given as

$$\frac{\mathrm{d}s_x}{\mathrm{d}t} = \sum_{y \epsilon \Gamma(x)}\frac{\mathrm{d}w_{xy}}{\mathrm{d}t} + m\frac{s_x}{\sum_z s_z}.$$

Time evolution equation of degree is given as

$$\frac{\mathrm{d}k_x}{\mathrm{d}t} = m\frac{s_x}{\sum_y s_y}.$$

In the second model, they explained the dynamics of friendship formation using "friends of friends" phenomenon. Network formation starts with a seed graph, the same as in BBV model. At every timestamp,

1. A new vertex x will be connected with a vertex y using strength preferential attachment.
2. the vertex x is connected with m neighbors of vertex y with the weight preferential probability, given as

$$P(x, z) = \frac{w_{yz}}{s_y},$$

where $z \in \Gamma(y)$.

The first connection is called the primary edge, and the remaining m connections are called secondary edges. To control the effect of newly added links, they do the local rearrangements of weights the same as the BBV model. This model perfectly illustrates the real-world social networks or collaboration networks. If a new person A forms a link with a person B, then there is a very high probability that A will make links with $B's$ friends.

Goh et al. [32] proposed a very specific model for traffic networks. In traffic networks, the weight of a link can be calculated by measuring the load of the traffic on that edge [33] or betweenness centrality of that edge [34]. The edge weight also depends on the degree of its endpoints as $w_{xy} \sim (k_x k_y)^{\theta}$. The load of a vertex is calculated in a similar manner as the strength of a vertex. Therefore, the load of the vertex is the sum of the load of all edges connecting to that vertex. The load of a vertex is directly proportional to the strength of that vertex $l_x \propto s_x$. The model starts with the seed graph the same as mentioned in the BBV model. At every timestamp, a new vertex is added with m edges using sub-preferential attachment probability as

$$P(x) \sim s_x^{\alpha} \sim l_x^{\alpha},$$

where $\alpha = 1/\eta$ and $l_x \sim k_x^{\eta}$. The load of each vertex is recalculated after every step. The networks generated using this model follow strength and degree power laws. This model gives a new view to understand the evolution of weighted networks with the increasing load in the system. Each new vertex and link can affect the shortest pathway in the networks. So, link weights are updated regularly. Some other networks, such as a friendship network or co-authorship network, where the edge weight depends on the harmony of their relationship, can be best explained using the BBV model.

9.3.5 Node-Deactivation Model

Wu et al. [35] presented a model inspired by the concept of degree-dependent node-deactivation model [36] and the strength-based traffic distribution [37]. This model considers that each node can have either of the two states: active or inactive. Pandya proposed the strength-based traffic distribution idea where increased traffic moves toward the bigger airports that can handle it [37], and thus the traffic on high strength node gets increased. In [35] model, at every time step, m nodes are in active state and a new node will join and will be connected with these m active vertices. The increased traffic on connected nodes is distributed as

$$w_{xy} = w_{xy} + \delta \frac{s_y^{in}}{\sum_{z\epsilon\Gamma(x)} s_z^{in}},$$

where s_x^{in} is the in-strength of node x. The newly added node will always be in the active state in the next time step. So, there are total $(m+1)$ active nodes, and one active node will be converted to inactive node with the probability given as

$$\prod(s_x^{in}) \propto \frac{\gamma - 1}{a + s_x^{in}},$$

where $a > 0$ and $\gamma - 1 = \frac{1}{\sum_{y\epsilon M} 1/(a+s_y^{in})}$, M contains all active nodes in the last time step. It shows that a high strength node has low probability to be deactivated in the next time step. The constant parameter a is used as a bias factor and affects the power-law exponent by varying the probabilities of node deactivation.

Tian et al. [38] studied the deactivation mechanism using the total strength of the nodes. They proposed a different probability function to deactivate a node given as

$$\prod(s_x) = \frac{\alpha}{s_x},$$

where $\alpha = \frac{1}{\sum_{y\epsilon M} 1/s_y}$. They also proved that the average clustering coefficient reduces as the number of active nodes increases, so $C \propto 1/M$. Due to the deactivation phenomenon, these networks possess disassortative behavior and hierarchical organization. These models can help us in a better understanding of disassortative networks. These networks also follow all weighted network power-law distributions.

9.3.6 Non-linear Growing Model

In some real-world networks, as new nodes keep coming, the rate of adding new connections by the newly added node increases exponentially. The reason is that the network size increases with time, and a new coming node has more options to make connections. This type of networks is called *exponentially growing networks* as the total number of edges grows exponentially with time. In these networks, when a new node joins at timestamp t, the number of connections made by this new node follows the power-law function. So, the number of edges added by this new node is t^θ, where θ is called acceleration parameter and $(0 \le \theta \le 1)$. Zhang et al. [39] presented a model to capture the evolution of such growing networks. In the given model, at time t, the graph has $(t + n_0)$ vertices (n_0 is the number of vertices in the seed graph) and $\int t^\theta \mathrm{d}t = t^{1+\theta}/(1+\theta)$ edges. The weight rearrangement dynamics is the same as in the BBV model. The authors showed that the value of the accelerating exponent is 0.56, 0.18, and 0.12 for the arXiv citation graph, autonomous Internet graph, and the email network, respectively. By setting different values of θ, one can get scale-free to exponential graphs, and the degree, strength, and edge-weight distribution changes from a small exponent to a large exponent. The clustering coefficient also depends on the value of θ, and as θ increases, the clustering coefficient also increases. If $\theta = 1$, the model converts to the basic weighted scale-free model.

Wang et al. [40] studied the non-linear growing network and propose two models to reproduce these networks. [41] proposed a model where each new node makes edges proportional to the size of the network $m \propto N$.

9.3.7 *Incremental Self-Growing Model*

Real-world networks self-grow, where the already existing nodes keep making new connections. For example, in a co-authorship network, new researchers will join and make connections with the existing nodes, and in the meantime, already existing researchers will also build new collaborations. In this section, we discuss models that capture both topological growth as well as self-growth.

In 2005, Wang et al. [42] proposed the first self-growing model for weighted networks. The network starts with n_0 fully connected nodes, and each edge is having weight w_0. The network grows by following two given mechanisms, at every timestamp,

1. **Topological Growth**: A new node will join and it will be connected with m existing nodes using strength preferential attachment.
2. **Mutual Selection Growth**: Each existing node x selects m' other nodes with the given probability function as

$$P(x, y) = \frac{s_y}{\sum_z s_z - s_x}.$$

If the chosen node y is already connected to x, their edge weight is increased by w_0. If they are not connected, an internal link (x, y) is formed with edge weight w_0.

The evolution equation of edge weight can be written as

$$\frac{\mathrm{d}w_{xy}}{\mathrm{d}t} = m\frac{s_x}{\sum_z s_z - s_x} \times m\frac{s_y}{\sum_z s_z - s_y}.$$

The strength of a node x can be changed if the new node is connected to it or this node is chosen for mutual selection growth. Th strength evolution equation is defined as

$$\frac{\mathrm{d}s_x}{\mathrm{d}t} = \sum_y \frac{\mathrm{d}w_{xy}}{\mathrm{d}t} + n \times \frac{s_x}{\sum_z s_z}.$$

This equation's solution shows that the total strength of nodes is uniformly increased with the network size. The authors showed that the model depicts the disassortative nature of real-world networks.

Xie et al. [43] extended the work in [42] and proposed a strength dynamic model where all edges are strengthen up continuously and new connections are added among existing nodes. In this model, the topological growth works same as in the [42]. However, they proposed a different approach for strength dynamics. At each time step, all edges update their weight using the strength coupling mechanism given as

$$w_{xy} = \begin{cases} w_{xy} + 1 \; with \; probability \quad Wp_{xy} \\ \quad w_{xy} \quad with \; probability \; 1 - Wp_{xy}, \end{cases}$$

where

$$p_{xy} = \frac{s_x s_y}{\sum_{m<n} s_m s_n}.$$

W shows the increasing rate of edge weight. If the value of Wp_{xy} is greater than 1, then it is assumed to be 1. A high value of W results in a higher clustering coefficient. Mu et al. [44] proposed a different probability function for the same model as

$$p_{xy} = \frac{2w_{xy}}{\sum_z s_z}.$$

After updating the weight of all edges, newly added edges create an extra flow to the connected nodes.

Tanaka and Aoyagi [45] presented a different self-growing strategy to make or strengthen the links among already existing nodes. In this model, at every timestamp t, ct (c is a constant) pairs of nodes are selected for the self-growth using the probability proportional to their strength as

$$P(x, y) = \frac{s_x s_y}{(\sum_z s_z)^2}.$$

In this model, the total number of vertices is $n \approx t$, so the strength of a vertex will be increased by $\Delta s_x \approx 2c$ when a new vertex is added. They also showed that for a co-authorship network $c = 1.5 \times 10^{-4}$. The lower value of c shows that the rate of joining of a new author is frequent as compared to the already existing scientists collaborating. The communication networks, such as email or friendship networks, where new nodes join very rarely, have a high value of c.

They further analyzed the evolution phenomenon of weighted complex networks and presented an approach to make scale-free networks with variable power-law exponent [46], where different values of the given parameters result in different power-law exponents. They defined the strength-driven preferential attachment probability as

$$P(x) = \frac{(s_x + \sigma)}{\sum_z (s_z + \sigma)},$$

where σ is a constant. In this model, the network starts with one vertex. At each time step, a new vertex will be added with m links having link weight 1 using the strength-driven preferential attachment probability as given above. At each time step, ct^{η} pairs of vertices are chosen for self-growth with the probability defined as

$$P(x, y) = \frac{(s_x + \sigma)(s_y + \sigma)}{\left(\sum_z s_z + \sigma\right)^2}.$$

After choosing a pair of vertices, links are added similar to other models. As a new vertex has maximum m edges, the strength of the new vertex is m. The value of σ is defined as $\sigma > -m$ so that a new vertex can also attract more edges in the next time step. The value of parameter c decides the density of the network, and this model shows exponential growth with time.

Real-world weighted networks can be assortative or disassortative. Leung and Chau [47] proposed a model to create both types of networks by setting value of given parameters. In this model, at timestamp t, a new node will be added to the graph and will make pt^{θ} links preferentially, where $p > 0$ and $0 < \theta < 1$. They modified the self-growth procedure to get assortativity in the network. At every timestamp, a link (x, y) is chosen preferentially and its weight is updated as

$$w_{xy} = w_{xy} + sgn(q),$$

where

$$sgn(q) = \begin{cases} 1 & if\ \ q > 0 \\ -1 & if\ \ q < 0 \\ 0 & if\ \ q = 0. \end{cases}$$

When a link has weight 0, it is removed from the network. For each timestamp t, self-growth process is repeated $|q|t^{\theta}$ times. As the value of q increases, network topological behavior changes from assortative to disassortative.

9.3.8 Triad-Formation Model

All real-world networks have a self-growing phenomenon, and Wang et al. [42] captured this growing phenomenon very well. Hao et al. [48] studied the self-growing phenomenon of real-world complex networks in detail and proposed a model based on their findings. When a new node is added to the network, it is connected to some existing nodes. With time, the already existing nodes also make some connections, but in the real world, these new connections are mostly based on *friends of friends* phenomenon. In real-world networks, people prefer to make connections with the neighbors of itheir neighbors, also called *triad formation* (TF). The proposed model has the following steps.

1. The network starts with a seed graph as in the other models.
2. With the probability p, a new node will be added and will make m connections using the preferential attachment. After adding m edges, the local rearrangement of weights is done as per the BBV model.
3. With probability $(1 - p)$, network is self-grown using m edges. Self-growing edges can be of two types: (i) Triad-Formation (TF) edge or (ii) Random edge. With probability φ, a TF edge is added using TF Rule, and with probability $(1 - \varphi)$, a random edge is added. To make a triad edge, first select an edge (x, y) randomly. Then select one other neighbor z of y using the following preferential attachment rule to make the triad. The probability to choose another node is defined as

$$\prod(z) = \frac{w_{yz}}{(s_y - w_{xy})}.$$

If there is no link between x and z, make a link having weight w_0; otherwise, increase its weight with some constant σ.

This modeling approach can be used to design networks with varying clustering coefficients by tuning probabilities of making PA, TF, and random links. The higher the probability of making triad links, the higher is the clustering coefficient. Zhang et al. [49] studied this process with PA and TF links. These networks also possess a high clustering coefficient. The average clustering coefficient gives power-law distribution with the degree of nodes, $C(k) \propto k^{-\gamma}$. Some more studies have been performed on the triad-formation-based model, including [50–53].

9.3.9 Traffic Flow-Driven Model

In traffic networks, edge weights can be calculated using traffic flow or betweenness centrality measure. In this section, we discuss a few models related to these approaches.

Hu et al. [54, 55] studied the pattern of passenger behavior on the transportation network. In an airline network, they found that passengers either choose a direct route to the destination or by one hop. This middle hop mostly belongs to the hubs of the network. They propose a model where a new node is directly connected to the destination with probability p or by passing through a third node with probability $(1 - p)$. This middle transfer node y on the path $x \to z$ is chosen using edge weight preferential attachment, defined as $\prod(x, y) = \frac{w_{zy}}{s_z}$. This human behavior gives birth to disassortativeness in transportation networks. These networks have high clustering coefficients when p is small; this shows that transform behavior is responsible for cluster formation and hierarchical organization.

Dai et al. [56] presented a model for the traffic flow of the transportation networks. This model is based on deterministic and random node attachment and shows scale-free properties. Zheng et al. [57] studied the flow of traffic with network topology. In this model, nodes represent the traffic flow states, and edge weight represents the transported traffic flow between two different traffic flow states. Each node contains a set of cells as each road can be divided into some cells. The link weight depends on both the state and velocity of the cells. For more detail, [57] can be referred.

9.3.10 Local World Model

In the above-discussed models, a newly added node makes connections globally or using the triad-formation rule; however, in real life, each node belongs to a group or community referred to as its local world. Li et al. [58] introduced the concept of the local world in unweighted networks. These local world-based evolving networks show a transition from exponential to scale-free degree distribution. They further presented an evolving model based on the concept of local world [59]. In this model, a network starts with the seed graph. For every new coming vertex, we follow the next given steps:

1. To determine the local world of a new coming vertex, we choose M vertices randomly from the network, and this group of vertices is called the local world of the new vertex.
2. The new vertex will be connected to m vertices in its local world using the preferential attachment. The probability of connecting new vertex x to vertex y is given as

$$P(x, y) = P'(y \epsilon M) \frac{s_y}{\sum_{z \epsilon M} s_z},$$

   where P' denotes the probability of vertex y belonging to the randomly chosen local world and $P'(y \epsilon M) = M/n(t)$, $n(t)$ shows the total vertices at time step t.
3. In the last step, local rearrangement of the edge weights is done to handle the extra created traffic by new edges. Weights can be rearranged in two ways: (i) using the BBV model rearrangement, (ii) using Hu et al. model [31] except here the constant increased weight is distributed only among the chosen local world vertices proportional to their strength. So, the increased weight of a vertex y $(y \epsilon M)$ is

$$\Delta W_y = W \frac{s_y}{\sum_{z \epsilon M} s_z}.$$

This extra introduced traffic on each vertex ΔW_y is preferentially distributed to all the connecting edges.

In this model, if the new vertex is linked with all vertices of the local world ($m = M$), degrees, strengths, and edge weights show exponentially decaying distribution. If the local world is chosen as the whole network, this model gives the same result as the BBV model. When we increase the size of M, the model shows a transition from assortative networks to disassortative networks.

Sun et al. [60] studied this phenomenon in more detail and presented a weighted model using local information that exhibits the transition from unweighted to weighted networks. In their proposal, at every timestamp, when we determine the local world, one of the following two procedures can happen. With probability p, the network has increment growth by adding a new node with m links in this local world. With probability $(1 - p)$, self-growth happens by adding m edges in the determined local world. So, in t timestamps, pt nodes and mt edges will be added to the network. This model depicts the real-world phenomenon where high-strength nodes keep strengthening their relation, and at the same time, they also make connections with new nodes. They also studied synchronization dynamics on the generated networks [61]. They found that the network synchronization can be increased by reducing the heterogeneity of the edge weights.

To improve the clustering coefficient of the above-suggested model, Zhang et al. [62] included the concept of triad formation. They suggested that when a new node will join its local world preferentially, it will make some triad links (be friend with friends of friends) [63]. This phenomenon happens in real life and helps in improving the clustering coefficient and the average path length. This model starts with a seed graph having n_0 nodes and m_0 edges. At every time step, a new node will be added, and it will make m preferential edges in its local world. After adding each PA link, with probability p, the new node also makes a TF link with any random neighbor of the last chosen node using PA law. By changing the value of p, we can get networks of varying clustering coefficients. After t time step, the network will have $(n_0 + t)$ nodes and $(1 + p)mt + m_0$ edges. This model gives logarithmic average path length with respect to the total nodes.

In 2007, Li et al. [64] presented a model based on the similar concept that nearest neighbors or friends of friends have a high probability of being your friend. The model is inspired by scientific collaboration networks. To use this information, they use a path-based preferential attachment. The probability of connection increases as the shortest path length decreases. Edge weight is defined as a function of the connecting time of that edge. If node x and y are connected at time T_{xy}, then edge weight w_{xy} is defined as

$$w_{xy} = f(T_{xy}).$$

This function can be of different types but they used linear function $w_{xy} = \alpha T_{xy}$ for the simulation. The model starts with the fully connected seed nodes and all edges having weight $w_{xy} = f(1)$. The initial time is set to 1, and at every timestamp, steps from 1 to 4 are followed as explained next:

1. A new vertex will be added to the network, and l existing nodes are chosen uniformly at random.
2. Every selected vertex (x) is assumed to be ready to make an edge. The probability of making an edge from vertex x to vertex y is defined as

$$P(x, y) = (1 - p)\frac{k_y}{\sum_z k_z} + (p - \delta)\frac{s_y}{\sum_z s_z} + \delta\frac{l_{xy}}{\sum_{z \epsilon \partial_x^{1,2}} l_{xz}},$$

   where l_{xy} is the similarity distance between vertex x and y, and ∂_x^d is the set of d distance neighbors of vertex x. In the given probability function, they only used 1 and 2 distance neighbors because they have high probability to encounter and be friends.
3. After selecting a vertex y' using above probability function, x and y' are connected and their connecting time is increased as

$$T_{xy'}(t + 1) = T_{xy'}(t) + 1.$$

4. The edge weight is updated as

$$w_{xy'}(t + 1) = f(T_{xy'}(t + 1)).$$

The networks generated using this model follow strength and degree power-law distribution. These networks also possess a very high clustering coefficient because of using similarity distance. This model can be generalized to get directed weighted networks.

9.3.11 Mutual Attraction Model

In real-world social networks, the friendship between two people depends on their mutual affinity, intimacy, attachment, and understanding. None of these parameters is considered in any above-discussed model. In friendship networks, each person has some attractiveness to grab more friends. A new person with a highly attractive nature can have more friends than an old less attractive person. Attractiveness can be defined as the nature of the person, talkativeness, understanding, confidence, loyalty, etc. For example, in the co-authorship network, the attractiveness of an author cannot be described by only the number of publications. It will depend on other parameters, such as enthusiasm, openness to work with others, research environment, and discipline.

Wang et al. [65] presented a model based on the attractiveness and mutual attraction of two people. The model starts with n_0 isolated nodes; each having initial attractiveness A, $(A > 0)$. At each time step, a new isolated node x joins the network. Then every node in the network chooses m other nodes as the perspective available options to make connections. Node x chooses node y with the probability given as

$$\prod(x, y) = \frac{s_y + A}{\sum_{z \neq x} s_z + A}.$$

However, the selection does not guarantee a connection. Now, if two nodes have chosen each other mutually, their edge weight is increased by 1 if they are not connected; otherwise, an edge having weight 1 is formed. The higher the value of A, the higher the chances for new nodes to get more connections. Parameter m will control the network density. As the value of m increases, each node selects more options, and therefore, hubs have a higher probability of getting more connections. A higher value of m leads to disassortative networks. This model works on common interests and mutual acknowledgments. The edge weight is updated if both nodes select each other, so the time evolution equation of edge weight can be written as

$$\frac{\mathrm{d}w_{xy}}{\mathrm{d}t} = m\frac{s_y + A}{\sum_{z \neq x} s_z + A} \times m\frac{s_x + A}{\sum_{z \neq y} s_z + A}.$$

Similar to attractiveness, the competitiveness of a node is also a very important feature. References [66–68] studied the role of this feature while designing competitive networks.

9.3.12 Geographical Constraint Model

In real-world networks, such as Internet network, road network, airport network, and power grid network, the spatial distance is a major constraint. In these networks, the nodes are placed or organized in a well-defined position, and most of the edges are connected among the geographically nearby nodes. The main parameters considered while making new connections in technological or spatial networks are the strength of a node, the cost of making a new connection, and the benefits of this new connection (cost–benefit analysis). Barrat et al. [69] proposed the first model to generate weighted spatial networks. The authors proposed a specific probability function to make the connections, when nodes are embedded in two dimensional space, and it is defined as

$$\prod(x, y) = \frac{s_y e^{-d_{xy}/r_c}}{\sum_z s_z e^{-d_{xz}/r_c}},$$

where d_{xy} is the Euclidean distance between node x and y, and r_c is the scaling factor. The network grows same as the BBV model using the proposed probability function.

Mukherjee and Manna [70] proposed a weighted evolving model where nodes are connected to the nearest link. The network starts with two random points on the geographic plain connected by an edge. As the network grows, nodes are placed randomly on the plain and connected to the network. To connect a node, the nearest edge is selected, and its one of the two endpoints is connected with equal probability. The edge weight depends non-linearly on the Euclidean distance. The strength of a node can have a linear or non-linear correlation with edge weight and defined as

$$s_x = \sum_y w_{xy}^{\alpha},$$

where α is a constant parameter to control the impact of edge weights on the node strength. This model follows all power-law distributions observed in real-world weighted networks.

Wenhai et al. [71] designed an evolving model for geographical networks. This model starts with a starting configuration of n_0 nodes connected by m_0 edges. Nodes are distributed on the plain uniformly at random. At each timestamp, a new node x is added and will have m edges; the probability of an existing node y to attract new connection is given as

$$\prod(x, y) = \frac{s_y f(D_{xy})}{\sum_z s_z f(D_{xz})},$$

where $f(D_{xy})$ is the cost function to establish a connection between node x and y, and it depends on the Euclidean distance D_{xy} of its endpoints, $f(D_{xy}) \propto (D_{xy})^{-\alpha}$. At each timestamp, all possible edges update their weights using the strength coupling mechanism [43].

9.3.13 Age Weighted Model

In unweighted networks, when a new node joins and makes connections, equal importance is given to all the links. However, in real life, all these links are not equally important. If a person is making a connection with an older node, it can be more important than a connection with a newer node.

To explain this phenomenon, Jeżewski [72] presented a model where edge weight will depend on the time interval of the birth time of both the endpoint nodes. In this model, a new node x is added at every timestamp t with birth time t_x using degree preferential attachment. The weight w_{xy} depends on origin time of both endpoints as t_x and t_y. So, if $t_x > t_y$, edge weight w_{xy} is

$$w_{xy} = c_N^{-1} t_{xy}^{\sigma_x},$$

and

$$t_{xy} = t_x - t_y,$$
$$\sigma_x = a + b \cdot r_x, r_x \epsilon [0, 1],$$
$$c_N = c_N^{a+b\langle r_x \rangle_N}, c > 0,$$

where N denotes the total number of nodes at timestamp t, and parameters a, b, and c are independent of total nodes N. $\langle r_x \rangle_N$ is average of all r_x so $\langle r_x \rangle_N = (\sum_{x \epsilon N} r_x)/N$. A user can generate different scale-free power-law networks using different values of weight exponent σ_x and the factor c_N. The author further studied weighted networks and proposed an evolving model where edge weight w_{xy} depends on the degree of its endpoints $w_{xy} = (k_x k_y)^\theta$ and $\theta \in (-1, 0]$ [73]. This model was motivated by weak interactions of highly connected nodes where a higher degree node pays less attention to its neighbor nodes and a lower degree node pays more attention and have strong bonds [74, 75]. The degrees and edge weights show power-law distribution in the generated networks.

Tian and Shi [76] presented a model based on the ranking of the nodes. They defined the probability function of a node x to get a new connection as

$$\prod(x) = \frac{R_x^{-\alpha}}{\sum_y R_y^{-\alpha}},$$

where R_x is the ranking of the node x. The rest of the model works as the basic BBV model except the probability function. The authors showed a high correlation between the node strength and age-based ranking of the nodes.

In 2009, Zhou et al. [77] proposed an age-based model using mutual selection approach. The model starts with n_0 isolated nodes and the age is initialized as $h = 1$. At each timestamp, a new node is added and each existing node selects m other nodes. Node x selects node y with the probability

$$\prod(x, y) = \frac{h_y^{-\alpha}}{\sum_z h_z^{-\alpha} - h_x^{-\alpha}},$$

where $\alpha > 0$ and controls the effect of node strength in getting new connections. By varying the value of α, one can generate assortative or disassortative networks. Growth is based on mutual selection, so connections are not updated unless two nodes select each other mutually. In these networks, a younger node has a higher age coefficient h and has a low probability to get more connections.

Wen et al. [78] combined the aging phenomenon with the concept of the local world. In this model, the authors determined the local world as in [59]. At each iteration, M nodes are selected uniformly at random to constitute the local world of the new node. Then a new node x is added to the network with m edges, and m is chosen randomly from the given range $m \epsilon [1, q]$. Every new age will be connected to a node y using the strength age preferential attachment probability

$$\prod(x, y) = \frac{s(t_x, t) \cdot (t - t_x^{-\alpha})}{\sum_{z \epsilon M} s(t_z, t) \cdot (t - t_z^{-\alpha})},$$

where t_x is the birth time of node x and $s(t_x, t)$ denotes the strength of node x at t time. α is a decay factor to control the effect of aging, and $\alpha \epsilon [0, 1)$. Weights are rearranged after adding each node. This model can be efficiently used to reproduce Internet or citation networks.

9.4 Signed Weighted Networks

In real-world networks, all relationships cannot be captured by positive edge weights. In some networks, such as frenemy networks or trust–distrust networks, there are two kinds of edges, positive and negative, that denote the positivity or negativity of the relationship, respectively. For example, in social networks, users have a positive or negative opinion about a person based on the bonding or the first impression. Researchers have studied such networks to understand their evolution and the evolution of positive and negative edge weights. An analysis of these networks shows whether the environment is hostile or harmonious. For an organization or institution, a harmonious environment is essential and desired. Researchers have proposed several models to generate synthetic signed networks that we will discuss next.

In 2004, Hu et al. [79, 80] proposed a model to understand students' relationship in a class. The model starts with N nodes, where everyone knows every other person. The strength of the relationships is stored in $n \times n$ adjacency matrix. Positive edge weights show harmonious relations, and negative edge weights show hostile relations. The model tries to capture the impact of several encounters between two people. In the modeling, the value 1 is assigned with probability p and -1 value is assigned with $(1 - p)$ probability to the elements of the matrix. At every time step, the following steps are executed:

1. A node x is selected randomly and x chooses one other node y for the interaction using the following given probability function:

$$P(xy) = \frac{|w_{xy}| + 1}{\sum_y (|w_{xy}| + 1)}.$$

2. After choosing x and y, w_{xy} is changed by ± 1 with some probability. For getting the value of this probability, a parameter γ_{xy} is defined based on the similarity of their relationships. If two people have more number of mutual friends or enemies, it is more likely that they will become good friends.

$$\gamma_{xy} = C^{-1} \sum_z w_{xz} \cdot w_{zy},$$

where

$$C = \sqrt{\sum_{z} w_{xz}^2} \cdot \sqrt{\sum_{z'} w_{yz'}^2}.$$

So, $\gamma_{xy} \in [-1, +1]$. If $\gamma_{xy} > 0$, the value of edge weight is increased by 1 with probability γ_{xy} and if $\gamma_{xy} < 0$, the value of edge weight is decreased by 1 with probability $|\gamma_{xy}|$.

The harmoniousness of the whole network is computed using

$$\sum_{x,y,w_{xy}>0} \frac{w_{xy}}{\sum_{x,y} |w_{xy}|}.$$

Similarly, the hostility of the network can be calculated. The authors also observed the critical value of p, and above this value, the environment starts to converge to more harmony. They further showed that this model follows real-world properties.

9.5 Mesoscale Structured Weighted Networks

Real-world networks have mesoscale structural properties, such as community and core–periphery structure. Next, we will discuss evolving models to generate mesoscale structured weighted networks.

9.5.1 Community Structured Weighted Networks

In complex networks, a community refers to a group of nodes having dense connections with each other and sparse connections outside. In community structured networks, links can be categorized as (i) *intra-community link* if both end nodes belong to the same community, otherwise (ii) *inter-community link*.

Li and Chen [81] proposed the first model having inbuilt community structure using the inner-community and inter-community preferential attachment rules. Let's assume M denotes the total number of community in a network. The preferential attachment rule is defined as

1. **Inner-Community Preferential Attachment**: If a node is chosen in a randomly selected community, we only consider inner-community degree for the preferential attachment. k_{xi}^1 is the inner degree of node x in community i. So, the node x is chosen with probability

$$\prod(k^1_{xi}) = \frac{k^1_{xi}}{\sum_y k^1_{yi}}.$$

2. **Inter-Community Preferential Attachment**: When a new node is making connections with nodes of other communities, it only considers inter-community degree of that node. k^2_{xj} is the inter-degree of node x in community j. A node x in community j will be chosen with the probability

$$\prod(k^2_{xj}) = \frac{k^2_{xj}}{\sum_{m,n,n\neq i} k^2_{m,n}}.$$

Similarly, the preferential strengthening rules are defined as

1. **Inner-Community Strengthening**: The probability to choose nodes x and y from community i is computed as

$$\prod(x, y, i) = \frac{w^1_{xyi}}{\sum_{m,n} w^1_{mni}}.$$

2. **Inter-Community Strengthening**: The probability to choose two existing node x and y belonging to two different communities i and j, respectively, is computed as

$$\prod(x, i, y, j) = \frac{w^2_{xiyj}}{\sum_{m,p,n,q} w^2_{mpnq}}.$$

The model works in the following manner:

1. **Initialization**: The network starts with M seed communities, each having m_0 fully connected nodes. These communities are connected with each other using $M(M-1)/2$ inter-community edges, where each community is connected to $(M-1)$ other communities.
2. **Growth**:
   (a) With some probability α, a community is selected uniformly at random, and a new vertex is added to it using the inner-community preferential attachment rule. With some probability β, this vertex will also make some inter-community links. All added links have weight 1.
   (b) With probability $1-\alpha$, there will be added no new vertex in the network, and a new intra-community (also referred to as inner-community) link is added. With probability η, a new inter-community link is added. Both links are added with weight 1 and using preferential strengthening rule.

The value of β and η is small, so that inter-community connections are sparse. In this model, inner degree, inter-degree, total degree, link weights, and strength follow a power-law distribution.

Zhou et al. [82] presented a model where the community size also follows power-law distribution as observed in real-world networks [83]. This model works on the steps mentioned below.

1. The network is initialized as in [81] model.
2. **Growth Phase**: In the growing phase, a new vertex or a new community will be added as described in steps (a) and (b).

   (a) A new vertex is added with probability p, and it makes m connections in the network. The newly added vertex makes the following two types of connections.

      i **PA Links**: A new vertex x first chooses a community i using community size preferential attachment (bigger the size of the community, higher the probability to be selected). The community size preferential attachment probability is defined as

      $$\prod(S_i) = \frac{S_i}{\sum_j S_j},$$

      where S_i is the size of community i. Then, it chooses a vertex y in community i using PA rule. This selected community creates the local world for the newly added vertex.

      ii **TF Links**: The vertex x makes triad-formation links with neighbors of y using PA. The vertex x makes one PA connection and remaining $(m - 1)$ are PA connections with probability φ or TF connections with probability $(1 - \varphi)$. PA connections are formed as intra-community connection with probability q and inter-community connection with probability $(1 - q)$. In both steps (i and ii), all links are added with weight $w_0 = 1$.

   (b) A new community is added to the evolving network with probability $(1 - p)$. One vertex is chosen randomly from this new community and makes m connections with other communities. The links are added in the same way as in the above steps, except no inner-community link will be added.

3. **Weight Rearrangement**: Each new link (x, y) creates extra traffic δ, and it is distributed proportionally among all outgoing edges of vertex y.

9.5.2 Core–Periphery Structured Weighted Networks

In a network, the nodes follow a hierarchical order that will evolve the core–periphery structure. The core is a dense central nucleus of the network. For example, in a co-authorship network, the core vertices are pioneer researchers of the area. The core nodes are highly connected with each other and also highly connected with periphery nodes of the network.

Saxena and Iyengar [84] studied the evolution of core in social networks and observed that periphery nodes make a high number of connections with other core nodes to move into the core of the network. They further proposed an evolving model to generate weighted networks having both community as well as core–periphery structure based on their observations. In the proposed model, the network evolves using the PA and TF (triad-formation) method. The network starts with a seed graph having c communities, and each community has a clique of n_0 nodes. All nodes in the seed graph are considered core nodes. At each timestamp, the following steps are executed.

1. A community is selected uniformly at random, and a new periphery node x will join to this community. The node x makes $m \cdot f$ intra-community connections using intra-community strength preferential attachment and $m \cdot (1 - f)$ inter-community connections using inter-community strength preferential attachment, where m denotes the total connections created by x, and f denotes the fraction of intra-community connections. When nodes x and y are connected, the BBV weight distribution function is used to balance the weights.
2. With probability q, a periphery node y is selected using the strength preferential attachment rule to move to the core of the network. The node y is connected with each core node with probability p.
3. Each existing node in the network makes triad-formation connections with probability r using the strength preferential attachment rule. If the selected node already has a connection with the given node, then their edge weight is increased by w_0.

Each new connection is initialized with weight $w_0 = 1$. In the proposed model, step 1 regulates the topological growth, and steps 2 and 3 regulate the self-growth of the network. Specifically, step 2 manages the evolution of the global core. The probability p controls the density of the core, and so it has a high value. The probability q controls the size of the core, and the probability r controls the overall self-growth and has a small value. The authors showed that the generated networks have characteristics observed in real-world networks. The other models to generate mesoscale structured networks are [85, 86].

9.6 Multilayered Weighted Networks

Multilayered networks are used to represent an environment where each node interacts with others in diverse contexts. Each layer represents a single interacting context, such as friendship, colleagues, family, etc. The nodes in different layers are the same, and the relationship in one layer is corresponding to one context. A multilayered network can be represented as $G = (G_1, G_2, G_3, \ldots G_k)$, where each graph G_i $(1 \leq i \leq n)$ represents a single-layered network and can be stored using weighted adjacency matrix.

Murase et al. [87] proposed a model to design multilayered weighted networks while maintaining the spatial constraint and Granovetter structure. Each layer of the network is designed using the single-layer generative model proposed in [88]. The global attachment follows the spatial preferential attachment, and two nearby nodes have a higher probability of being connected. The authors used the percolation method to study these networks' properties and observed that geographical proximity plays a crucial role in designing such networks. Therefore, in [89], authors studied the correlation between network topology and edge weights in multilayered networks and proposed a framework to design null models of multiplex networks.

9.7 Other Miscellaneous Models

Here, we cover models that cannot be categorized in the above categories.

9.7.1 Random Weight Assignment Model

Park et al. [90] proposed a very simple and minimal *random weight assignment model* as an extension of the BA model to design weighted networks. This model is designed on the concept of node betweenness and the scaling of betweenness with nodes' strength. The model first generates a scale-free unweighted network using the Barabasi–Albert model [21] of n nodes. To convert it to the weighted network, first, we assign the number $0 - n$ to each node randomly and then divide this value by n to normalize all values in the preferred range [0, 1]. Now, the weight of a link w_{xy} is computed as $(w_x + w_y)/2$. This edge weight value shows the traffic transferred through the given edge when traffic flows between every two nodes using optimal path [91]. To improve the quality of this model, the nodes' weight can be assigned using the intrinsic properties of the nodes.

9.7.2 Weighted Stochastic Block Model

Aicher et al. [92] studied *Weighted Stochastic Block Model* (WSBM) where edge weights are picked from an exponential family distribution. This model depends on two parameters z and p, where z is a vector of size n containing block assignment of each node and p is a $n \times n$ matrix containing the probabilities p_{xy} of connecting two nodes x and y. Different values of p will generate assortative, disassortative, multi-partite, hierarchical, or core–periphery structure.

9.7.3 Edge Preferential Attachment Model

Chen and Chen [93] proposed a model by considering *edge preferential attachment*, where a highly weighted edge is more powerful to attract more connection. At every time step, an edge xy is selected preferentially and a new node is connected with its both endpoints. The preferential probability to select an edge xy is defined as

$$\prod(xy) = \frac{(1-q)w_{xy} + q}{\sum_{mn}((1-q)w_{mn} + q)},$$

where $q \in [0, 1]$ and it helps to control the power-law exponent of the strength, degree, and edge-weight distributions. The weight of the selected connection (x, y) is increased by one and newly connected edges have weight 1. If an edge (x, y) is selected preferentially, the degree of its endpoints is increased by 1 and the strength of its endpoints is increased by 2. So, this model provides a linear correlation between strength and degree given as $s_x = 2k_x - 2$.

9.7.4 Weighted Fractal Networks

Carletti and Righi [94] presented the first detailed evolving model for *Weighted Fractal Networks* (WFN). These networks are characterized by two main parameters s and f, where, s ($s > 0$) is the number of copies, and f ($0 < f < 1$) is the scaling factor. The network starts with an initial weighted graph G_0 and a family of WFN is constructed using predefined mapping $G_k = T_{s,f,a}(G_{k-1})$. $T_{s,f,a}$ is a mapping function that depends on the scaling factor, the number of copies, and a labeled node a that is also called attaching node. In WFN, the node strength has a power-law probability distribution, and its exponent is the Hausdorff dimension [95]. They further studied non-homogeneous WFN, where different scaling is defined for all edges. They extended this work and proposed deterministic non-homogeneous and stochastic weighted fractal network models [96].

9.7.5 Capacity Constraint Model

Wu et al. [97] presented a model based on the capacity constraint of the nodes. A vertex cannot be connected with more than its capacity. For each vertex, its attractive index $A_x(t)$ at time t is defined using its capacity C_x as

$$A_x(t) = s_x(t)\left(1 - \frac{s_x(t)}{C_x}\right).$$

At each timestamp, a new vertex will be added and will make m connections using the attractive index preferential attachment. The increased weight is distributed proportionally to the attractiveness of the other endpoint.

9.7.6 Overlapped Clique Evolution Model

Yang et al. [98] presented a *weighted clique evolving model* for overlapped clique growth. This model starts with c_0 cliques, each having n_0 vertices, and all the edges have weight $w_0 = 1$. At each iteration, a new clique c is added, and it has n nodes. A new clique contains n_1 old nodes and n_2 new nodes; old nodes are selected using the following selection mechanism:

1. **Preferential selection mechanism**: Probability to select a vertex x depends on the total number of cliques h_x that a vertex joins. It is defined as

$$\prod(x) = \frac{h_x}{\sum_z h_z}.$$

2. **Random selection mechanism**: An old vertex is selected randomly from all the existing vertices.

The distribution of the number of cliques joined by a vertex $P(h)$ follows a power law. They also studied three bus transportation networks of Shanghai, Hangzhou, and Beijing, China, and showed that they are similar to overlapping clique networks generated using the proposed model. [99, 100] also studied similar weighted clique evolution models.

9.8 Conclusion

In this chapter, we have presented an organized literature review on the evolution of different kinds of weighted networks and evolving models to explain the underlying evolutionary mechanisms. The chapter presented a systematic review of different modeling frameworks. We further discussed the modeling phenomenon for inbuilt mesoscale properties, such as community structure or hierarchical organization. We also covered different modeling approaches that are proposed to generate real-world networks having special properties. The chapter will help better understand the structural properties of real-world networks and dynamic phenomenon taking place on these weighted networks.

References

1. Banavar, J.R., Colaiori, F., Flammini, A., Maritan, A., Rinaldo, A.: Topology of the fittest transportation network. Phys. Rev. Lett. **84**(20), 4745 (2000)
2. Onnela, J.-P., Saramäki, J., Hyvönen, J., Szabó, G., De Menezes, M.A., Kaski, K., Barabási, A.-L., Kertész, J.: Analysis of a large-scale weighted network of one-to-one human communication. New J. Phys. **9**(6), 179 (2007)
3. De Montis, A., Barthélemy, M., Chessa, A., Vespignani, A.: The structure of inter-urban traffic: a weighted network analysis (2005). arXiv:physics/0507106
4. Bagler, G.: Analysis of the airport network of India as a complex weighted network. Phys. A **387**(12), 2972–2980 (2008)
5. Granovetter, M.S.: The strength of weak ties. Am. J. Sociol. 1360–1380 (1973)
6. Lin, N., Ensel, W.M., Vaughn, J.C.: Social resources and strength of ties: structural factors in occupational status attainment. Am. Sociol. Rev. 393–405 (1981)
7. Brown, D.W., Konrad, A.M.: Granovetter was right the importance of weak ties to a contemporary job search. Group & Organ. Manag. **26**(4), 434–462 (2001)
8. Wegener, B.: Job mobility and social ties: social resources, prior job, and status attainment. Am. Sociol. Rev. 60–71 (1991)
9. Ashman, D., Brown, L.D., Zwick, E.: The strength of strong and weak ties. Nonprofit Manag. Leadersh. **9**(2), 153–172 (1998)
10. Kunegis, J., Lommatzsch, A., Bauckhage, C.: The slashdot zoo: mining a social network with negative edges. In: Proceedings of the 18th International Conference on World Wide Web, pp. 741–750. ACM (2009)
11. Brzozowski, M.J., Hogg, T., Szabo, G.: Friends and foes: ideological social networking. In: Proceedings of the SIGCHI Conference on Human Factors in Computing Systems, pp. 817–820. ACM (2008)
12. Guha, R., Kumar, R., Raghavan, P., Tomkins, A.: Propagation of trust and distrust. In: Proceedings of the 13th international conference on World Wide Web, pp. 403–412. ACM (2004)
13. Ziegler, C.-N., Lausen, G.: Propagation models for trust and distrust in social networks. Inf. Syst. Front. **7**(4–5), 337–358 (2005)
14. Li, M., Fan, Y., Chen, J., Gao, L., Di, Z., Jinshan, W.: Weighted networks of scientific communication: the measurement and topological role of weight. Phys. A **350**(2), 643–656 (2005)
15. Jeżewski, W.: Scale-free properties of weighted networks with connectivity-driven topology. Phys. A **354**, 672–680 (2005)
16. Latora, V., Marchiori, M.: Economic small-world behavior in weighted networks. Eur. Phys. J. B-Condens. Matter Complex Syst. **32**(2), 249–263 (2003)
17. Opsahl, T., Panzarasa, P.: Clustering in weighted networks. Soc. Netw. **31**(2), 155–163 (2009)
18. Arenas, A., Fernández, A., Gómez, S.: Analysis of the structure of complex networks at different resolution levels. New J. Phys. **10**(5), 053039 (2008)
19. Newman, M.E.J.: Assortative mixing in networks. Phys. Rev. Lett. **89**(20), 208701 (2002)
20. Yook, S.-H., Jeong, H., Barabási, A.-L.: Weighted evolving networks. Phys. Rev. Lett. **86**(25), 5835 (2001)
21. Barabási, A.-L., Albert, R.: Emergence of scaling in random networks. Science **286**(5439), 509–512 (1999)
22. Antal, T., Krapivsky, P.L.: Weight-driven growing networks. Phys. Rev. E **71**(2), 026103 (2005)
23. Bianconi, G.: Emergence of weight-topology correlations in complex scale-free networks. EPL (Europhys. Lett.) **71**(6), 1029 (2005)
24. Mukherjee, G., Manna, S.S.: The weighted scale-free network with self-organizing link weight dynamics. J. Stat. Mech: Theory Exp. **2006**(05), L05001 (2006)
25. Bianconi, G., Barabási, A.-L.: Competition and multiscaling in evolving networks. EPL (Europhys. Lett.) **54**(4), 436 (2001)
26. Wang, S., Zhang, C.: Weighted competition scale-free network. Phys. Rev. E **70**(6), 066127 (2004)

27. Zheng, D., Trimper, S., Zheng, B., Hui, P.M.: Weighted scale-free networks with stochastic weight assignments. Phys. Rev. E **67**(4), 040102 (2003)
28. Riccaboni, M., Schiavo, S.: Structure and growth of weighted networks. New J. Phys. **12**(2), 023003 (2010)
29. Yamasaki, K., Matia, K., Buldyrev, S.V., Fu, D., Pammolli, F., Riccaboni, M., Stanley, H.E.: Preferential attachment and growth dynamics in complex systems. Phys. Rev. E **74**(3), 035103 (2006)
30. Barrat, A., Barthélemy, M., Vespignani, A.: Weighted evolving networks: coupling topology and weight dynamics. Phys. Rev. Lett. **92**(22), 228701 (2004)
31. Hu, B., Yan, G., Wang, W.-X., Chen, W: Generalized BBV models for weighted complex networks (2005). arXiv:nlin/0505041
32. Goh, K.I., Kahng, B., Kim, D.: Traffic-driven evolution of weighted scale-free networks (2004). arXiv:cond-mat/0410078
33. Goh, K.-I., Kahng, B., Kim, D.: Universal behavior of load distribution in scale-free networks. Phys. Rev. Lett. **87**(27), 278701 (2001)
34. Saxena, A., Iyengar, S. Centrality measures in complex networks: a survey (2020). arXiv:2011.07190
35. Wu, Z.-X., Xu, X.-J., Wang, Y.-H.: Generating structured networks based on a weight-dependent deactivation mechanism. Phys. Rev. E **71**(6), 066124 (2005)
36. Klemm, K., Eguiluz, V.M.: Highly clustered scale-free networks. Phys. Rev. E **65**(3), 036123 (2002)
37. Pandya, R.V.R.: A note on (2004). arXiv:cond-mat/0406644
38. Tian, L., Shi, D.-N., Zhu, C.-P.: The effect of generalized deactivation mechanism in weighted networks. Phys. A **380**, 611–620 (2007)
39. Zhang, Z., Fang, L., Zhou, S., Guan, J.: Effects of accelerating growth on the evolution of weighted complex networks. Phys. A **388**(2), 225–232 (2009)
40. Wang, L., Zhang, G.Q., Chen, T.L.: An extensive weight-driven network with non-linear growth information. EPL (Europhys. Lett.) **84**(5), 58006 (2008)
41. Sen, Q., Guan-Zhong, D., Lin, W., Ming, F.: A weighted network model with accelerated evolution (2007)
42. Wang, W.-X., Bo, H., Zhou, T., Wang, B.-H., Xie, Y.-B.: Mutual selection model for weighted networks. Phys. Rev. E **72**(4), 046140 (2005)
43. Xie, Y.-B., Wang, W.-X., Wang, B.-H.: Modeling the coevolution of topology and traffic on weighted technological networks. Phys. Rev. E **75**(2), 026111 (2007)
44. Mu, J., Sun, H., Pan, J., Zhou, J.: A novel evolving network model with widely weighted dynamics. In: 2010 8th World Congress on Intelligent Control and Automation (WCICA), pp. 3034–3039. IEEE (2010)
45. Tanaka, T., Aoyagi, T.: Scale-free networks with self-growing weight (2007). arXiv:cond-mat/0701314
46. Tanaka, T., Aoyagi, T.: Weighted scale-free networks with variable power-law exponents. Phys. D **237**(7), 898–907 (2008)
47. Leung, C.C., Chau, H.F.: Weighted assortative and disassortative networks model. Phys. A **378**(2), 591–602 (2007)
48. Hao, B., Zhou, Y., Jing, Y., Zhang, S.: General BBV model of weighted complex networks. In: International Conference on Communication Software and Networks. ICCSN'09, pages 295–298. IEEE, 2009
49. Zhang, B., Xie, Y., Zhang, X., Wang, D.: A weighted network model based on triad formation mechanism. In: 2010 International Conference on Intelligent Computation Technology and Automation, pp. 390–392. IEEE (2010)
50. Jing, Y., Hao, B., Yu, H., Zhang, S.: A general BBV model with high clustering scale-free networks. In: 2009 International Conference on Future Networks, pp. 57–60. IEEE (2009)
51. Wang, D., Li, Y., Jin, X.: Effect of complex network structure on synchronizability. In: 2012 10th World Congress on Intelligent Control and Automation (WCICA), pp. 3481–3484. IEEE (2012)

52. Yang, C.-X., Tang, M.-X., Tang, H.-Q., Deng, Q.-Q.: Local-world and cluster-growing weighted networks with controllable clustering. Int. J. Mod. Phys. C **25**(05), 1440009 (2014)
53. Li, S.L., Wang, D., Zhang, L.: A weighted scale-free network with triad formation. In: Advanced Materials Research, vol. 225, pp. 442–445. Trans Tech Publications (2011)
54. Hu, Y., Zhu, D., Zhu, N.: A weighted network evolution model based on passenger behavior (2007). arXiv:0709.4143
55. Hu, Y., Zhu, D., Zhu, N.: Weighted network evolution with transfer behavior. In: International Conference on Information Management, Innovation Management and Industrial Engineering. ICIII'08, vol. 2, pp. 532–535. IEEE (2008)
56. Dai, M., Rongrong, L.: a weighted network evolution with traffic flow under preferential attachment. Int. J. Mod. Phys. C **24**(04), 1350020 (2013)
57. Zheng, J.-F., Gao, Z.-Y.: A weighted network evolution with traffic flow. Phys. A **387**(24), 6177–6182 (2008)
58. Li, X., Chen, G.: A local-world evolving network model. Phys. A **328**(1), 274–286 (2003)
59. Pan, Z., Li, X., Wang, X.: Generalized local-world models for weighted networks. Phys. Rev. E **73**(5), 056109 (2006)
60. Sun, X., Feng, E., Li, J.: From unweighted to weighted networks with local information. Phys. A **385**(1), 370–378 (2007)
61. Barahona, M., Pecora, L.M.: Synchronization in small-world systems. Phys. Rev. Lett. **89**(5), 054101 (2002)
62. Zhang, Z., Rong, L., Wang, B., Zhou, S., Guan, J.: Local-world evolving networks with tunable clustering. Phys. A **380**, 639–650 (2007)
63. Holme, P., Kim, B.J.: Growing scale-free networks with tunable clustering. Phys. Rev. E **65**(2), 026107 (2002)
64. Li, M., Jinshan, W., Wang, D., Zhou, T., Di, Z., Fan, Y.: Evolving model of weighted networks inspired by scientific collaboration networks. Phys. A **375**(1), 355–364 (2007)
65. Wang, W.-X., Bo, H., Wang, B.-H., Yan, G.: Mutual attraction model for both assortative and disassortative weighted networks. Phys. Rev. E **73**(1), 016133 (2006)
66. Guo, J.-L., Fan, C., Ji, Y-L.: Universality of competitive networks for weighted networks (2011). arXiv:1101.1638
67. Jin-Li, G.: Co-evolution of the brand effect and competitiveness in evolving networks. Chin. Phys. B **23**(7), 070206 (2014)
68. Guo, J., Fan, C., Ji, Y.: Universality of competitive networks in complex networks. J. Syst. Sci. Complex. **28**(3), 546–558 (2015)
69. Barrat, A., Barthélemy, M., Vespignani, A.: The effects of spatial constraints on the evolution of weighted complex networks. J. Stat. Mech: Theory Exp. **2005**(05), P05003 (2005)
70. Mukherjee, G., Manna, S.S.: Weighted scale-free networks in Euclidean space using local selection rule. Phys. Rev. E **74**(3), 036111 (2006)
71. Wenhai, X., Siying, Z., Qisheng, G.: Weighted evolving network with geographical constraints. In: Control and Decision Conference, 2008. CCDC 2008. Chinese, pp. 4814–4817. IEEE (2008)
72. Jeżewski, W.: Scaling in weighted networks and complex systems. Phys. A **337**(1), 336–356 (2004)
73. Jeżewski, W.: On the emergence of scaling in weighted networks. Phys. A **379**(2), 691–700 (2007)
74. Indekeu, J.O.: Special attention network. Phys. A **333**, 461–464 (2004)
75. Giuraniuc, C.V., Hatchett, J.P.L., Indekeu, J.O., Leone, M., Castillo, I.P., Van Schaeybroeck, B., Vanderzande, C.: Trading interactions for topology in scale-free networks. Phys. Rev. Lett. **95**(9), 098701 (2005)
76. Tian, L., Shi, D.-N.: Rank-based model for weighted network with hierarchical organization and disassortative mixing. Eur. Phys. J. B **56**(2), 167–171 (2007)
77. Zhou, Y.-B., Cai, S.-M., Wang, W.-X., Zhou, P.-L.: Age-based model for weighted network with general assortative mixing. Phys. A **388**(6), 999–1006 (2009)

78. Wen, G., Duan, Z., Chen, G., Geng, X.: A weighted local-world evolving network model with aging nodes. Phys. A **390**(21), 4012–4026 (2011)
79. Hu, B., Jiang, X.-Y., Ding, J.-F., Xie, Y.-B., Wang, B.-H.: A model of weighted network: the student relationships in a class (2004). arXiv:cond-mat/0408125
80. Bo, H., Jiang, X.-Y., Ding, J.-F., Xie, Y.-B., Wang, B.-H.: A weighted network model for interpersonal relationship evolution. Phys. A **353**, 576–594 (2005)
81. Li, C., Chen, G.: Modelling of weighted evolving networks with community structures. Phys. A **370**(2), 869–876 (2006)
82. Zhou, X., Xiang, L., Xiao-Fan, W.: Weighted evolving networks with self-organized communities. Commun. Theor. Phys. **50**(1), 261 (2008)
83. Clauset, A., Newman, M.E.J., Moore, C.: Finding community structure in very large networks. Phys. Rev. E **70**(6), 066111 (2004)
84. Saxena, A., Iyengar, S.R.S.: Evolving models for meso-scale structures. In: 2016 8th International Conference on Communication Systems and Networks (COMSNETS), pp. 1–8. IEEE (2016)
85. Saxena, A.,Iyengar, S.R.S., Gupta, Y.: Understanding spreading patterns on social networks based on network topology. In: Proceedings of the 2015 IEEE/ACM International Conference on Advances in Social Networks Analysis and Mining 2015, pp. 1616–1617 (2015)
86. Gupta, Y., Saxena, A., Das, D., Iyengar, S.R.S.: Modeling memetics using edge diversity. In: Complex Networks VII, pp. 187–198. Springer (2016)
87. Murase, Y., Török, J., Jo, H.-H., Kaski, K., Kertész, J.: Multilayer weighted social network model. Phys. Rev. E **90**(5), 052810 (2014)
88. Kumpula, J.M., Onnela, J.-P., Saramäki, J., Kertesz, J., Kaski, K.: Model of community emergence in weighted social networks. Comput. Phys. Commun. **180**(4), 517–522 (2009)
89. Menichetti, G., Remondini, D., Bianconi, G.: Correlations between weights and overlap in ensembles of weighted multiplex networks. Phys. Rev. E **90**(6), 062817 (2014)
90. Park, K., Lai, Y.-C., Ye, N.: Characterization of weighted complex networks. Phys. Rev. E **70**(2), 026109 (2004)
91. Braunstein, L.A., Buldyrev, S.V., Cohen, R., Havlin, S., Stanley, H.E.: Optimal paths in disordered complex networks. Phys. Rev. Lett. **91**(16), 168701 (2003)
92. Aicher, C., Jacobs, A.Z., Clauset, A.: Adapting the stochastic block model to edge-weighted networks (2013). arXiv:1305.5782
93. Chen, S., Chen, Q.: A weighted network model based on the preferential selection of edges. Appl. Math. Sci. (Hikari) **1**(21–24), 1145–1156 (2007)
94. Carletti, T., Righi, S.: Weighted fractal networks. Phys. A **389**(10), 2134–2142 (2010)
95. Mandelbrot, B.B.: The Fractal Geometry of Nature/Revised and Enlarged Edition, vol. 1, 495 p. WH Freeman and Co., New York (1983)
96. Carletti, T.: Stochastic weighted fractal networks (2010). arXiv:1002.0203
97. XiaoHuan, W., Zhu, J.F., WeiWei, W., Ge, W.: A weighted network evolving model with capacity constraints. Sci. China Phys. Mech. Astron. **56**(9), 1619–1626 (2013)
98. Yang, X.-H., Wang, B., Sun, B.: A novel weighted evolving network model based on clique overlapping growth. J. Cent. South Univ. Technol. **17**, 830–835 (2010)
99. Yang, X.-H., Jiang, F.-L., Chen, S.-Y., Wang, W.-L.: Modeling evolution of weighted clique networks. Commun. Theor. Phys. **56**(5), 952 (2011)
100. Li, Z.-W., Yang, X.-H., Jiang, F.-L., Chen, G., Weng, G.-Q., Zhu, M.: Dynamically weighted clique evolution model in clique networks. Math. Probl. Eng. **2013** (2013)

Chapter 10
Learning Graph Representations

Rucha Bhalchandra Joshi and Subhankar Mishra

Abstract Social and information networks are gaining huge popularity recently due to their various applications. Knowledge representation through graphs in the form of nodes and edges should preserve as many characteristics of the original data as possible. Some of the interesting and useful applications on these graphs are graph classification, node classification, link prediction, etc. The Graph Neural Networks have evolved over the last few years. Graph Neural Networks (GNNs) are efficient ways to get insight into large and dynamic graph datasets capturing relationships among billions of entities also known as knowledge graphs. In this paper, we discuss the graph convolutional neural networks graph autoencoders and spatio-temporal graph neural networks. The representations of the graph in lower dimensions can be learned using these methods. The representations in lower dimensions can be used further for downstream machine learning tasks.

10.1 Introduction

The complex networks like social networks, citation networks, etc. can very well be represented in the form of graphs. The questions related to these graphs can be addressed using neural networks. However, feeding the graphs as input to the neural networks needs to be done cleverly. This is due to the vast and complex nature of the graphs. The nodes of the graph need to be represented in lower dimensional vector form so that it can easily be given as input to the neural network to address the problems. These representations should characterize the information contained in the graphs. They should capture the structural as well as the feature information contained in the graph. The representations in lower dimensional space can be then

R. B. Joshi (✉)
National Institute of Science Education and Research, Bhubaneswar 752050, India
e-mail: rucha.joshi@niser.ac.in

S. Mishra
Homi Bhabha National Institute, Anushaktinagar 400094, Mumbai, India
e-mail: smishra@niser.ac.in

A. Biswas et al. (eds.), *Principles of Social Networking*,
Smart Innovation, Systems and Technologies 246,
https://doi.org/10.1007/978-981-16-3398-0_10

fed to a neural network to address the tasks on the graphs. These representations are necessary in order to reduce to some extent the overhead caused due to the large size of the network.

These representations in lower dimensional space are also called the embeddings. Several shallow embedding techniques like DeepWalk [19] and Node2vec [5] were earlier introduced to generate node representations. They considered only the vertex set and the adjacency matrix of the graph in order to generate the embeddings. An encoding function and a similarity function that measures the similarity between similar nodes in the embedding space need to be defined. The parameters are optimized based on this similarity function. The shortcoming of such methods are, however, the large number of parameters to be optimized and also that these methods did not consider the feature information to generate the node embeddings. To overcome these shortcomings, the deep graph neural networks are used. There have been several approaches to produce the representations for nodes which considered the node features in addition to the structure of the node neighborhood. Initial methods are from spectral graph theory, which mostly require matrix factorization. To improve on such methods the spatial based methods were introduced. These methods consider information diffusion and message passing. The neighborhood feature information is aggregated to generate the representation of a particular node. Figure 10.1 is the portrayal of nodes in original graph mapped to their representations in embedding space.

In this paper we review the popular graph neural network methods that are primarily based on neighborhood aggregation. The graph neural networks were introduced by [4]. Then there were several improvements on the same in terms of how the neighborhood aggregation is done. The neighborhood aggregation is looked at as the convolution over the node in the graph. The spectral graph theory has the notion of convolutions on the graph where a graph signal, which is feature vector of a node in

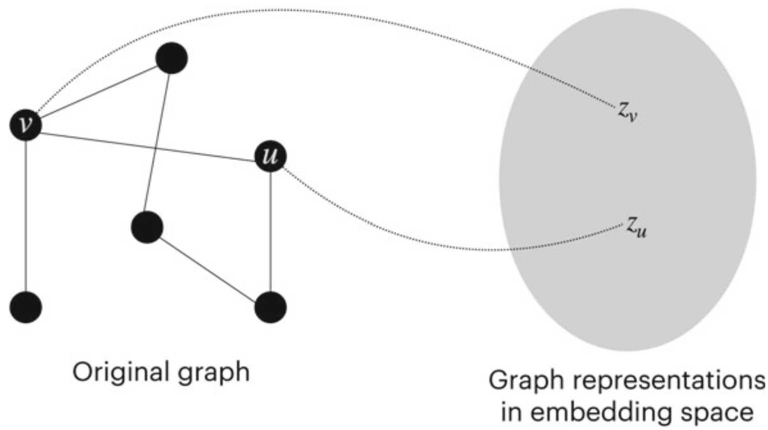


Fig. 10.1 Representing nodes in embedding space. Similar nodes in original graph must be similar in embedding space

this case, is convolved in the Fourier domain. This idea have been used in the spectral based methods whereas the spatial based methods broadly rest on the concept of neighborhood aggregation. The graphs autoencoders aim at reconstructing the original adjacency matrix. The architecture includes an encoder and a decoder. We have discussed the graph autoencoding techniques in this paper. The spatio-temporal graph neural networks are designed to work with the data that can be easily modeled using spatio-temporal graphs. The graph changes over the period of time, depending on the system that is being modeled. The graph neural network used to generate the embedding for such data structure should consider the spatial as well as the temporal dependencies in the graph. We discuss some of the spatio-temporal graph neural network methods in this paper.

The rest of the paper is organized as follows. In Sect. 10.2 we discuss the preliminaries. We discuss spectral and non-spectral graph convolutional methods in Sect. 10.3, autoencoding techniques in 10.4 and spatio-temporal graph neural networks in Sect. 10.5. Section 10.6 is discussions and Sect. 10.7 discusses the future directions. We conclude the paper with Sect. 10.8.

10.2 Preliminaries

A *graph* is represented as $G = (V, E)$ where V is a set of vertices or nodes and E is the set of edges. Let $v_i \in V$ be the ith node in the vertex set V. The edge connecting nodes v_i and v_j is represented as $e_{ij} = (v_i, v_j)$. n is the number of nodes in the graph and m is the number of edges. $\mathbf{A}$ is the adjacency matrix of the graph. The dimension of $\mathbf{A}$ is $n \times n$. The entry $\mathbf{A}_{ij} = 1$ if $e_{ij} \in E$. The neighborhood $N(v)$ of a node v is defined as $N(v) = \{u \in V | (v, u) \in E\}$.

The feature vector corresponding to node v is represented as $\mathbf{x}_v \in \mathbb{R}^d$ where d is the size of node feature vector. The matrix of feature vectors of nodes in a graph is $\mathbf{X} \in \mathbb{R}^{n \times d}$. The hidden representation corresponding to node v is given as $\mathbf{h}_v^k$ in kth layer of the Graph Neural Network. The final representation corresponding to a node in the embedding space is denoted as $\mathbf{z}_v$.

Table 10.1 summarizes the frequently used notations with their descriptions.

10.3 Convolutional Graph Neural Networks

The convolutional graph neural networks draw motivations from the conventional convolutional neural networks (CNNs). These graph neural networks can be divided into two categories based on the principle they work on: the *spectral based* convolutional graph neural networks and the *spatial based* convolutional graph neural networks. The early spectral based methods essentially make use of the adjacency matrix and degree matrix of the graph and perform the convolution operation in the Fourier domain. The convolution filter is applied to the graph signal in the Fourier

Table 10.1 Notations

Notation	Description
G	A graph
V	The set of nodes
E	The set of edges
v_i	A vertex in V
$e_{i,j}$	An edge in E connecting vertices v_i and v_j
n	Number of nodes
m	Number of edges
$\mathbf{A}$	The adjacency matrix of graph G
$\mathbf{D}$	The diagonal matrix of node degrees of graph G
$N(v)$	The neighborhood of v
$\mathbf{x}_v$	Feature vector of node v
$\mathbf{X}$	The feature matrix of graph G
d	The dimension of the node representation in previous layer
d'	The dimension of the node representation generated in current layer
$\mathbf{h}_v^k$	Node v's representation in kth hidden layer
$\mathbf{H}^k$	Matrix of node representations in kth hidden layer
$\mathbf{z}_v$	Node v's representation in the embedding space
$\mathbf{Z}$	Matrix of node representations in the embedding space
$\mathbf{L}$	Laplacian of a matrix
$\mathcal{L}$	Normalized Laplacian of a matrix
$\mathbf{U}$	Matrix of eigenvalues
Λ	Matrix of eigenvectors
$\boldsymbol{\lambda}_i$	ith eigenvectors
θ	Learnable convolutional kernel
Θ	Diagonal matrix of learnable convolutional kernels
σ	Non-linear activation function
$\mathbf{W}_k, \mathbf{B}_k$	Learnable parameters for kth layer
AGG	Some aggregation function
$MEAN$	Element-wise mean aggregator
γ	Element-wise pooling operator
$LSTM$	Long short-term memory aggregator
π	Permutation function
α_{ij}	Attention coefficient between i and j
$LeakyRELU$	Leaky rectified linear unit activation
ϵ	A learnable parameter to determine a node's contribution to the next layer update
p	Distribution of input data
q	Estimation of distribution of latent variable
$\mathbb{E}$	Expectation
$KL(\cdot\|\|\cdot)$	KL divergence between two distributions
$\mathcal{N}(\mu, \sigma)$	Normal distribution with mean μ and standard deviation σ
$\star_G$	Diffusion convolution operator
$r^{(t)}$	Reset gate vector
$u^{(t)}$	Update gate vector
$C^{(t)}$	Candidate gate vector

domain, and then transformed back to the graph domain. The limitation of these kinds of convolutional methods is that these are inherently transductive and hence need the entire graph for the computation. In the transductive setting, the model is specific to the graph it is trained on, and may not be extendable to the graph that has not been seen by the model. These methods also do not use the shared parameters. In contradiction to transductive methods, the inductive graph neural network methods are such that the parameters can be used to generate the embeddings for unseen graphs.

The graph structure is given by the adjacency matrix. Although a particular adjacency matrix is not the only way to represent the graph, there could be multiple adjacency matrices corresponding to a particular graph. Hence, the representation in lower dimensional space should be specific to the graph and not to the adjacency matrix. Irrespective of any permutation of the adjacency matrix of a graph, the graph neural network should give the same representation for a particular task. It is essential that the function generating the representations should either be invariant or equivariant, depending on whether the task at hand is graph level or node level. For the graph level task like graph classification, the graph neural network function needs to be permutation invariant as described in the following equation:

$$f\left(Q\mathbf{A}Q^T, \mathbf{X}\right) = f(\mathbf{A}, \mathbf{X}) \tag{10.1}$$

where Q is a permutation matrix, $\mathbf{A}$ is adjacency matrix of the graph, $\mathbf{X}$ is the feature matrix corresponding to the graph and f denotes the function approximated by the neural network.

For node-level tasks like node classification, the graph neural network function needs to be permutation equivariant as given in the equation below:

$$f\left(Q\mathbf{A}Q^T, \mathbf{X}\right) = Qf(\mathbf{A}, \mathbf{X}) \tag{10.2}$$

10.3.1 Spectral Based

The spectral based methods perform operations on the graph by making use of the adjacency matrix of the graph, Laplacian matrix of the graph, etc. The Laplacian of the graph is the measure of smoothness of it. The *non-normalized graph Laplacian* is defined as $\mathbf{L} = \mathbf{D} - \mathbf{A}$ where $\mathbf{D}$ is the diagonal degree matrix of the graph G defined as $\mathbf{D}_{ii} = \sum_j \mathbf{A}_{ij}$. The *normalized graph Laplacian* is $\mathcal{L} = \mathbf{I}_n - \mathbf{D}^{-1/2}\mathbf{A}\mathbf{D}^{-1/2}$. $\mathcal{L}$ is a real symmetric positive semidefinite matrix. It has a complete set of orthogonal eigenvectors with associated real positive eigenvalues. The matrix can be factorized as $\mathcal{L} = \mathbf{U}\Lambda\mathbf{U}^T$ where $\mathbf{U} = [u_0, u_1, \ldots, u_{n-1}] \in \mathbb{R}^{n\times n}$ is a matrix of eigenvectors arranged according to eigenvalues and Λ is a diagonal matrix with eigenvalues $\{\lambda_0, \lambda_1, \ldots, \lambda_{n-1}\}$ as diagonal elements.

The graph signal is a function $\mathbf{x}$ that maps vertices to $\mathbb{R}$. The *graph Fourier transform* of a given graph signal $\mathbf{x}$ is defined as $\hat{f}(\mathbf{x}) = \mathbf{U}^T\mathbf{x}$ and the *inverse graph*

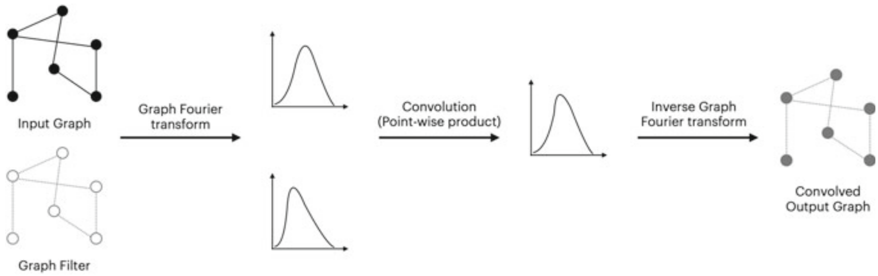


Fig. 10.2 Working of spectral based graph convolutional methods. The graph and the kernel are convolved once they are transformed into the Fourier domain. The result of the convolution is transformed back as the convolved graph by applying the inverse Fourier transform

Fourier transform is given by $f(\hat{\mathbf{x}}) = \mathbf{U}\hat{\mathbf{x}}$. This working of the convolutional graph neural networks is shown in the following figure.

The *convolution* operation in the spectral based method is defined by applying a filter which is a diagonal matrix $\mathbf{g}_\theta = diag(\theta)$, where $\theta \in \mathbb{R}^n$ to the Fourier transform of the signal as follows (Fig. 10.2):

$$\mathbf{g}_\theta * \mathbf{x} = \mathbf{U}\mathbf{g}_\theta\mathbf{U}^T\mathbf{x} \tag{10.3}$$

Spectral graph convolutional networks [1] consider the signal with multiple K channels and hence f_k filters in layer k for $k = 1, \ldots, K$. The input signal to each of the layer is $\mathbf{H}_i^{k-1} \in \mathbb{R}^{n \times f_{k-1}}$ and the output signal generated is $\mathbf{H}_i^k \in \mathbb{R}^{n \times f_k}$. The convolution operation is defined as

$$\mathbf{H}_j^{k+1} = \sigma\left(\mathbf{U}\sum_{i=1}^{f_{k-1}} \Theta_{i,j}^k \mathbf{U}^T\mathbf{H}_j^k\right) \quad (j = 1, \ldots, f_{k-1}) \tag{10.4}$$

where $\Theta_{i,j}^k$ is diagonal matrix of learnable parameters. This operation costs $O(n^3)$.

ChebNet [3] approximates the filter $\mathbf{g}_\theta$ as truncated Chebyshev's polynomials up to K degree of the polynomial of the diagonal matrix of the eigenvalues Λ. Chebyshev's polynomials are recursively defined as $T_0(x) = 1$, $T_1(x) = x$, $T_{k+1}(x) = 2xT_k(x) - T_{k-1}(x)$. The convolutional filter is parameterized as

$$\mathbf{g}_\theta(\Lambda) = \sum_{k=0}^{K} \theta_k T_k(\hat{\Lambda}) \tag{10.5}$$

where $\theta \in \mathbb{R}^K$ is a vector consisting of Chebyshev's coefficients, and $T_k(\hat{\Lambda})$ is defined as $\hat{\Lambda} = \frac{2\Lambda}{\lambda_{max}} - \mathbf{I}_n$. The convolution operation on the graph signal $\mathbf{x}$ using the above filter is hence defined as

$$\mathbf{g}_\theta * \mathbf{x} = \sum_{k=0}^{K} \theta_k T_k(\hat{\mathbf{L}})\mathbf{x} \tag{10.6}$$

where $\hat{\mathbf{L}} = \frac{2\mathcal{L}}{\lambda_{max}} - \mathbf{I}_n$ and $T_k(\hat{\mathbf{L}}) = \mathbf{U}T_k(\hat{\Lambda})\mathbf{U}^T$. ChebNet reduces the complexity of the filtering operation from $O(n^3)$ to $O(Km)$.

CayleyNets [11] define the convolution filter as follows:

$$\mathbf{x} * \mathbf{g}_\theta = c_0\mathbf{x} + 2Re\left\{\sum_{j=0}^{r} c_j(h\mathcal{L} - i\mathbf{I})^j(h\mathcal{L} + i\mathbf{I})^{-j}\mathbf{x}\right\} \tag{10.7}$$

where $\mathcal{L}$ is a normalized graph Laplacian matrix, $Re(.)$ gives the real part of the Cayley's polynomial, c and h are learnable parameters.

The filters proposed by *Graph Convolutional Networks* (GCN) [9] convolve over the graph and limit the K value from ChebNet to 2. Having the larger value of K may over-smoothen the information in the node, by aggregating too much information from the neighborhood nodes. The convolution operation for GCN is

$$\mathbf{g}_\theta * \mathbf{x} = \theta_0\mathbf{x} + \theta_1\mathbf{D}^{-\frac{1}{2}}\mathbf{A}\mathbf{D}^{-\frac{1}{2}}\mathbf{x} \tag{10.8}$$

where θ_0 and θ_1 are learnable parameters. Equation 10.8 can be written in matrix form as follows:

$$\mathbf{H} = \tilde{\mathbf{D}}^{\frac{1}{2}}\tilde{\mathbf{A}}\tilde{\mathbf{D}}^{\frac{1}{2}}\mathbf{X}\Theta \tag{10.9}$$

where $\tilde{\mathbf{A}} = \mathbf{A} + \mathbf{I}_n$ and $\tilde{\mathbf{D}}_{ii} = \sum_j \tilde{\mathbf{A}}_i j$. The identity matrix is added to adjacency matrix in order to include the contribution from the node itself.

Dual Graph Convolutional Networks (DGCN) introduced by [27] applies two graph convolutions on the same inputs in order to capture the local and the global consistencies. It uses Positive Pointwise Mutual Information (PPMI) matrix for encoding the information.

10.3.2 Non-Spectral Based

The graph neural networks that are not spectral based primarily use the principle of message passing. The node feature information is shared across the neighborhood of the node. The methods differ in the way they apply the aggregations over the neighborhood (Fig. 10.3).

NN4G [16] is a constructive feedforward neural network. The neighborhood information is summed up which is why it causes the hidden states to have different scales.

The GCN model aggregates neighborhood features and reduces the impact of high degree neighboring nodes. Equation 10.8 can also be viewed as follows for each of the nodes v in kth layer:

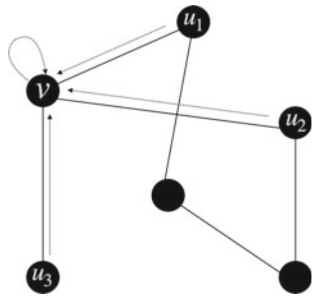


Fig. 10.3 Neighborhood aggregations. To generate the representation of vertex (e.g. v), the spatial based methods consider the contributions from the neighborhood nodes (u_1, u_2, u_3 in this example). The methods specify how the neighborhood information is aggregated

$$\mathbf{h}_v^k = \sigma\left(\mathbf{W}_k \sum_{u \in \mathcal{N}(v) \bigcup v} \frac{\mathbf{h}_u^{k-1}}{\sqrt{|\mathcal{N}(u)||\mathcal{N}(v)|}}\right) \tag{10.10}$$

The contribution of the high degree neighbors is lowered in this manner. The GCN model works in a transductive manner for fixed graphs. Many of the earlier approaches optimize the node embeddings using matrix factorization-based objective functions.

An inductive approach called GraphSAGE was introduced by [6]. GraphSAGE samples a fixed size neighborhood of a node and applies the aggregators on those. Unlike the previous transductive approaches, GraphSAGE does not generate embeddings unique to a particular graph. The trained model can be applied to a new similar graph. The embedding for a node v is computed as follows:

$$\mathbf{h}_v^k = \sigma\left(\mathbf{W}_k \cdot CONCAT\left(\mathbf{h}_v^{k-1}, AGG\left(\{\mathbf{h}_u^{k-1}, \forall u \in \mathcal{N}(v)\}\right)\right)\right) \tag{10.11}$$

where AGG is an aggregation function applied on the neighborhood of a node. The aggregated neighborhood is concatenated with the current node's previous layer representation. The three aggregators examined in [6] are as follows.

The *mean aggregator* which takes element-wise mean of the neighborhood representations of a node is given in Eq. 10.12.

$$AGG = MEAN\left(\left\{\mathbf{h}_v^{k-1}\right\} \cup \left\{\mathbf{h}_u^{k-1}, \forall u \in \mathcal{N}(v)\right\}\right) \tag{10.12}$$

The *pooling aggregator* is given in Eq. 10.13, where a neural network is applied to the representations of neighborhood nodes and then element-wise max-pool operation (γ) is applied.

$$AGG = \gamma\left(\left\{\sigma'\left(\mathbf{W}_{pool}\mathbf{h}_u^{k-1} + b\right), \forall u \in \mathcal{N}(v)\right\}\right) \tag{10.13}$$

The third aggregator is *Long Short-Term Memory (LSTM) aggregator*. The LSTM architecture depends on the order of the input, hence the aggregator is applied to a random permutation of the neighbors. It is given in Eq. 10.14.

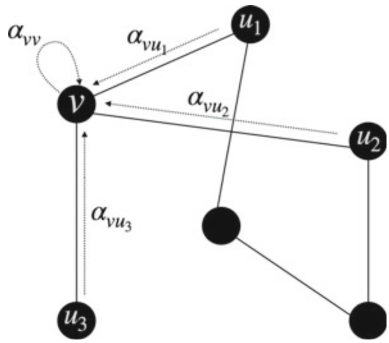


Fig. 10.4 Attention coefficients. Each of the neighbors may not contribute equally to the representation of a node, hence the attention coefficient is calculated in Graph Attention Networks. In this example, the attention coefficients are shown as α_{vu_i} where u_i are neighbors of v

$$AGG = LSTM\left(\left[\mathbf{h}_u^{k-1}, \forall u \in \pi\left(\mathcal{N}(v)\right)\right]\right) \tag{10.14}$$

The Graph Attention Networks [23] introduced attention-based method to generate the node representation by attending over the neighborhood of the node. The attention coefficient of a node and each of its neighbor is computed as given in Eq. 10.15 and a graphical representation is shown in Fig. 10.4.

$$\alpha_{ui} = \frac{exp\left(LeakyReLU\left(\mathbf{a}^T\left[\mathbf{W}\mathbf{h}_u|\mathbf{W}\mathbf{h}_i\right]\right)\right)}{\sum_{j\in\mathcal{N}(v)} exp\left(LeakyReLU\left(\mathbf{a}^T\left[\mathbf{W}\mathbf{h}_v|\mathbf{W}\mathbf{h}_j\right]\right)\right)} \tag{10.15}$$

where we compute importance of node i's features to node v's representation. It is normalized using softmax over the neighborhood. The attention mechanism used here is Multi-Layered Perceptron applied to the concatenation of the linearly transformed representations of the node v and node i. The attention coefficient is further used to take the weighted linear combination of the neighborhood representations of the node in order to form next layer representation for the current node. A non-linear function σ is applied to the linear combination as given by Eq. 10.16.

$$\mathbf{h}_v^k = \sigma\left(\sum_{i\in\mathcal{N}(v)\cup v} \alpha_{vi}\mathbf{W}\mathbf{h}_i^{k-1}\right) \tag{10.16}$$

To stabilize the learning process, *multi-head* attention is used. The attention coefficients are calculated by applying different attention mechanisms in different attention heads. The representations generated using each of the attention mechanisms are either concatenated as in Eq. 10.17 which increases the dimensions of the new representation to $\mathbb{R}^{L\times d'}$. We consider that there are L attention heads.

$$\mathbf{h}_v^k = \Big\Vert_{l=1}^{L} \sigma\left(\sum_{i\in\mathcal{N}(v)\bigcup v} \alpha_{vi}^{k,l}\mathbf{W}_k^l\mathbf{h}_i^{k-1}\right) \tag{10.17}$$

The above equation generates the representation with dimensions L times more. In order to stabilize the learning as well as generate representation of size d', the L representations are averaged as in Eq. 10.18.

$$\mathbf{h}_v^k = \sigma\left(\frac{1}{L}\sum_{l=1}^{L}\sum_{i\in\mathcal{N}(v)\bigcup v}\alpha_{vi}^{k,l}\mathbf{W}_k^l\mathbf{h}_i^{k-1}\right) \tag{10.18}$$

The non-linearity is applied after taking the average of the neighborhood representation from each of the attention heads.

Graph Isomorphism Networks [25] is a framework that generalizes the Weisfeiler-Lehman (WL) [24] isomorphism test. The node feature representation update function is given as follows:

$$\mathbf{h}_v^k = MLP^{(k)}\left(\left(1+\epsilon^{(k)}\right)\cdot\mathbf{h}_v^{k-1} + \sum_{u\in\mathcal{N}(v)}\mathbf{h}_u^{k-1}\right) \tag{10.19}$$

where ϵ is a learnable parameter. The graph neural network is said to be powerful if the aggregation function it uses is injective, that is, if it maps the non-isomorphic graphs to different representations. In the previous models, max-pooling or mean pooling were used to do the neighborhood aggregation. These schemes fail to fully capture the structural dissimilarity between two dissimilar graphs. GIN uses the sum pooling and manages to map nodes with different structure to different node representations.

In some corner cases, GIN fails to produce the different representations for nodes with different rooted subtree structures. These cases are resolved using Relational Pooling [17] architecture.

10.4 Graph Autoencoders

The graph autoencoders (GAE) consists of two parts, encoder and decoder. The encoder produces the latent representations corresponding to the nodes of the graph and the decoder part of a GAE aims at reconstructing the original adjacency matrix of the graph. The GAE is shown in Fig. 10.5.

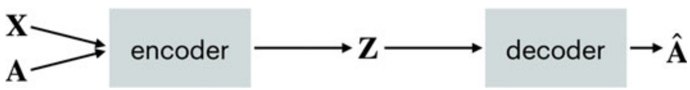


Fig. 10.5 Graph Autoencoder (GAE). The **X** and **A** are the feature matrix and the adjacency matrix of the graph, respectively. **Z** is the matrix of latent representations and $\hat{\mathbf{A}}$ is the reconstructed adjacency matrix

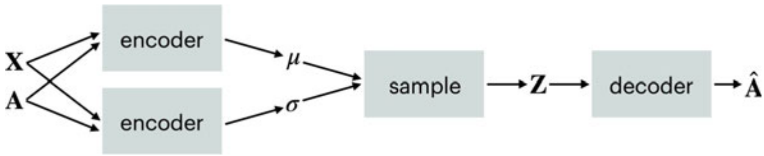


Fig. 10.6 Graph Variational Autoencoder (GAE). The mean μ and variance σ of the input is calculated by two respective encoders in order to know the distribution

Table 10.2 Various methods discussed in the paper

Category	Method	Update
Spectral based	Spectral Graph Convolutional Networks [1]	$\mathbf{H}_j^{k+1} = \sigma(\mathbf{U}\sum_{i=1}^{f_{k-1}} \Theta_{i,j}^k \mathbf{U}^T \mathbf{H}_j^k)$ $(j = 1, \ldots, f_{k-1})$
	ChebNet [3]	$\mathbf{H} = \sum_{k=0}^{K} T_k(\hat{\mathbf{L}})\mathbf{X}\Theta_k$
	GCN [9]	$\mathbf{H} = \tilde{\mathbf{D}}^{\frac{1}{2}}\tilde{A}\tilde{\mathbf{D}}^{\frac{1}{2}}\mathbf{X}\Theta$
Non spectral based	GraphSAGE [6]	$\mathbf{h}_v^k = \sigma\left(\left[\mathbf{W}_k \cdot AGG\left(\{\mathbf{h}_u^{k-1}, \forall u \in \mathcal{N}(v)\}\right), \mathbf{B}_k \mathbf{h}_v^{k-1}\right]\right)$
	Graph Attention Networks [23]	$\mathbf{h}_v^k = \sigma\left(\sum_{i \in \mathcal{N}(v) \cup v} \alpha_{vi} \mathbf{W} \mathbf{h}_i^{k-1}\right)$
	Graph Isomorphism Networks [25]	$\mathbf{h}_v^k = MLP^{(k)}\left(\left(1+\epsilon^{(k)}\right) \cdot \mathbf{h}_v^{k-1} + \sum_{u \in \mathcal{N}(v)} \mathbf{h}_u^{k-1}\right)$
Graph Autoencoders	GAE [10]	$\mathbf{Z} = GCN\,(\mathbf{X}, \mathbf{A})$ where $\hat{\mathbf{A}} = sigmoid\left(\mathbf{Z}\mathbf{Z}^T\right)$
	Linear GAE [20] [21]	$\mathbf{Z} = \tilde{\mathbf{A}}\mathbf{W}$ and $\hat{\mathbf{A}} = sigmoid(\mathbf{Z}\mathbf{Z}^T)$
Spatio-temporal Graph Neural Networks	DCRNN [12]	$r^{(t)} = \sigma\left(\mathbf{W}_r \star_G \left[\mathbf{X}^{(t)}, \mathbf{H}^{(t-1)}\right] + b_r\right)$ $u^{(t)} = \sigma\left(\mathbf{W}_u \star_G \left[\mathbf{X}^{(t)}, \mathbf{H}^{(t-1)}\right] + b_u\right)$ $C^{(t)} = \tanh\left(\mathbf{W}_c \star_G \left[\mathbf{X}^{(t)}, \left(r^{(t)} \odot \mathbf{H}^{(t-1)}\right)\right] + b_c\right)$ $\mathbf{H}^{(t)} = u^{(t)} \odot \mathbf{H}^{(t-1)} + \left(1 - u^{(t)}\right) \odot C^{(t)}$
	STGCN [26]	$\mathbf{H}^{(l+1)} = \Gamma_1^l *_\tau ReLU\left(\Theta^l * \left(\Gamma_0^l *_\tau \mathbf{H}^l\right)\right)$

The variational graph autoencoders learn the latent representations in a probabilistic way. Kipf and Welling [10] introduced Graph variational auto-encoders (VGAE). The conventional working of a GVAE is shown in Fig. 10.6. The VGAE uses two layered GCN [9] as the inference model and the inner product of the representations as the generative model. The inference model or the encoder of VGAE that consists of GCN layers is as follows (Table 10.2):

$$q\,(\mathbf{A}|\mathbf{Z}) = \prod_{i=1}^{N} q\,(\mathbf{z}_i|\mathbf{X}, \mathbf{A})\,, \quad \text{where} \quad q\,(\mathbf{z}_i|\mathbf{X}, \mathbf{A}) = \mathcal{N}\left(\mathbf{z}_i|\mu_i, diag\left(\sigma_i^2\right)\right) \tag{10.20}$$

where μ and σ are calculated using GCNs. The first layer of their GCNs share the weights. The two layers of the GCN work as per Eq. 10.10, which can be written as $GCN(\mathbf{X}, \mathbf{A}) = \tilde{\mathbf{A}}\text{ReLU}(\tilde{\mathbf{A}}\mathbf{X}\mathbf{W}_0)W_1$, where W_0 and W_1 are weight matrices corresponding to first and second layers of the GCN and the matrix $\tilde{\mathbf{A}} = \mathbf{D}^{-\frac{1}{2}}\mathbf{A}\mathbf{D}^{-\frac{1}{2}}$. The generative model or the decoder for the GVAE is given based on the similarities of the latent space variables, which are generated using the 'reparameterization trick'[8]. This is necessary as the back-propagation cannot be done when $q\ (\mathbf{z}_i|\mathbf{X}, \mathbf{A})$ is sampled, since it cannot be differentiated. The generative model or the decoder is as follows:

$$p\ (\mathbf{A}|\mathbf{Z}) = \prod_{i=1}^{N}\prod_{j=1}^{N} p\left(A_{ij}|\mathbf{z}_i, \mathbf{z}_j\right), \quad \text{where} \quad p\left(A_{ij} = 1|\mathbf{z}_i, \mathbf{z}_j\right) = sigmoid\left(\mathbf{z}_i^T \mathbf{z}_j\right) \tag{10.21}$$

The objective function for GVAE is

$$\mathcal{L} = \mathbb{E}_{q(\mathbf{Z}|\mathbf{X},\mathbf{A})}\left[\log p(\mathbf{A}|\mathbf{Z})\right] - \mathrm{KL}[q(\mathbf{Z}|\mathbf{X}, \mathbf{A})||p(\mathbf{Z})] \tag{10.22}$$

where the first term gives the expectation of the likelihood of generating the adjacency matrix A when the latent variables are sampled from the distribution $q(\mathbf{Z}|\mathbf{X}, \mathbf{A})$. The second term in the equation calculates the Kullback–Leibler divergence between the two distributions $q(\mathbf{Z}|\mathbf{X}, \mathbf{A})$ and $p(\mathbf{Z})$.

The non-probabilistic version of the Graph Autoencoders (GAE) consists of only the two-layer GCN as encoder and the inner product of the matrices of latent variables as decoder that reconstructs the adjacency matrix $\hat{\mathbf{A}}$. This is given in the following equations:

$$\hat{\mathbf{A}} = sigmoid\left(\mathbf{Z}\mathbf{Z}^T\right) \quad \text{where} \quad \mathbf{Z} = GCN\ (\mathbf{X}, \mathbf{A}) \tag{10.23}$$

The reconstruction loss in GAE is minimized as follows:

$$\mathcal{L} = \mathbb{E}_{q(\mathbf{Z}|\mathbf{X},\mathbf{A})}\left[log\ p(\hat{\mathbf{A}}|\mathbf{Z})\right] \tag{10.24}$$

The shortcoming of the GVAE is that it primarily preserves only the topological structure of the graph. The Adversarially Regulated Graph Autoencoders (ARGA) [18] encode the contents of the nodes along with the topological structure of graph. This model makes use of adversarial regularization. The latent representations are matched with the prior distribution so that the discriminator can discriminate the latent variable $\mathbf{z}_i$ if it is from the encoder or from the prior distribution. Adversarially Regulated Variational Graph Autoencoders (ARVGA) are also discussed in the paper where it makes use of the VGAE instead of a GAE.

The autoencoding techniques that we have discussed makes use of GCNs as an encoder. Salha et al. [20, 21] recently proposed to use a linear model as an encoder instead of the GCN. This model is called Linear Graph Auto Encoders (LGAE). The encoder and decoder of LGEA is as follows:

$$\mathbf{Z} = \tilde{\mathbf{A}}\mathbf{W} \quad and \quad \hat{\mathbf{A}} = sigmoid(\mathbf{Z}\mathbf{Z}^T) \tag{10.25}$$

where $\mathbf{W}$ is the weight matrix corresponding to the linear encoder. The encoder linearly maps the adjacency matrix to the latent space.

The encoder of the linear graph variational autoencoders (LGVAE) outputs the distribution by giving μ and σ as follows:

$$\mu = \tilde{\mathbf{A}}\mathbf{W}_{\mu} \quad \text{and} \quad \log \sigma = \tilde{\mathbf{A}}\mathbf{W}_{\sigma} \quad \text{then} \quad \mathbf{z}_i \sim \mathcal{N}(\mu_i, diag(\sigma_i^2)) \tag{10.26}$$

Except the encoder, rest of the model is similar to that of the GVAE as discussed above.

10.5 Spatio-Temporal Graph Neural Networks

The spatio-temporal systems have dependency on both time and space. The dependency can be observed in a form of changes in the structure or the features over the period of time. If the systems can be well modeled using a graph, we need to also incorporate the changes that occur in this graph. These types of graphs are dominated by temporal in addition to spatial and feature information as in earlier cases. The neural networks that would be used to find the embeddings of these graph, hence, need to consider the changes in the graph happening over the period of time. We can broadly categorize these neural networks as spatio-temporal graph neural networks (STGNNs).

The typical choice for handling the spatial dependencies is using some neural network that produces the embedding considering the structural and feature information of the graph. A typical example of this could be using a Graph Convolutional Network [9]. Whereas in order to handle the temporal information, a suitable variation of a recurrent neural network may be used. While this is a very broad way of finding embeddings using STGNNs, various other combinations can be made based on the need of the system.

The Graph Convolutional Recurrent Neural Network [22] proposed two architectures. Both the architectures use the LSTM in order to model the temporal dependency. They differ from each other in the method that is used to model the spatial dependency. One of the two uses the Convolutional Neural Networks over graph, while the other one uses the Graph Convolutions.

The Diffusion Convolution Recurrent Neural Network (DCRNN) [13] uses the diffusion convolution to model the spatial dependency while the temporal dependency is modeled using a Gated Recurrent Unit (GRU) [2]. The diffusion is defined as follows (Fig. 10.7):

$$\mathbf{X} \star_G \mathbf{W} = \sum_{k=0}^{K} \left(\mathbf{W}_O \left(\mathbf{D}_O^{-1} \mathbf{A} \right)^k + \mathbf{W}_I \left(\mathbf{D}_I^{-1} \mathbf{A}^T \right)^k \right) \mathbf{X} \tag{10.27}$$

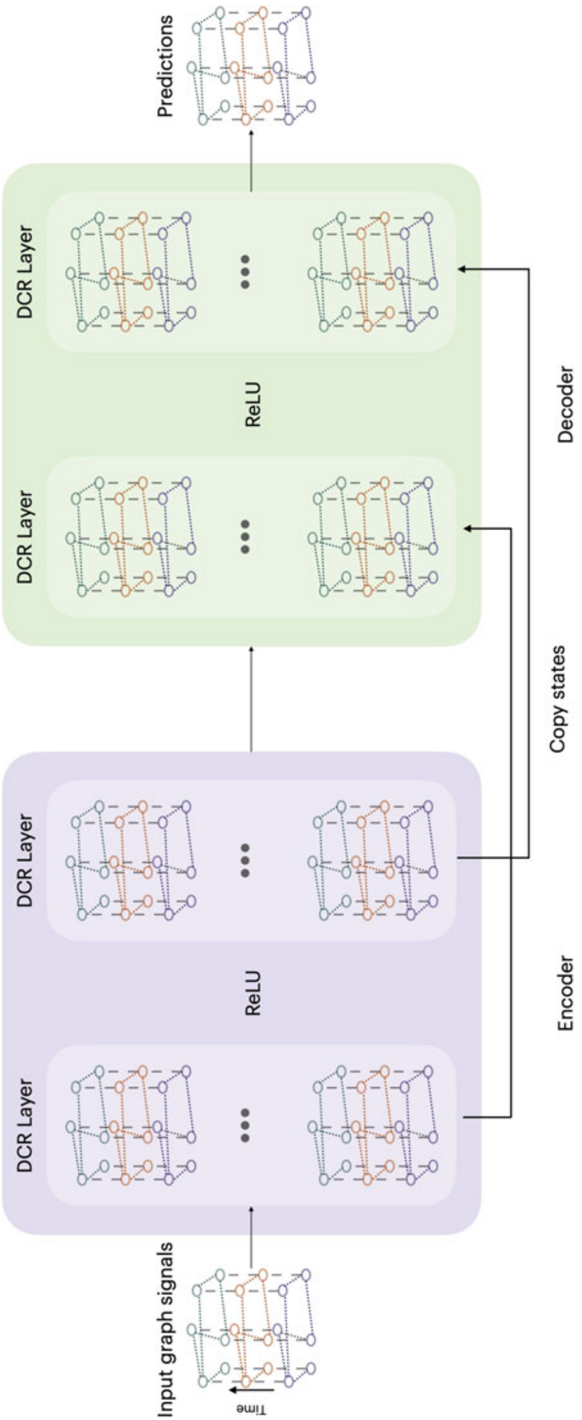


Fig. 10.7 Diffusion convolutional recurrent neural networks [12]

Table 10.3 Diffusion convolution recurrent neural network terminology

$r^{(t)}$	Reset gate vector
$u^{(t)}$	Update gate vector
$C^{(t)}$	Candidate gate vector
$\mathbf{H}^{(t)}$	Output vector
$\mathbf{W}, \mathbf{b}$	Parameter matrix and vector

where $\star_G$ is a diffusion convolution operation over graph G. The diffusion operation is applied K times. $\mathbf{W}_O$ and $\mathbf{W}_I$ are learnable parameters for bidirectional diffusion. $\mathbf{D}_O$ is the out-degree matrix and $\mathbf{D}_I$ is the in-degree matrix.

The diffusion convolution layer is defined as follows:

$$\mathbf{H} = \sigma\left(\mathbf{X} \star_G \mathbf{W}\right) \tag{10.28}$$

where $\mathbf{W} \in \mathbb{R}^{d' \times d}$, $\mathbf{H} \in \mathbb{R}^{n \times d}$ and $\mathbf{X} \in \mathbb{R}^{n \times d'}$. The diffusion convolution can be applied in directed as well as undirected setting. Note that the spectral graph convolution is similar diffusion convolution when applied in undirected graphs.

The temporal dependency is captured using GRU as follows:

$$r^{(t)} = \sigma\left(\mathbf{W}_r \star_G \left[\mathbf{X}^{(t)}, \mathbf{H}^{(t-1)}\right] + b_r\right) \tag{10.29a}$$

$$u^{(t)} = \sigma\left(\mathbf{W}_u \star_G \left[\mathbf{X}^{(t)}, \mathbf{H}^{(t-1)}\right] + b_u\right) \tag{10.29b}$$

$$C^{(t)} = \tanh\left(\mathbf{W}_c \star_G \left[\mathbf{X}^{(t)}, \left(r^{(t)} \odot \mathbf{H}^{(t-1)}\right)\right] + b_c\right) \tag{10.29c}$$

$$\mathbf{H}^{(t)} = u^{(t)} \odot \mathbf{H}^{(t-1)} + \left(1 - u^{(t)}\right) \odot C^{(t)} \tag{10.29d}$$

The terms used in STGNNs are listed in Table 10.3. The encoder–decoder architecture is used in DCRNNs to make the predictions based on the data. The encoder generates a fixed-length embeddings for the data which is then passed to a decoder that makes the predictions.

The Dynamic Diffusion Convolutional Recurrent Neural Network (D-DCRNN) [15] is similar to DCRNN, as it has an encoder–decoder architecture with both of them having the convolutional and the recurrent layers. The adjacency matrix in the D-DCRNN is dynamic in nature which is computed from the current state of the data.

Most of the STGNN architectures have been designed by keeping the traffic networks in mind. The recurrent networks are slow with respect to aspects such as complex gate mechanisms and dynamic changes. In order to overcome these, the architecture called Spatio-temporal Graph Convolutional Networks (STGCN), proposed in [26], makes use of convolutions. The STGCN architecture as shown in Fig. 10.8 basically has two blocks of spatio-temporal convolutions followed by the output layer. Each of the spatio-temporal block sandwiches a spatial convolutional block between to temporal gated convolutional blocks as shown in Eq. 10.30. Unlike

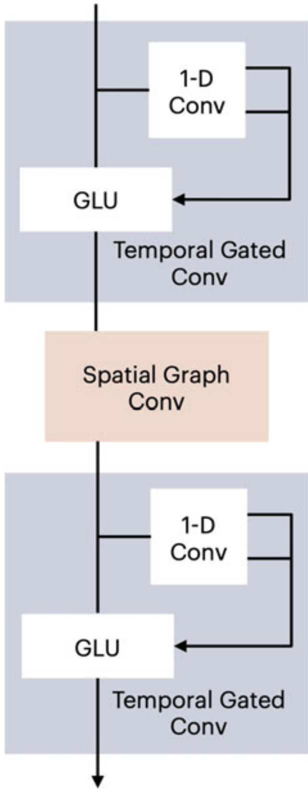


Fig. 10.8 The spatio-temporal block in STGCN [26] that sandwiches a spatial convolutional block between two temporal gated convolutional blocks

the previously discussed architectures, STGCN does not have the encoder–decoder architecture, but it still incorporates the bottleneck strategy by making use of 64 filters for temporal gated convolutional blocks and 16 filters for spatial graph convolution block. In each of the temporal gated convolutional block, residual connection is used. Gated Linear Unit (GLU) acts as an activation in these blocks.

$$\mathbf{H}^{(l+1)} = \Gamma_1^l *_\tau ReLU\left(\Theta^l * \left(\Gamma_0^l *_\tau \mathbf{H}^l\right)\right) \tag{10.30}$$

where Γ_0^l and Γ_1^l are the upper and lower temporal kernels in the spatio-temporal block l, Θ is the spectral graph convolution kernel and $ReLU$ is the rectified linear unit.

Table 10.4 Summary of features exhibited by the methods discussed in this paper

Method	Spectral	Spatial	Temporal	Transductive	Inductive
Spectral graph convolutional networks [1]	●	○	○	●	○
ChebNet [3]	●	○	○	●	○
GCN [9]	●	○	○	●	○
GraphSAGE [6]	○	●	○	○	●
Graph attention networks [23]	○	●	○	○	●
Graph isomorphism networks [25]	○	●	○	○	●
GAE [10]	○	●	○	○	●
Linear GAE [9, 20, 21]	○	●	○	○	●
DCRNN [12]	○	●	●	○	●
STGCN [26]	○	●	●	○	●

10.6 Discussion

While the previous approaches have been transductive in nature, the GraphSAGE was the approach introduced as inductive one. The inductive approaches can generate the embeddings for unseen networks using the model trained on similar graphs. Most of the approaches before GraphSAGE work in transductive setting, that is, they cannot generalize to similar new data. Though they can be modified to work for new similar graphs, the modification is computationally expensive. Graph attention networks do work in an inductive manner. The results shown by GAT are better when compared with GraphSAGE. The weightage in the form of attention coefficient corresponding to each pair of the neighbors determines the contribution of the neighbor in the embedding. The graph structures that cannot be distinguished by the GNNs like GCN, GraphSAGE, are indentified in GIN paper. The GIN architecture is as powerful as WL-isomorphism test in distinguishing the graph structures (Table 10.4).

During the training of DCRNNs, the final states of encoder are passed on to the decoder which then generates the predictions given the ground truth. During the testing time, the predictions replace the ground truth, which leads to variation in the distribution. In order to overcome this, the actual ground truth and predictions are given to the decoder with some probability ϵ_i and $1 - \epsilon_i$, respectively. While these methods rely on the recurrent neural networks, STGCN does not employ it since the RNN iterations are slower on the traffic data for which the model was essentially designed. To overcome all the shortcomings of using recurrent networks, the STGCN implements the convolutional networks across the temporal component. Though the models are especially designed for traffic forecasting, they can be extended for other systems that can be modeled as spatio-temporal graphs.

10.7 Future Directions

The existing graph representation learning techniques have their own strengths about producing representations. However, considering the scope we suggest the following directions for future work. Although there has been some work [7, 14] in these directions, there is still a lot of significant work to be done.

- First, these techniques work primarily for static graphs, whereas many real-life systems modeled as graphs are dynamic in nature. Considering the dynamicity of the graphs, it is important to learn the representations of the dynamic graphs.
- Second, multi-layered graphs' representations may need a different approach than the existing ones.
- Thirdly, when there is diversity in the characteristics of the nodes in the graph, that is, the nodes exhibit heterogeneous behavior, the representations need to capture the variations in the working of the nodes.

10.8 Conclusion

In this paper, we have reviewed popular graph representation learning methods. These have been broadly categorized into spectral based methods and non-spectral based methods. The spectral based methods consider the spectrum of the graph in order to perform operations such as convolutions on it. The non-spectral based methods makes use of the information from neighborhood of nodes. Each of the methods may have their own way of defining how the neighborhood information is aggregated. We also discussed graph autoencoders, which learn the latent representations for the graphs based on which they reconstruct the graph adjacency matrix back. The variational autoencoders are also discussed, which are probabilistic models, which estimates the distribution of the data, from which the latent representation is chosen and the adjacency matrix is reconstructed. These autoencoders are helpful in the tasks such as link predictions, where they predict the probability of any two nodes having an edge connecting them. We have also discussed some of the spatio-temporal graph neural networks that are used for the data that is modeled as spatio-temporal graphs. The STGNNs use neural networks to include the spatial as well as the temporal dependencies in the graph (Fig. 10.9).

We further gave the future direction of research in this area. Real-world examples that are modeled as dynamic graphs can to be addressed using graph neural networks, which is an open area for research. Graph Neural Networks for multi-layered graphs

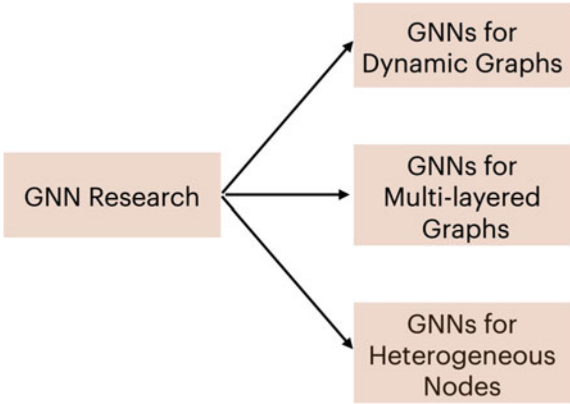


Fig. 10.9 Future research directions

and the graphs where different nodes have different physics behind their working are types of graphs other than static graphs, which need to be addressed using graph neural networks.

References

1. Bruna, J., Zaremba, W., Szlam, A., LeCun, Y.: Spectral networks and locally connected networks on graphs. arXiv:1312.6203 (2013)
2. Cho, K., Van Merriënboer, B., Gulcehre, C., Bahdanau, D., Bougares, F., Schwenk, H., Bengio, Y.: Learning phrase representations using RNN encoder-decoder for statistical machine translation. arXiv:1406.1078 (2014)
3. Defferrard, M., Bresson, X., Vandergheynst, P.: Convolutional neural networks on graphs with fast localized spectral filtering. In: Advances in Neural Information Processing Systems, pp. 3844–3852 (2016)
4. Gori, M., Monfardini, G., Scarselli, F.: A new model for learning in graph domains. In: Proceedings. 2005 IEEE International Joint Conference on Neural Networks, vol. 2, pp. 729–734 (2005)
5. Grover, A., Leskovec, J.: Node2Vec: scalable feature learning for networks. In: Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, pp. 855–864 (2016)
6. Hamilton, W., Ying, Z., Leskovec, J.: Inductive representation learning on large graphs. In: Guyon, I., Luxburg, U.V., Bengio, S., Wallach, H., Fergus, R., Vishwanathan, S., Garnett, R. (eds.) Advances in Neural Information Processing Systems, vol. 30, pp. 1024–1034. Curran Associates, Inc. (2017). http://papers.nips.cc/paper/6703-inductive-representation-learning-on-large-graphs.pdf
7. Iiyama, Y., Cerminara, G., Gupta, A., Kieseler, J., Loncar, V., Pierini, M., Qasim, S.R., Rieger, M., Summers, S., Onsem, G.V., Wozniak, K., Ngadiuba, J., Guglielmo, G.D., Duarte, J., Harris, P., Rankin, D., Jindariani, S., Liu, M., Pedro, K., Tran, N., Kreinar, E., Wu, Z.: Distance-weighted graph neural networks on FPGAS for real-time particle reconstruction in high energy physics. arXiv:abs/2008.03601 (2020)
8. Kingma, D.P., Welling, M.: Auto-encoding variational bayes. arXiv:1312.6114 (2013)
9. Kipf, T.N., Welling, M.: Semi-supervised classification with graph convolutional networks. arXiv:1609.02907 (2016)

10. Kipf, T.N., Welling, M.: Variational graph auto-encoders. In: NIPS Workshop on Bayesian Deep Learning (2016)
11. Levie, R., Monti, F., Bresson, X., Bronstein, M.M.: Cayleynets: graph convolutional neural networks with complex rational spectral filters. IEEE Trans. Signal Process. **67**(1), 97–109 (2018)
12. Li, Y., Yu, R., Shahabi, C., Liu, Y.: Diffusion convolutional recurrent neural network: data-driven traffic forecasting. arXiv:1707.01926 (2017)
13. Li, Y., Yu, R., Shahabi, C., Liu, Y.: Diffusion convolutional recurrent neural network: data-driven traffic forecasting. In: International Conference on Learning Representations (2018)
14. Luo, W., Zhang, H., Yang, X., Bo, L., Yang, X., Li, Z., Qie, X., Ye, J.: Dynamic heterogeneous graph neural network for real-time event prediction. In: Proceedings of the 26th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining. p. 3213-3223. KDD '20, Association for Computing Machinery, New York, NY, USA (2020). https://doi.org/10.1145/3394486.3403373
15. Mallick, T., Kiran, M., Mohammed, B., Balaprakash, P.: Dynamic graph neural network for traffic forecasting in wide area networks. arXiv:2008.12767 (2020)
16. Micheli, A.: Neural network for graphs: a contextual constructive approach. IEEE Trans. Neural Netw. **20**(3), 498–511 (2009)
17. Murphy, R.L., Srinivasan, B., Rao, V., Ribeiro, B.: Relational pooling for graph representations. arXiv:1903.02541 (2019)
18. Pan, S., Hu, R., Long, G., Jiang, J., Yao, L., Zhang, C.: Adversarially regularized graph autoencoder for graph embedding. In: IJCAI (2018)
19. Perozzi, B., Al-Rfou, R., Skiena, S.: Deepwalk: online learning of social representations. In: Proceedings of the 20th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, pp. 701–710 (2014)
20. Salha, G., Hennequin, R., Vazirgiannis, M.: Keep it simple: graph autoencoders without graph convolutional networks. In: Workshop on Graph Representation Learning 33rd Conference on Neural Information Processing Systems (NeurIPS) (2019)
21. Salha, G., Hennequin, R., Vazirgiannis, M.: Simple and effective graph autoencoders with one-hop linear models. In: European Conference on Machine Learning and Principles and Practice of Knowledge Discovery in Databases (ECML-PKDD) (2020)
22. Seo, Y., Defferrard, M., Vandergheynst, P., Bresson, X.: Structured sequence modeling with graph convolutional recurrent networks. In: International Conference on Neural Information Processing, pp. 362–373. Springer (2018)
23. Veličković, P., Cucurull, G., Casanova, A., Romero, A., Liò, P., Bengio, Y.: Graph attention networks. In: International Conference on Learning Representations (2018). https://openreview.net/forum?id=rJXMpikCZ
24. Weisfeiler, B., Lehman, A.A.: A reduction of a graph to a canonical form and an algebra arising during this reduction. Nauchno-Tech. Informatsia **2**(9), 12–16 (1968)
25. Xu, K., Hu, W., Leskovec, J., Jegelka, S.: How powerful are graph neural networks? In: International Conference on Learning Representations (2019). https://openreview.net/forum?id=ryGs6iA5Km
26. Yu, B., Yin, H., Zhu, Z.: Spatio-temporal graph convolutional networks: A deep learning framework for traffic forecasting. In: Proceedings of the 27th International Joint Conference on Artificial Intelligence (IJCAI) (2018)
27. Zhuang, C., Ma, Q.: Dual graph convolutional networks for graph-based semi-supervised classification. In: Proceedings of the 2018 World Wide Web Conference, pp. 499–508 (2018)

Chapter 11
Social Media Data Collection and Quality for Urban Studies

Álvaro Bernabeu-Bautista, Leticia Serrano-Estrada, and Pablo Martí

Abstract As of today, many studies have demonstrated the possibilities that geolocated data from social networks have for the study of urban phenomena. This chapter offers a retrospective and panoramic view of a selection of social networks that have been used to understand a wide range of urban dynamics. Findings from this review and previous experiences evidence that the social networks often selected and used for the purpose of assessing city dynamics share key characteristics such as (i) the locative properties; (ii) the data privacy and availability; (iii) the data potentiality to inform about specific phenomena related to the urban environment; and, (iv) the fact that they are mobile device-based platforms and, thus, users are considered as sensors, and their traces as crowd-sourced sensory information. Five exemplary social networks that meet these four conditions are dealt with in detail (Google Places, Foursquare, Twitter, Instagram and Airbnb), highlighting the opportunities and challenges they portray with respect to data collection and quality for the purpose of urban studies.

Á. Bernabeu-Bautista · L. Serrano-Estrada · P. Martí (✉)
Building Sciences and Urbanism Department, University of Alicante, Carretera San Vicente del Raspeig s/n, 03690 San Vicente del Raspeig, Alicante, Spain
e-mail: pablo.marti@ua.es

Á. Bernabeu-Bautista
e-mail: alvaro.bautista@ua.es

L. Serrano-Estrada
e-mail: leticia.serrano@ua.es

A. Biswas et al. (eds.), *Principles of Social Networking*,
Smart Innovation, Systems and Technologies 246,
https://doi.org/10.1007/978-981-16-3398-0_11

11.1 Introduction

11.1.1 Relevance of Social Networks

The use of social web services and, more specifically, geolocated social networks, is increasingly booming and extending to different age groups that over time have become more familiar and adapted to new technologies. During the period between 2000 and 2020, Internet use has increased significantly, registering a growth of 593% in Europe and 1266% globally and reaching a penetration rate of 63.2% in the world [1]. Similarly, the use of social networks and online services by adult Internet users has experienced an exponential increase in recent years. In 2020, from the 58% of the global population considered as Internet users, 83% have used social networks, which represent more than 3.8 billion users in the world and denote stability in the penetration of these platforms [2].

Traditionally, the study and analysis of cities from historical and morphological approaches have always been closely linked to both bibliographical and field work. However, following the above-described raising tendency in the use of new information technologies, these have become a source of data useful for understanding urban phenomena. Indeed, since social networks and other web services have transformed the way we communicate, learn, acquire information, develop skills, improve productivity at work and access to leisure [3] through their user-friendly, agile and public platforms, urban researchers have worked for over a decade in theorizing and developing methods that adopt these as volunteered generated information (VGI) [4]. Specifically, previous research has assessed a wide range of urban phenomena, from very different approaches and scales, to understand how the city's inhabitants behave [5], where they go [6], at what times [7] or what kind of activities they perform [8], among others.

With social networks, users no longer act as receivers of information but have also become producers of large amounts of data, often linked to a specific geographic location, as many of these platforms have locative characteristics. Thus, on each of these platforms, users leave a geolocated "fingerprint" with geographic coordinates when they post or share a message [9]. When accessible, these types of social networks, also known as LBSNs or location-based social networks, include large databases with a wide range of specific variables like text, photographs, quantitative data, etc. As such, a wide range of research questions related to human activity in the city have been answered through the representation and interpretation of data obtained from these sources. Despite the fact that social networks are not used by the entire population, they can constitute a representative sample of citizens' preferences, opinions and urban activities [10, 11].

Research developed in these fields has addressed very diverse issues in different case study cities, such as the identification of areas of interest [12, 13]; the perception of public spaces [14, 15]; the pinpointing of spaces that may be susceptible for intervention [16], the analysis of tourist consumer markets [17]; or the adequate planning of mobility infrastructures [18, 19]. These exemplary research works present various methods and approaches to the use of geolocated data from social networks and web services for the study and interpretation of people's behavior in the city, not only on an individual basis but also with respect to other people.

In view of all the above, this research aims to provide an overview of potential future directions in the field of LBSN research for urban studies. Specifically, it offers a retrospective and panoramic view of a selection of social networks that have recently been used to understand a wide range of urban dynamics.

11.1.2 Social Networks and Their Data

According to the British anthropologists Alfred Radcliffe-Brown and John Barnes, a social network is a structure formed by individuals or groups connected to each other by some kind of relationship or common interest [20, 21]. Currently, online social networks function as social structures formed by a group of users who share common interests, relationships and/or activities, where social meetings take place and information on consumption preferences are revealed [22]. In short, social networks are digital platforms of global communication that connect a vast number of users [23].

All social networks, therefore, share common characteristics, for instance, they create a network of contacts, individuals, or groups, which are registered using a profile that identifies them and allows them to participate in the platform, creating, sharing, or interacting with other users through the content of the network itself [24]. Despite these common basic characteristics, the content of each social network does not present a unique pattern, thus the information included can acquire multiple formats depending on the characteristics or specific functionality of the platform (Fig. 11.1). The analysis of these characteristics within the current panorama of social networks and web services allows to establish a classification of five types based on their use and functionality:

- Social networks focus on contacts, which allow the establishment of connections between users with common links, such as friendships or interests, offering them to interact with the content they share. These social networks range from more generic platforms such as Facebook to more specialized networks such as LinkedIn.
- Microblogging social networks based on the creation of entries or short text messages that can be accompanied by multimedia content such as images or videos and that also allow interaction between users through direct messages, subscriptions, or themes such as Twitter or Tumblr.
- Social networks based strictly on multimedia sharing, accessible by other users who can interact with it through comments or reactions. This content includes images like Instagram, Flickr or Pinterest, videos like Youtube or music like Spotify.
- Specialized social networks offer users a service or functionality different from other types, even though they may include some similar features. Some examples

Fig. 11.1 Comparison matrix between different social networks and web services, highlighting their main data type, privacy and main-use device. *Source* Authors.

are Airbnb (users publicizing or looking for temporary accommodation), Google Places platform (registration, comments, and rating of urban and economic activities) or Foursquare and Sina Weibo (check-in or registration of users' presence at a given place, allowing the sharing of user-generated content such as pictures of the place, opinions, etc.).

- Instant messaging services allow sending textual or other type of content, i.e., videos, pictures, documents, to other users through mobile devices, such as Telegram or WhatsApp.

Based on the use and popularity of different social networks, it is possible to confirm the relevance of Facebook as a social network of contacts, WhatsApp as an instant messaging service and Youtube as a social network of multimedia content, compared to other networks with similar characteristics such as Facebook Messenger or Vimeo. Similarly, the most used social network for sharing images is Instagram as opposed to Pinterest or Flickr, while the preferred one for sharing opinions or creating entries or short messages is Twitter [2]. It must be stated that Instagram and Twitter, in addition to messaging services such as WhatsApp, are more widely used on mobile devices than on any other devices such as computers or tablets, which means that their use can be more intense outside the private context of home or work.

Despite their relevance and popularity, according to the authors' experience, not all the above-mentioned social networks can be used for the study of urban phenomena, however, there are four key characteristics that can pinpoint those that indeed have a potential for this purpose.

- The geolocation of their data, or at least, that the data provided have the necessary information to be able to infer the exact location from which they were generated or to associate the data to a specific place or area. The location of these data allows conclusions to be drawn about the specific characteristics, distribution and/or concentration of the "footprints" left by users.
- The privacy and accessibility of the data, that is, if the data provided by the social network are open and/or can be easily collected.
- The potential of the data variables to offer information that reflects urban and/or social activity.
- The predominant use of social networks in mobile device-based platforms, which allows social network users to share information about their whereabouts from anywhere.

Once these four characteristics are met, the decision as to which social network variables are useful for urban studies largely depend on (i) the aimed approach or the set research objectives and (ii) the information format such as text, images, opinions, activities, quantitative data or temporal data, among others.

11.2 Social Networks for Urban Studies

From the comparison between the social networks and web services revised in the previous section (Fig. 11.1), five exemplary social networks that fulfill the four above-mentioned criteria are Twitter, Instagram, Foursquare, Google Places and Airbnb. Evidently, there are other social networks that have the same or similar functionalities and are used in specific geographical contexts. For instance, Sina Weibo [25, 26], mostly used in China, is check-in-based and thus has a similar functionality as Foursquare.

The selected social networks have accessible geolocated data that can be retrieved through their API service. Moreover, they are mostly used through mobile devices and thus the users' "footprints" reflect both, human and urban activities. Specifically, the particular characteristics and potential use of these social networks are as follows [10]:

- Foursquare includes users' opinions, images and quantitative data that reflect people's preferences about urban activities and spaces.
- Google Places provides information about urban and economic activities, as well as the users' evaluation about these in the form of a rating value and/or comments.
- Twitter includes text from which hot topics, themes or opinions about urban spaces can be depicted, as well as temporary information that refers to the moment in which a certain activity is registered by the user.
- Instagram contains images that often reflect users' preferred city areas and/or their perception of the space. For instance, photographed kids playing in an urban public space could suggest that the place is perceived as safe from traffic.
- Airbnb provides information about temporary non-regulated accommodation.

Each of these social networks is further explained below through their characteristics, their functionality and their data. Moreover, previous studies will be briefly discussed as they provide evidence about the reliability and validity of these sources for assessing different topics in the field of urban studies.

11.2.1 Foursquare

Foursquare is a social network with a mobile device-based application (Swarm) that allows users to register their presence in venues they visit through a check-in. These venues are previously registered by a social network user and associated to a specific predefined category and sub-category that describes the type of activity that takes place in the venue. Users can also rate the venues, share photos about their experience in the venue, add comments or "tips" and/or share these venues with other users. The fundamental concepts behind the functionality of Foursquare are that it is a mobile, local and social network [27]. The *mobile* concept relates to the fact that, as it is installed in mobile devices, it takes part in the citizens' daily lives allowing them to keep a record of their activity and experience in a venue, whether it is a city landmark, a restaurant or an event. The *local* concept is based on the fact that users themselves participate in the construction of Foursquare's database, often with information about their local and nearby environment. Finally, the *social* concept refers to the platform's interactive nature, not only among the registered users but also between the users and the platform itself. The latter consists of a system of achievements that rewards users based on the frequency they check-in or comment on a place, influencing their visibility and recognition within the social network's "society". Although this system is not currently as relevant as it used to be a few years back, the social aspect has been reinforced by the possibility to share images and/or comments. Moreover, the platform's constant location tracking keeps a record of users passing by a venue, sending "passive notifications" in the mobile device about personalized recommendations. Indeed, Foursquare comprises a database that stores users' preferences in the form of number of check-ins and passersby users, the photos they share and their opinions.

The distinction that the application makes between the cumulative number of check-ins and the number of visits or passerby users is an important aspect to consider for the study of urban phenomena. When venues are ranked by these values, two different things can be analyzed. First, the better-ranked venues in terms of check-ins represent the number of times one or several users have purposely registered their presence in the space, whereas the best-ranked venues in terms of visits give an indication of the most popular places among unique users, similar to the values obtained by traditional people counting methods. In addition, geolocated photographs and opinions or tips that are associated with the venues are valuable for analyzing the perception that users have of those spaces and which are the most relevant qualities of their experience while visiting the venues. From the point of view of the urban image, both tips and photographs also allow to physically characterize a space, as well as to recognize the most representative elements. Moreover, from the shared pictures, it is often possible to infer the types of user profile information that visit the space, as well as the types of activity they perform while visiting the venues.

Since Foursquare listings of venues consist of establishments and urban spaces that have purposely been registered and that have at least been checked in once, the retrieved data are rather useful for assessing the demand for economic activities and for identifying activity areas in cities. Indeed, through the analysis of the most checked-in venues and their location patterns, it is possible to characterize urban areas based on their activity specialization. In this regard, another aspect worth highlighting from this social network is that venues are classified into 10 predefined categories and five-level subcategories that provide detailed information about the type of activity that takes place in a venue. This information together with other variables such as the venue rating score, venue address (street, number, neighborhood, district, city and country); geographic coordinates (longitude and latitude); average score (rating); opinions provided by users (tips); photographs shared by users; and, the venue's identifier code, are useful for inferring the collective preferences and opinions.

Previous work that has used geolocated data from Foursquare for the study of urban phenomena has covered a wide range of research topics such as the spatiotemporal patterns of users' mobility between urban spaces in the city [28, 29]; the identification of significant touristic areas [30]; the study of urban activity patterns, clustering and hubs [31–33]; the identification and location of socially relevant public spaces [34], among others. Interesting findings derived from these studies conducted in rather different geographic locations are worth highlighting as they can shed light over other case studies. For instance, the study about the spatial and temporal activity patterns in London evidenced the strong relationship between the location of hot spots in the city, the tourist movement patterns and the key transportation hubs and other functional places, suggesting that this relationship largely depends on the daily rhythm of the areas where these venues are located [28]. Moreover, in a study conducted in 10 cities, it was observed that the spatial relationship between venues defines clusters of activity, either because they are in proximity or because they are well connected by a transportation network, indicating that users tend to move between locations in a limited spatial distance, thus creating sub-spaces within the city [29]. The research about the tourist-functional relations in Barcelona showed that the spatial distribution of tourist activities is highly concentrated in both, the central area of the city, where most key landmarks are located, and the airport. Furthermore, the spatial relation among tourist activities is mostly defined by the location of city attractions where these users tend to make more check-ins [30]. A different work that took place in Seoul demonstrated that some of the Foursquare categories have a strong correlation with population flows and it is possible to recognize a network of activity hubs between them [31]. A research that took place in Finland adopts a new categorization of venues, following Jan Gehl's optional and necessary activities that allowed a thorough understanding of activity patterns in urban spaces [32]. The spatial analysis of venue categories was also useful to find correlations between clusters of activities related to arts and entertainment in Istambul and high densities of food or beverage venues [33]. Finally, the collective preferences for open public spaces registered within the Outdoors and Recreation Foursquare category have been used to identify urban elements, activity hotspots and green areas that have a strong potential to be part of the green infrastructure network in the city of Valencia [34].

11.2.2 Google Places

Google Places is a social network whose objective is to gather, organize and make publicly accessible information about any place in the world [35]. As in the case of Foursquare, Google Places includes places of interest such as outdoor spaces, transportation and infrastructure; however, this service is especially linked to businesses and other economic activities. It gathers information about a given space or activity and is visualized into Google's geospatial web tools, for example, Google Maps or Google Street view, along with other details such as: address, phone number, website, opening and closing hours, etc., allowing any user to attach photos, videos or writing reviews about the place. In other words, Google Maps represents the base map, and Google Places is the tool for geolocating and geotagging elements or activities onto the map [27]. In recent years, the use of Google Places has increased due to the advantages it offers to new businesses [36], specially to those related to services, shopping, hotels and restaurants. They usually appear first in the search options due to the geolocation feature that tracks the user, displaying those places nearby, which are not always the most relevant, so it also works as a great promotional tool for self-employed and/or entrepreneurs, increasing the potential customers.

The datasets collected from Google Places include two different location attributes: a pair of geographic coordinates (longitude and latitude) and the address (street, number, neighborhood, district, city and country). In addition, the places registered are classified into tags that further define the type and specialization of a place or activity. Other variables included in the social network datasets are the name of the place, the ID or unique identifier number, the place's average rating, the comments and photographs shared either by users who visit the place, by the owner of the establishment or by the user who registered the place in the platform.

Due to the above-mentioned characteristics, Google Places is useful for urban studies for assessing the location and types of urban and economic activities in cities. Since the listed places are mostly economic activities, and these are both geolocated and classified into tags, the data from this social network can shed light over different urban phenomena. For instance, the analysis of the density and diversity of economic activity [37, 38], which is rather useful for identifying areas with greater economic activity and their specialization, providing an opportunity to promote specific strategies for the local economic planning of the cities.

Other types of analysis conducted with this source include: the identification of zoning areas according to the types of activity in the city [39, 40]; the detection of places of interest as well as the collective opinion about them through data from Google Places and other web services [41]; the study of the mixture of uses in urban areas for balancing the provision of businesses in particular neighborhoods and to improve walkability, promoting the development of clusters of activities [38]; the analysis of urban activity patterns by estimating the timeframes in which different land uses are active based on their hours of operation [39]; the comparison between the distribution of activities and the administrative divisions of the city [40]; and, in a different approach, Google Places' activity tags and reviews can be complemented with Open Street Maps data to extract the leisure activity potential and distribution in a city [41].

11.2.3 Twitter

Twitter is a social network based on microblogging services, that is, services for sharing short messages, or tweets in this case, with a maximum length of 280 characters. Optionally, the platform allows attaching static or dynamic images, surveys, short videos to the tweets, as well as the specific location from which the message is being sent. When a user is registered on Twitter, he or she can subscribe or "follow" other users to receive notifications once they share a new post. Indeed, a user can interact with the posted tweets through four pre-established options: (i) publish a comment in response to the tweet, (ii) express interest on the tweet through the "Like" option, (iii) forward or "retweet" the post as a direct message to other users, or (iv) share the post publicly on his own profile feed so that the original message can be seen by his followers. Tweets can also include hashtags, which are textual elements or metadata (word or phrase) preceded by a hash or pound character ("#") that refer to relevant concepts defined by the user sending the tweet. A hashtag is a label that is applied to all tweets that relate to the same concept, thus the tweets in a dataset can be categorized by these labels. If a hashtag is constantly shared among many users, it becomes a trending topic within the network.

Another way of interacting with the Twitter community is through the "mentions", which are formed by the @ character followed by a Twitter username. Therefore, Twitter is a social network that allows users to share and exchange interests, experiences, and opinions with other users [42].

Twitter is often used for research in urban studies as it offers the possibility to conduct different types of analysis. The data collected from this social network include a wide range of variables such as: the geolocation of the message if the user has agreed to share the exact location; the date and time when the message was sent; the textual content of tweets; the attached photos and videos, if that is the case; the language of the tweets; the cumulative number of retweets for every tweet; the amount of tweets per user, among others. These variables provide clues about citizen flows and seasonal variations in people concentration patterns as well as other dynamics, such as the influx of people to certain events (e.g. cultural or other kind of festivities or protests). The content of the messages is often indicative of the opinion and/or experience that users have about spaces or activities that take place in the city. These can be recognized through the location labels or hashtags, which allow characterization of these places and activities according to the collective opinion and perception. Furthermore, citizen habits, routines and social activities can be often inferred from the tweet texts and location. Both, the patterns of spatial and temporal presence in certain urban spaces and the collective opinion about them allow identifying those issues or elements that are relevant to citizens and can inform decision making for future strategies or interventions.

Previous research has explored the relevance of geolocated and non-geolocated data from Twitter for the study of urban issues such as the characterization of the urban environment from users' opinion and feelings [43–45], the identification of urban dynamics and mobility patterns at various scales [46, 47] or the monitoring of people flow in urban areas or events [48–50]. Specifically, sentiment analysis and topic extraction via Twitter shared texts can be used to analyze large sports events, like in the case of London Olympic Games 2012, where hourly spatiotemporal patterns and positive and negative sentiments from residents and visitors were depicted [43]; and, to explore to what extent urban parks contribute to general well-being of residents in cities [44] or the quality of life in different neighborhoods [45]. In a different line of research, the aggregated location of tweets was useful for land use identification and to gather insights about population commuting flows between different municipalities of Madrid [46]. Also, in the American context, the tweet patterns have shown that residents of poor minority neighborhoods do travel about as widely across their cities and to as many neighborhoods as those of other more privileged neighborhoods [47]. Another research in Singapore explored the effectiveness of using tweets for crowd flow prediction [48]. In the case of Valencia, in Spain, tweets were used to analyze people concentration at specific events like sports events, festivities or special retail days [49] while in Melbourne, tweets where an alternative to pedestrian sensors while counting the number of people that walks by specific areas in the city center [50].

11.2.4 Airbnb

Airbnb is a short-term property rental service and a social network that aims at "creating a world where everyone can belong everywhere, through local, inclusive, authentic, diverse and sustainable tourism" [51]. The properties for rent vary in size and characteristics. The lodging offer ranges from a sofa bed in a living room to an entire island [52]; however, with a very few exceptions, Airbnb's database mainly includes apartments, complete houses (57%) and/or private rooms (41%) [53]. A registered user can either register a property for rent or search for a property according to certain criteria related to the destination of interest; the dates of stay; the size of the accommodation; the maximum number of guests; etc. The property search can be done either from the website itself or from the mobile application and returns a series of results that can be further filtered according to the accommodation price, the neighborhood where it is located and/or the lodging amenities. The filtered results allow users to access the lodgings' complete description, photos shared by the owner and/or opinions from previous guests; directly contact the user who has registered the accommodation through direct message for requesting more information; and book and pay for the stay. Airbnb's fee depends on the listing price (before taxes) and is charged from both the person who rents it (6–12%) and the accommodation owner (3%).

All Airbnb accommodations are geolocated, (with two pair of coordinates associated) and include other locative information such as the country and city where they are located. Moreover, the datasets collected from Airbnb include other types of variables such as the property's date of registration; the average rating according to the guests' evaluation; the type of accommodation and the type of property, depending on whether it is fully or partially available; the unique accommodation identifier code (property ID); and the unique user identifier code (host ID).

With the above characteristics, the geolocated data from Airbnb offer the possibility to study trends on the temporary non-regulated accommodation on offer and on demand, as well as the distribution of this type of activity throughout the city; the proximity to other relevant areas or spaces; the availability of nearby services; the types of accommodation and their relation to the morphological characteristics of city areas; the economic aspects related to the rental price and/or management that can range from a single accommodation per owner to multiple ownership; etc. Assessing these and other related phenomena would contribute to establish a fair level playing field for other economic activities, based on the trends of lodging and their impact on the current city life.

The relatively recent accommodation dynamics reflected in the patterns of Airbnb data have been explored by urban researchers who have dealt with various topics such as the penetration of this type of temporary accommodation in cities [54, 55]; the comparison between the regulated hotel offer and the non-regulated Airbnb accommodation offer [56]; and the parameters that influence the price of non-regulated accommodation like location or attractiveness of the area [57–59], among others.

The study about Airbnb penetration in U.S. cities showed that, despite the differences in the ethnic composition of the population and the socio-economic characteristics, central areas with a strong presence of educated population are those with the highest Airbnb penetration [54], while in the European context, Airbnb is highly concentrated in dense areas where hotels are traditionally located, while also complementing the hotel industry in suburban areas [55]. A different study of how Airbnb impacts the hotel offer showed that Airbnb mildly "cannibalizes" hotel sales and expands the market for the industry, although it directly affects the vulnerability of high-end hotels to lower Airbnb hosting costs, especially with the recent attempt of Airbnb accommodations to behave more like the traditional fixed-capacity hotel offer [56]. A different research approach to the analysis of Airbnb properties on offer focuses on the number of reviews. Specifically, interesting findings in this area show that the reviews are mostly concentrated in those parts of the city that have a younger population, a significant number of housing units, and a high number of points of interest; aspects which have also an impact on property prices [57]. In very specific geographic contexts such as the Spanish Mediterranean Arc, the impact of Airbnb location on property prices has also been explored and it is suggested that the accommodation price increases as the distance from the delimited tourist area (mainly the city's historical center) increases and the distance to coastal areas decreases [58]. Interestingly, in other contexts such as Hong Kong, other key determinants of Airbnb listing prices besides the location are the room type and the hosts' listings count [59].

11.2.5 Instagram

Instagram is a social network based on photo and video sharing, available for both mobile devices and web services. Users can upload content and share it with their followers or a select group of friends, as well as view, show interest, or "like" and comment on other users' posts [60]. When a user registers on the social network, a profile is created in which images and videos can be shared and stored. This profile can be public when any Instagram user can access it, or private when only followers and/or allowed friends can visit it, upon request. Photos and videos shared in the user profile can include text and/or tags or hashtags, which have a similar function to Twitter tags. For instance, it is possible to conduct a thematic search for photos or videos that have been tagged with a specific term. Likewise, when sharing a publication, the user can indicate the exact location from which the photo is being shared, and thus the post is associated to a "pin" or location marker. This provides the option for searching all posts generated from a specific location or area. Tags, location markers and other users' profiles can be searched on the social network website itself. Other features worth noting from Instagram are the timestamped content; the possibility to rank the most influential publications through the number of interactions (likes or comments); and the potential to analyze photographs specifically related to certain subjects, hot topics, and/or activities through the associated tags. Likewise, each photograph has an identification code and a link to the original post, which allow the contextual analysis of the content.

In view of the above, the visual and textual information collected from Instagram has the potential to provide clues on the types of activity people engage in when they visit certain urban spaces; to pinpoint meeting points and landmarks as well as relevant features of the urban image; to infer the collective perception about the environment; to characterize urban spaces according to citizens' habits and/or routines; and, to gain an overall picture of the predominant user profiles in a given space or urban area.

Previous research in urban studies has explored the possibilities offered Instagram shared photographs to address issues such as the identification of tourist areas and how they are perceived [61, 62] and the identification of relevant urban elements in public spaces [63, 64]. For instance, a study conducted in Granada has evidenced the relevance that tourist areas and points of interest have in the image of the city through the outstanding volume of shared photographs that include elements in those areas [61]. In a similar research line focused on the tourist phenomena, a comparative study between Instagram images shared in Algarve in Portugal and Costa del Sol in Spain demonstrated the potential of user-shared photographs for constructing the image of touristic cities as a claim for visitors [62]. A different conclusion has been drawn from Instagram data by a research conducted in Lodz in Poland, which shows that its content does not necessarily reflect the overall general urban space image, but local and specific places, because a considerable share of photos with outdoor city views were focused on newly established places and very specific elements [63]. Finally, a research focused on Melbourne Street art demonstrates that through the spatial and temporal features of Instagram images, it is possible to identify historically significant events related to street art culture in the city [64].

11.3 Data Variables: Diversity and Quality

The key characteristics and main data variables of each social network dealt with in the above section can be grouped into five types: identifier code [ID]; locative variables [LOC]; data timestamp [TEMP]; data format (text [TEX], photographs [FOT] or ratings [VAL]) and data categories [CLAS]. As a summary, the following Table 11.1 lists the data variables and their corresponding characteristics for each of the five exemplary social networks.

Figure 11.2 shows the different urban-related topics that can be dealt with by using the geolocated data of the five above-described social networks and their variables.

The quality and usability of the data variables for research in urban studies often depend on the social network functionality itself and the case study under analysis.

In the first case, the data provided by Foursquare, for example, are classified into hierarchical categories, whereas those of the other four social networks do not have a structured categorization. Thus, verification, filtering and classifying the data prior analysis are often required. For instance, data from Google Places, with over

Table 11.1 Diversity of data variables useful for analyzing urban phenomena. *Source* Authors, based on the work of Martí et al. [10, 65]

	Foursquare	Google places	Twitter	Instagram	Airbnb
Location [LOC]	Longitude	Longitude	Longitude	Location pin, marker	Longitude
	Latitude	Latitude	Latitude		Latitude
	Street, number, neighborhood, district, city, country	Street, number, neighborhood, district, city, country	City, country		City, country
Temporality [TEMP]	Accumulated data since venue creation	–	Tweet publication time (day and hour)	Image publication time (day and hour)	Date and time the property was registered in database
Data [TXT,FOT,VAL]	Venue name	Place name	Tweets (text)	Image or video	Property name
	Number of check-ins and visits	Number of comments	Retweets	Number of likes	Average daily rate
	Number of visitors or users	–	Number of tweets per user	Number of images per location pin	Maximum guests allowed
	Rating	Rating	–	–	Rating
	Photographs	Photographs	Photographs	Photographs	Photographs
	Tips	–	–	–	–
Info classification [CLAS]	Main category, Category	Types	Language, hashtags (#) and mentions (@)	Hashtags (#)	Property Type and Listing Type
Data ID [ID]	Venue ID	Place ID	Tweet ID	Image ID and URL	Property ID
					Host ID

a hundred non-structured place tags, can become approachable if the places are classified into fewer categories (e.g., the 10 place categories of Foursquare [37]).

In the second case, the availability and quality of the data within the case study area are crucial for deciding whether a social network or a specific variable is useful or not. For instance, there are geographical locations in which the penetration of certain social networks is not enough and, therefore, the data from that specific source may not be representative or valid to draw conclusions.

These challenges in the quality of the variables can be addressed if different sources are used. Indeed, the social networks covered in this work have proven to be complementary when combined and used as superimposed layers of information, offering a thorough understanding of the urban reality [66, 67].

	Foursquare	Google Places	Twitter	Instagram	Airbnb
Preferences about activities and urban spaces					
What are the most relevant public spaces and economic activities according to the number of users and their rating [LOC, VAL]?	●	●			●
Identification of social activities that are carried out preferably in some urban spaces [LOC, FOT]	●	●		●	
Identification of attractive places close to the non-regulated accommodation according to their rating [LOC, VAL]	●	●			●
Characterization of areas, spaces and economic activities according to user preferences [LOC, TXT, CLAS, VAL]	●	●	●		
Urban image and its perception					
Physical characterization of urban spaces [LOC, TXT, FOT]	●	●	●	●	●
Opinion and perception about urban spaces or economic activities [LOC, TXT]	●	●	●	●	●
Identification of predominant user profile in an urban environment [LOC, FOT]	●	●	●	●	
Identification of characteristic elements of public spaces [LOC, FOT]	●	●		●	
Spatial and temporal patterns of urban spaces use					
Affluence of users who have visited a place [LOC, VAL]	●		●		
Location of users in public or private spaces [LOC, TXT, CLAS]	●		●	●	
Identification of temporary patterns of presence in public or private spaces [LOC, TEMP]	●		●		
Supply and demand of economic activities					
What are the most demanded types of economic activities [LOC, VAL]	●	●			●
Patterns of location of economic activities [LOC]	●	●			●
Characterization of the supply of economic activities according to types [LOC, CLAS]		●			●
Characterization of the demand for economic activities according to types [LOC, CLAS]	●				
Characterization of the tourist supply, accommodation and activities [LOC, CLAS]	●	●			●
Location of areas of specialization according to economic activities [LOC, CLAS, TXT]		●			
Identification of activity areas					
Characterization of urban areas according to habits or routines [LOC, TXT, FOT]			●	●	
Characterization of urban areas according to their economic specialization [LOC, CLAS, TXT]	●	●			●
Identification of meeting points in the city [LOC, TXT, VAL, FOT]	●		●	●	
Characterization of urban areas according to its attractiveness for social activities [LOC, CLAS]	●		●	●	
Identification of areas of tourist specialization [LOC, CLAS]	●	●		●	●

Fig. 11.2 Urban phenomena that can be analyzed with the five social networks selected. *Source* Authors

11.4 Concluding Remarks

The impact of online social networks as new technological tools for modern society and the changes that they have caused in the way we communicate, share, learn and consume information have influenced our social behavior, not only in the private context but also in the public sphere of the city. Often, the social activities people engage in are conditioned by the idea of sharing where they go, what they do, who they meet and what they are thinking, leaving behind digital footprints that can be quite useful to understand how the city is used and perceived by its citizens. From the wide range of social networks that exist at the present time, this chapter aimed to select those that represent an opportunity as data sources for the study of urban phenomena.

Specifically, five social networks have been covered: Foursquare, Google Places, Twitter, Airbnb and Instagram. These social networks share four relevant characteristics: (1) the data include locative information, (2) the information is open and available to be collected without major restrictions or privacy issues if the analysis is performed on aggregated datasets, (3) the data reflect certain aspects of urban and/or human activities and (4) they are mostly used in mobile devices, so users can be considered as sensors and their data as crowd-sourced sensory information. Among the many topics that can be covered by these five social networks, Foursquare allows for the depiction of urban preferences; Google Places data are key for analyzing patterns and types of economic and urban activity; Twitter is useful for the study of spatiotemporal traces of people movement and gathering insights about the public opinion; Airbnb offers an overall understanding of phenomena related to non-regulated accommodation in the city; and, Instagram photographs often show evidence of how people perceive the urban environment and what elements in urban spaces are most significant.

Previous research showcases the many possibilities that social network data have for urban studies, from the perspective of urban public spaces, citizens' preferences, opinion, mobility and social behavior, among others, considering the vast amount of available information about location, temporality, type of data and data classification. Moreover, social networks like Twitter, Foursquare and Instagram are commonly explored in papers in the fields of geography, remote sensing, transportation and urban planning [68], thus, allowing for an interdisciplinary approach to city dynamics. These user-updated geolocated data also offer new opportunities for integration with other open data, such as governmental or administrative data, thus updating and complementing information that may quickly become obsolete.

All in all, many different pathways through which these data provide clues about the current reality of the city can contribute to inform future decision-making in planning processes, aiming to consider people preferences and activity traces as key sources of information.

References

1. Miniwatts Marketing Group: World Internet Users Statistics and 2020 World Population Stats. https://www.internetworldstats.com/stats.htm. Accessed 17 Dec 2020
2. We Are Social: Digital 2020: 3.8 billion people use social media—We Are Social. https://wearesocial.com/blog/2020/01/digital-2020-3-8-billion-people-use-social-media. Accessed 17 Dec 2020

3. Rubio Gil, Á.: Generación digital: patrones de consumo de Internet, cultura juvenil y cambio social. Rev. Estud. Juv. **88** (2010) (Ejemplar Dedic. a Juv. y Nuevos Medios Comun. págs. 201–221 2010). ISSN-e 0211–4364
4. Goodchild, M.F.: Citizens as sensors: web 2.0 and the volunteering of geographic information. GeoFocus **7**, 8–10 (2007). https://doi.org/10.1007/s10708-008-9190-4
5. Cheng, Z., Caverlee, J., Lee, K., Sui, D.Z.: Exploring millions of footprints in location sharing services. In: Fifth International AAAI Conference on Weblogs Social Media (2011)
6. Li, D., Liu, J.: Uncovering the relationship between point-of-interests-related human mobility and socioeconomic status. Telemat. Inform. **39**, 49–63 (2019). https://doi.org/10.1016/j.tele.2019.01.001
7. Longley, P.A., Adnan, M., Lansley, G.: The geotemporal demographics of Twitter usage. Environ. Plan. A. **47**, 465–484 (2015). https://doi.org/10.1068/a130122p
8. Järv, P.: Mining Tourist Behavior from Foursquare Check-ins
9. Blanford, J.I., Huang, Z., Savelyev, A., MacEachren, A.M.: Geo-located tweets. Enhancing mobility maps and capturing cross-border movement. PLoS One **10** (2015). https://doi.org/10.1371/journal.pone.0129202
10. Martí, P., Serrano-Estrada, L., Nolasco-Cirugeda, A.: Social media data: challenges, opportunities and limitations in urban studies. Comput. Environ. Urban Syst. **74**, 161–174 (2019). https://doi.org/10.1016/j.compenvurbsys.2018.11.001
11. Yuan, Y., Lu, Y., Chow, T.E., Ye, C., Alyaqout, A., Liu, Y.: The missing parts from social media-enabled smart cities: who, where, when, and what? Ann. Am. Assoc. Geogr. **110**, 462–475 (2020). https://doi.org/10.1080/24694452.2019.1631144
12. Nguyen, T.V.T., Han, H., Sahito, N.: Role of urban public space and the surrounding environment in promoting sustainable development from the lens of social media. Sustainability **11** (2019). https://doi.org/10.3390/su11215967
13. Chen, M., Arribas-Bel, D., Singleton, A.: Understanding the dynamics of urban areas of interest through volunteered geographic information. J. Geogr. Syst. **21**, 89–109 (2019). https://doi.org/10.1007/s10109-018-0284-3
14. Santos, F.A., Silva, T.H., Loureiro, A.A.F., Villas, L.A.: Automatic extraction of urban outdoor perception from geolocated free texts. Soc. Netw. Anal. Min. **10**, 88 (2020). https://doi.org/10.1007/s13278-020-00702-2
15. Jang, K.M., Kim, Y.: Crowd-sourced cognitive mapping: a new way of displaying people's cognitive perception of urban space. PLoS One **14** (2019). https://doi.org/10.1371/journal.pone.0218590
16. Rahmat, H.: Open online platforms and the collaborative production of micro urban spaces: towards an architecture of civic engagement. In: Open Cities|Open Data: Collaborative Cities in the Information Era, pp. 107–128. Palgrave Macmillan (2019). https://doi.org/10.1007/978-981-13-6605-5_5
17. Alonso-Almeida, M. del M., Borrajo-Millán, F., Yi, L.: Are social media data pushing overtourism? The case of Barcelona and Chinese Tourists. Sustainability **11** (2019). https://doi.org/10.3390/SU11123356
18. Yang, Y., Heppenstall, A., Turner, A., Comber, A.: Who, where, why and when? using smart card and social media data to understand urban mobility. ISPRS Int. J. Geo-Inf. **8**, 271 (2019). https://doi.org/10.3390/ijgi8060271
19. Lieske, S.N., Leao, S.Z., Conrow, L., Pettit, C.: Assessing geographical representativeness of crowdsourced urban mobility data: an empirical investigation of Australian bicycling. Environ. Plan. B Urban Anal. City Sci. (2019). https://doi.org/10.1177/2399808319894334
20. Radcliffe-Brown, A.: Structure and Function in Primitive Society. Cohen and West Ltd., London (1952)
21. Barnes, J.: Class and committees in a Norwegian Islan Parish. Hum. Rel. **7**, 39–58 (1954)
22. Ponce, I.: Monográfico: Redes sociales (2012)
23. Real Academia Española. https://www.rae.es/. Accessed 17 Dec 2020
24. IAB Estudio: Estudio Anual Redes Sociales 2020. IAB Spain **71** (2020)

25. Ebrahimpour, Z., Wan, W., Velázquez García, J.L., Cervantes, O., Hou, L.: Analyzing social-geographic human mobility patterns using large-scale social media data. ISPRS Int. J. Geo-Inf. **9**, 125 (2020). https://doi.org/10.3390/ijgi9020125
26. Khan, N.U., Wan, W., Yu, S.: Location-based social network's data analysis and spatio-temporal modeling for the mega city of Shanghai. China. ISPRS Int. J. Geo-Inf. **9**, 76 (2020). https://doi.org/10.3390/ijgi9020076
27. Beltrán López, G.: Geolocalización y redes sociales. Un mundo social, local y móvil. Bubok, España (2012)
28. Vanhoof, M., Godoy-lorite, A., Murcio, R., Iacopini, I., Zdanowska, N., Raimbault, J., Milton, R., Arcaute, E., Vanhoof, M., Godoy-lorite, A., Murcio, R., Iacopini, I., Zdanowska, N., Vanhoof, M., Godoy-lorite, A., Murcio, R., Iacopini, I.: Using Foursquare data to reveal spatial and temporal patterns in London. NetMob. **6** (2019)
29. Novović, O., Grujić, N., Brdar, S., Govedarica, M., Crnojević, V.: Clustering foursquare mobility networks to explore urban spaces. In: Advances in Intelligent Systems and Computing, pp. 544–553. Springer (2020). https://doi.org/10.1007/978-3-030-45697-9_53
30. Yang, L., Marmolejo Duarte, C.: Identifying tourist-functional relations of urban places through Foursquare from Barcelona. GeoJournal **9** (2019). https://doi.org/10.1007/s10708-019-10055-9
31. Jae-Hee, C., Il-Jung, S.: Clustering foursquare users' collective activities: a case of Seoul. J. Bigdata **5**, 55–63 (2020). https://doi.org/10.36498/kbigdt.2020.5.1.55
32. Cerrone, D., Baeza, J.L., Lehtovuori, P.: Optional and necessary activities: operationalising Jan Gehl's analysis of urban space with Foursquare data. Int. J. Knowl. Based Dev. 68–79 (2020). Inderscience Publishers. https://doi.org/10.1504/IJKBD.2020.106836
33. Üsküplü, T., Terzi, F., Kartal, H.: Discovering activity patterns in the city by social media network data: a case study of Istanbul. Appl. Spat. Anal. Pol. **13**, 945–958 (2020). https://doi.org/10.1007/s12061-020-09336-5
34. Martí, P., García-Mayor, C., Nolasco-Cirugeda, A., Serrano-Estrada, L.: Green infrastructure planning: unveiling meaningful spaces through Foursquare users' preferences. Land Use Pol. **97**, 104641 (2020). https://doi.org/10.1016/j.landusepol.2020.104641
35. Barreneche, C.: Una página Web para cada lugar en el mundo: Google ,codificación y comodificación del espacio. Actas del II Congr. Int. sobre Imagen, Cult. y Tecnol. 231–241 (2012)
36. Copparoni, M.: Google Maps, cada vez más utilizado para decidir una compra. https://www.lavoz.com.ar/negocios/google-maps-cada-vez-mas-utilizado-para-decidir-una-compra. Accessed 17 Dec 2020
37. Serrano Estrada, L., Bernabeu Bautista, Á., Ciriquián, P.M.: Actividades económicas y urbanas en ejes estructurantes metropolitanos. La aportación de los datos geolocalizados de Google Places. Urbano **23**, 80–97 (2020). https://doi.org/10.22320/07183607.2020.23.42.07
38. Hidalgo, C.A., Castañer, E., Sevtsuk, A.: The amenity mix of urban neighborhoods. Habitat Int. **106**, 102205 (2020). https://doi.org/10.1016/j.habitatint.2020.102205
39. Wilson, L., Danforth, J., Harvey, D., Licalzi, N.: Quantifying the urban experience: establishing criteria for performance based zoning. Simul. Ser. 237–244 (2018). https://doi.org/10.22360/simaud.2018.simaud.031
40. Landsman, D., Kats, P., Nenko, A., Sobolevsky, S.: Zoning of St. Petersburg through the prism of social activity networks. Proc. Comput. Sci. **178**, 125–133 (2020). https://doi.org/10.1016/j.procs.2020.11.014
41. van Weerdenburg, D., Scheider, S., Adams, B., Spierings, B., van der Zee, E.: Where to go and what to do: extracting leisure activity potentials from Web data on urban space. Comput. Environ. Urban Syst. **73**, 143–156 (2019). https://doi.org/10.1016/j.compenvurbsys.2018.09.005
42. Caldevilla Domínguez, D.: Las Redes Sociales. Tipología, uso y consumo de las redes 2.0 en la sociedad digital actual **33**, 45–68 (2010)
43. Kovács-Győri, A., Ristea, A., Havas, C., Resch, B., Cabrera-Barona, P.: #London2012: towards citizen-contributed urban planning through sentiment analysis of twitter data. Urban Plan. **3**, 75–99 (2018). https://doi.org/10.17645/up.v3i1.1287

44. Plunz, R.A., Zhou, Y., Carrasco Vintimilla, M.I., Mckeown, K., Yu, T., Uguccioni, L., Sutto, M.P.: Twitter sentiment in New York City parks as measure of well-being. Landsc. Urban Plan. **189**, 235–246 (2019). https://doi.org/10.1016/j.landurbplan.2019.04.024
45. Zivanovic, S., Martinez, J., Verplanke, J.: Capturing and mapping quality of life using Twitter data. GeoJournal **85**, 237–255 (2020). https://doi.org/10.1007/s10708-018-9960-6
46. Osorio-Arjona, J., García-Palomares, J.C.: Social media and urban mobility: using twitter to calculate home-work travel matrices. Cities **89**, 268–280 (2019). https://doi.org/10.1016/j.cities.2019.03.006
47. Wang, Q., Phillips, N.E., Small, M.L., Sampson, R.J.: Urban mobility and neighborhood isolation in America's 50 largest cities. Proc. Natl. Acad. Sci. USA **115**, 7735–7740 (2018). https://doi.org/10.1073/pnas.1802537115
48. Goh, G., Koh, J.Y., Zhang, Y.: Twitter-informed crowd flow prediction. In: IEEE International Conference on Data Mining Workshops, ICDMW, pp. 624–631. IEEE Computer Society (2019). https://doi.org/10.1109/ICDMW.2018.00097
49. Martín, A., Julián, A.B.A., Cos-Gayón, F.: Analysis of Twitter messages using big data tools to evaluate and locate the activity in the city of Valencia (Spain). Cities **86**, 37–50 (2019). https://doi.org/10.1016/j.cities.2018.12.014
50. Bhowmick, D., Winter, S., Stevenson, M.: Using georeferenced twitter data to estimate pedestrian traffic in an urban road network. In: Leibniz International Proceedings in Informatics, LIPIcs. Schloss Dagstuhl- Leibniz-Zentrum fur Informatik GmbH, Dagstuhl Publishing (2020). https://doi.org/10.4230/LIPIcs.GIScience.2021.I.1
51. Airbnb Inc.: Get to know Airbnb. https://www.airbnb.com/d/howairbnbworks. Accessed 03 July 2020
52. Wortham, J.: Matching Travelers With Rooms, via the Web—The New York Times. https://www.nytimes.com/2011/07/25/technology/matching-travelers-with-rooms-via-the-web.html. Accessed 17 Dec 2020
53. Airbnb Inc.: What do the different home types mean? https://www.airbnb.co.uk/help/article/317/what-do-the-different-home-types-mean?ibbe=1&topic=199. Accessed 01 June 2020
54. Quattrone, G., Greatorex, A., Quercia, D., Capra, L., Musolesi, M.: Analyzing and predicting the spatial penetration of Airbnb in U.S. cities. EPJ Data Sci. **7**, 31 (2018). https://doi.org/10.1140/epjds/s13688-018-0156-6
55. Gyódi, K.: Airbnb in European cities: business as usual or true sharing economy? J. Clean. Prod. **221**, 536–551 (2019). https://doi.org/10.1016/j.jclepro.2019.02.221
56. Li, H., Srinivasan, K.: Competitive dynamics in the sharing economy: an analysis in the context of airbnb and hotels. Mark. Sci. **38**, 365–391 (2019). https://doi.org/10.1287/mksc.2018.1143
57. Dudás, G., Vida, G., Kovalcsik, T., Boros, L.: A socio-economic analysis of Airbnb in New York City. Reg. Stat. **7**, 135–151 (2017). https://doi.org/10.15196/RS07108
58. Perez-Sanchez, V., Serrano-Estrada, L., Marti, P., Mora-Garcia, R.-T.: The what, where, and why of Airbnb price determinants. Sustainability **10**, 4596 (2018). https://doi.org/10.3390/su10124596
59. Cai, Y., Zhou, Y., Ma, J., Scott, N.: Price determinants of Airbnb listings: evidence from Hong Kong. Tour. Anal. **24**, 227–242 (2019). https://doi.org/10.3727/108354219X15525055915554
60. Instagram Inc.: ¿Qué es Instagram?|Ayuda de Instagram. https://www.facebook.com/help/instagram/424737657584573. Accessed 18 Dec 2020
61. Cantón-Correa, F.-J., Alberich-Pascual, J.: Construcción social de la imagen de una ciudad a través de Instagram: el caso de Granada The social construction of the image of a city through Instagram: The case of Granada 1699–2407 (2019). https://doi.org/10.3145/epi.2019.ene.08
62. Iglesias-Sánchez, P.P., Correia, M.B., Jambrino-Maldonado, C., de las Heras-Pedrosa, C.: Instagram as a co-creation space for tourist destination image-building: Algarve and costa del sol case studies. Sustainability **12**, 1–26 (2020). https://doi.org/10.3390/su12072793
63. Zasina, J.: The Instagram image of the city. Insights from Lodz, Pol. Bull. Geogr. **42**, 213–225 (2018). https://doi.org/10.2478/bog-2018-0040
64. Honig, C.D.F., MacDowall, L.: Spatio-temporal mapping of street art using Instagram. First Mon. **22**, (2017). https://doi.org/10.5210/fm.v22i3.7072

65. Martí Ciriquián, P., Nolasco-Cirugeda, A., Serrano-Estrada, L.: Los datos geolocalizados de las redes sociales en el estudio de cuestiones urbanas complejas: cinco temas, cinco redes. ACE- Archit. Ciudad y Entorno. **14**, 35–60 (2019). http://doi.org/10.5821/ace.14.41.8217
66. Martí, P., García-Mayor, C., Serrano-Estrada, L.: Taking the urban tourist activity pulse through digital footprints. Curr. Issues Tour. (2020). https://doi.org/10.1080/13683500.2019.1706458
67. Martí, P., García-Mayor, C., Serrano-Estrada, L.: Identifying opportunity places for urban regeneration through LBSNs. Cities **90**, 191–206 (2019). https://doi.org/10.1016/j.cities.2019.02.001
68. Owuor, I., Hochmair, H.H.: An overview of social media apps and their potential role in geospatial research (2020). https://doi.org/10.3390/ijgi9090526

Chapter 12
Introduction to Sentiment Analysis Covering Basics, Tools, Evaluation Metrics, Challenges, and Applications

Akrati Saxena, Harita Reddy, and Pratishtha Saxena

Abstract Sentiment analysis has been applied to the datasets collected from social networking websites to get valuable insights. The exemplary growth of social networking has attracted researchers, and there has been a vast contribution in this area. In this chapter, we discuss the basics of sentiment analysis and its methodology, including data collection, data pre-processing, and feature extraction methods. Next, we discuss enhancement techniques for sentiment analysis, including text categorization, feature selection, data integration, ontology-based approaches, and so on. The chapter further provides information about available sentiment analysis tools, challenges, and evaluation metrics. We also discuss sentiment analysis applications and insights for further attention.

12.1 Introduction

The idea of assessing opinions dates back to World War II, where individuals attempted gauging public opinion for political motives [1]. Sentiment analysis is the analysis of subjective details in an authors' writing to understand their opinion, outlook, and mood concerning the subjective entities. As the World Wide Web (WWW) advanced, the period after 2004 saw a surge in the volume of publications focusing on sentiment analysis using automated computer-aided techniques, especially for analyzing online customer reviews. Over a period of time, the applications have diversified into broader areas of finance, entertainment, marketing, psychology,

A. Saxena (✉)
Department of Mathematics and Computer Science, Eindhoven University of Technology, Eindhoven, The Netherlands
e-mail: a.saxena@tue.nl

H. Reddy
Surat, Gujarat, India

P. Saxena
G. L. Bajaj Institute of Technology and Management, Greater Noida, India
e-mail: pratistha.saxena@glbitm.ac.in

A. Biswas et al. (eds.), *Principles of Social Networking*,
Smart Innovation, Systems and Technologies 246,
https://doi.org/10.1007/978-981-16-3398-0_12

security, public policy, health care, and so on [2–6]. Similarly, the techniques have also advanced from traditional sentiment classification to algorithms under natural language processing (NLP), machine learning, and deep learning. Recently, sentiment analysis has been extensively applied to (i) social media platforms, such as Facebook and Twitter, (ii) recommendation platforms, such as Amazon, Netflix, and Spotify, (iii) blogs websites, such as Medium, and (iv) public discussion forums, such as Reddit. Initially, sentiment classification focused on identifying the positive, negative, and neutral sentiments. The concept was extended to the areas of emotion mining, human-agent interaction [7], recognizing sarcasm, determining public mood [8], and affective states like author personality [9]. The growth of virtual communities requires such applications to adapt to larger online social networks. For example, a sentiment analysis tool, called SentiHealth-Cancer [10], monitors cancer patients' moods via Facebook community posts. Conversely, information from social networks is useful to bolster the sentiment analysis at the user level. Tan et al. [11] demonstrate the benefits of including social relationships for user-level sentiment analysis.

Sentiment analysis of a text document is done at varying levels of granularity, ranging from word-level classification to document-level and feature-level analysis. The methodology of performing sentiment analysis includes multiple steps starting from the collection of suitable domain-specific or multi-domain data. The data needs to undergo pre-processing for removing unnecessary parts of the documents which do not contribute to the sentiment of the text and cleaning the text for further analysis. This is followed by the extraction of a subset of features from the document, which play an essential role in indicating the sentiment expressed by the document. Then, sentiment analysis techniques are used to leverage these features and assign suitable sentiment classes to the documents [12].

There are multiple ways to enhance the performance of sentiment analysis, including the incorporation of ontology-based approaches, which can help in identifying the relationships between the domain concepts [13]. Finally, multiple evaluation metrics are available to assess the sentiment classification approaches. Several sentiment analysis tools have been designed by researchers for performing sentiment analysis or for assisting this process. Sentiment analysis has been successfully used in multiple fields, including business and IoT; still, there are several challenges in this topic that we will cover.

In this chapter, we present a systematic review on sentiment analysis, including (i) sentiment analysis levels, (ii) sentiment analysis methodology, (iii) enhancement methods, (iv) evaluation metrics, (v) sentiment analysis tools, (vi) challenges, and (vii) further applications in social networks. Lastly, we will conclude the chapters and will provide some insights for further exploring this research domain.

12.2 Sentiment Analysis Levels

The first step for analyzing sentiment is to classify the polarity of a given text document, and this classification can be done at multiple levels, such as word, sentence, document, and feature/aspect levels. This step mainly identifies the expressed sentiment in the given word, sentence, or document as neutral, positive, or negative. Next, we explain it in detail.

12.2.1 Word Level

In sentiment classification, a word is categorized by using the prior-polarity of the words and phrases. The sentiments at the word level are mainly assigned using the following two types of methods, (i) dictionary-based approaches and (ii) corpus-based approaches. For example, words such as 'happy', 'joy' are highly positive, and 'sad', 'hate' are highly negative.

In some cases, phrase-level sentiment analysis is done for a better understanding of the opinion. Word and phrase-level sentiment labels are further used to classify sentences and documents.

12.2.2 Sentence Level

Sentence-level sentiment analysis is applied in opinion mining by classifying a sentence as *subjective* or *objective*. The following are the examples of subjective and objective sentences and the sentiment conveyed by subjective sentences:

- Objective: (i) Tom went to the Taj hotel three days before. (ii) This car is blue.
- Subjective: (i) Taj hotel provides the best service. (ii) This is a very good car.

Subjective sentences can be classified as (i) positive or (ii) negative. For example,

- Positive: (i) Taj hotel provides the best service. (ii) This is a very good car.
- Negative: (i) Their service is poor. (ii) This car has bad mileage.

In social networking microblog data, emoticons are also used to identify the sentiment [14]. In the review dataset, voting-up or voting-down, or star rating can also be used in sentiment classification.

12.2.3 Document Level

In the document-level analysis, a document such as a blog, news article, and review is classified as expressing either negative, positive, or neutral sentiment on the basis of the overall sentiment that is expressed in the sentences by the writer. Each document focuses on a single topic, and the task is to classify the opinion with respect to the given topic. For example, we show examples of Amazon reviews for different classes from the "Casio Vintage Series Digital Grey Small Dial Men's Watch-A158WA-1Q (D011)" product.

– Positive: "This one's a classic. Styles may come and go this Vintage Casio will always have its own ardent followers. I am love understated but high on performance watches and this fits the bill. I received prompt delivery by seller Crystalarc Lifestyle and the piece was packed properly. In fact I got one from a relatively new manufactured batch."
– Negative: "The poor quality of the product is very visible. It was supposed to be a gift to someone very close, who owned a same Casio model. But this looked and felt too inferior to the old one. Really disappointed with the purchase. I returned the item and will not recommend it."

A document can also be labeled as 'None' if the opinion cannot be annotated as positive, neutral, or negative with respect to a topic or if the author does not express any opinion [15]. Emoticons, voting, and rating can be helpful in classifying the documents based on the sentiments.

12.2.4 Feature/Aspect Level

In Feature- or Aspect-level analysis, first, the object features in the given source text are determined [16]. Then, we evaluate if the opinions expressed on the identified features are neutral, negative, or positive. It performs finer-grained analysis as, in many cases, sentence- and document-level analysis are not able to classify the exact opinion of the writer. The main approaches in this direction can be classified as frequency-based, relation-based, model-based, and hybrid approaches.

12.3 Sentiment Analysis Methodology

In this section, we summarize the methodology followed to analyze the conveyed sentiment.

12.3.1 Data Collection

Researchers have performed a great amount of work on text-based sentiment analysis, but evaluation of sentiment can also be done on visuals, audio, and videos. There are multiple sources of data for sentiment analysis [17, 18]. Social networking websites like *Facebook* and *Twitter* allow people to connect with each other and post content, which can also include their personal opinions about issues ranging from politics to sports. Many messages posted on social media are short as compared to traditional text documents and may contain images or links to images and other multimedia. Twitter posts have a limit of 280 characters. People can also create their personal blogs and publish information or opinions on these pages. Forums or discussion boards like *Reddit* allow multiple users to discuss and express their opinions on topic threads.

There are several publicly available social network datasets. We can also extract data directly from social networking websites as websites such as Twitter provide an Application Programming Interface (API) to extract publicly available data from historical tweets as well as streaming data in real time [19, 20].[1] Twitter's Search API can be used to access historical tweets and has three variants for usage—standard, premium, and enterprise. Search API can be used to get tweets matching a query in the form of a JSON response and has limits on the number of requests made per time window. There are some libraries which can be leveraged for accessing the API, for example, *Tweepy*[2] and *python-twitter*[3] in Python, and *Twitter4J* [21] in Java. Likewise, Facebook Graph API has been used to query data such as comments on Facebook [22, 23].

Other sources of data that have been typically used for sentiment analysis include reviews written by customers on online websites, such as *IMDB*, *Amazon*, and *Flipkart*, and travel websites such as *TripAdvisor*. Another source of data for sentiment analysis is the news websites that publish full-fledged articles. Some of the websites may also have a comment section under the articles for readers to express opinions.

12.3.2 Data Pre-processing

Once the raw data is collected, the data has to be cleaned through the pre-processing step. This step has to be applied in the same way on both training and test data to be further sent for feature extraction. As part of data cleaning, unnecessary symbols such as digits or other special characters, including punctuation, are removed from the text. Non-textual content also needs to be removed as part of the pre-processing step if it is not being considered for further analysis [17]. Social media posts have informal text with spelling mistakes, slang words, abbreviations, URLs, etc., and the text may need some specialized pre-processing, unlike the conventional text documents. To enhance the prediction of text polarity, such irregularities in the text may be cleaned by methods such as removing hashtags and URLs, spell checking, and replacement

[1] https://developer.twitter.com/en/docs.

[2] http://docs.tweepy.org/en/latest/.

[3] https://python-twitter.readthedocs.io/en/latest/.

of slang terms [24]. The following are some of the extra steps that may be followed for Twitter data cleaning [20, 24–26]:

– Removal of non-English characters or tweets if the analysis is being done only on English language tweets;
– Removal of special characters used on Twitter, such as hashtags (#), mentions (@), and retweet symbols (RT);
– Removal of external links in the form of URLs and emoticons (however, emoticons may be leveraged for sentiment analysis);
– Replacement of elongated words caused due to repetition of letters back to original form as well as social media slang words such as ‘hlo’ (for hello);
– Expansion of well-known abbreviations and acronyms;
– Splitting words attached without spaces into separate words;
– Spell checking tweet words using a dictionary.

12.3.3 Feature Extraction and Selection

After pre-processing data, we need to extract the required features from the text for further analysis. After normalizing the casing of the text, the prominent steps as part of natural language processing (NLP) include stemming, lemmatization, and tokenization, while other measures such as removing stop words and identifying parts of speech may also be taken [27]. Tokenization involves decomposing a text document into tokens, which are smaller and meaningful units like words. A simple approach to tokenization is on the basis of space delimiter and punctuation marks. To bring the different derived words of the same base word to a common form, stemming and lemmatization may be used. Stemming involves converting certain word into its root form; an example of a stemming algorithm is Porter’s Stemmer.[4] While stemming is more of a crude approach in reducing the words by usually cutting off the ending of the words, lemmatization involves morphological analysis and reducing the word to its base *lemma* form. Stop words are the words like ‘is’ or ‘and,’ which are very frequently used in the language but do not contribute much to the meaning or emotion of the content, and hence are often removed in NLP tasks. Some approaches, however, retain stop words [25]. Some of the extracted features from the text include the following.

– Presence and frequency of unigrams, bi-grams, and n-grams are common features, where a sequence of n units is considered as an n-gram, which in the case of word n-grams is a sequence of n nearby words [28]. In character n-grams, the unit is a character.
– Presence and occurrence counts of terms can be used to find out the relevance of the terms in the given text. A variant is Term Frequency-Inverse Document Frequency (TF-IDF), which considers the frequency of a given term in the document and also

[4] Refer https://tartarus.org/martin/PorterStemmer/.

scales down the word's importance in relation to the text document if the word is commonly present in most documents.

– Neural network-based techniques are used to derive representations of words in a vector space or 'word embeddings' based on the similarity of words in a corpus; examples include the Skip-gram and the Continuous Bag of Words (CBOW) models [29].
– Syntax and style-based features of text including vocabulary richness, distributions by word lengths, function word frequency, character, and word-based features are one of the feature sets that can be used to determine sentiment [23].
– Negative and positive words, as well as words, expressions, and idioms expressing an opinion, can help in indicating the sentiment and opinion expressed in the given text [20, 26]. Negation will change the polarity of the sentiment expressed in the text.
– Part-Of-Speech (POS) tagging is used for labeling every word with its part of speech, e.g., adjective and adverb [30]. Some of these tags can be useful for predicting the subjectivity and sentiment expressed in the text [26].

Feature selection is leveraged to choose only those features that contribute toward classifying the input. It helps in enhancing the efficiency of learning from the data and also prevents the negative impact of noisy features and features that are irrelevant to the performance of the classifier [31]. Feature selection includes methods such as eliminating features based on the value of a metric, for example, Mutual Information (MI), and methods that find optimal feature subset based on prediction performance on the subset [32].

12.3.4 Sentiment Analysis Techniques

An essential step in the process of sentiment analysis is determining the sentiment conveyed by the document and assigning the document an appropriate sentiment class based on the available features. The techniques for this step can be broadly grouped into (i) machine learning (ML)-based techniques, (ii) lexicon-based techniques, and (iii) hybrid techniques [12]. While ML techniques leverage ML classification models, such as naive Bayes classifier, lexicon-based techniques leverage a dictionary of sentiment words or an opinion lexicon. Hybrid techniques, as the name suggests, leverage the advantages of both the ML and lexicon-based approaches. We will discuss sentiment analysis techniques in detail in the coming chapter.

12.4 Sentiment Analysis Enhancement

In the following subsections, we discuss some of the sentiment analysis enhancement techniques.

12.4.1 Data Pre-processing and Feature Selection

Firstly, data cleaning techniques to convert the text into a proper format for classification, which we discussed in Sect. 12.3 as part of the data pre-processing techniques, can help in improving the results of sentiment analysis techniques. Some parts of a textual document do not contribute to its meaning and are not useful for determining the sentiment expressed by the text, for example, HTML tags, and these elements can be removed as part of data cleaning [33]. Removal of hashtags and mentions, spell checking, and replacement of slang terms are some of the steps that are taken [24]. This helps in decreasing the noise and dimensions of the input text documents and aids classification. Data *transformation* techniques include removal of stop words and white spaces, stemming, etc., and *filtering* techniques include feature selection to choose only features having a higher contribution in determining text sentiment [33].

The idea behind feature selection is choosing only a portion of the available feature variables to obtain good sentiment analysis prediction results. For this, the importance of the individual features for the prediction task has to be determined, and features of lesser relevance may be eliminated. In filter methods, feature variables are ranked based on a certain metric, which indicates the relevance of each feature, and the features having metric values below a cutoff are eliminated [32]. Examples of two filter methods include correlation criteria such as the Pearson correlation coefficient and Mutual Information (MI). Wrapper methods leverage search algorithms to obtain the set of feature variables that maximize an objective function. Sequential selection algorithms include Sequential Floating Forward Selection (SFFS) [34, 35] and Sequential Feature Selection (SFS), and heuristic search algorithms include genetic algorithms [32, 36]. Other ways of obtaining improved feature selection include Feature Frequency, which considers the words that are most frequent in a text, Feature Presence, which considers the presence of words in a text, and Term Frequency-Inverse Document Frequency (TF-IDF) [13].

12.4.2 Text Categorization and Summarization

For opinion mining, Hu et al. [37] used a text summarization-based method where the importance of sentences in TripAdvisor comment data was computed based on the author, comment time, usefulness, and the sentence itself. The similarity between sentences was then computed based on both content and sentiment, and the sentences were clustered into groups, out of which representative sentences from the top clusters were selected. In another work, sentiment analysis was done on summaries of text documents [38]. Text documents were pre-processed, and after feature extraction, a fuzzy approach was used for text summarization, after which the sentiment of the summary was ascertained.

Pang et al. [39] used a text categorization approach to identify those sentences of a document that are subjective and ignored the objective parts to produce shorter extracts. These shorter extracts portray a more accurate picture of the sentiment expressed by the text and also prevent the polarity classifier from using irrelevant parts of the document. Similarly, in other works, text summarization has been used to reduce redundant information and extract the most relevant information from the text before performing sentiment analysis [40].

12.4.3 Ontology-Based Approaches

The field of semantic computing leverages computer science as well as social sciences for sentiment and opinion mining [41]. Ontologies can be utilized for modeling terms and the relationship between them, and they are used for characterizing knowledge on the semantic web [42]. The domain's ontology can be leveraged for fine-grained sentiment detection. Common-sense knowledge can be organized into entities, concepts, and conceptual primitives [43, 44]. SenticNet was one of the first resources for sentiment and opinion analysis based on common-sense reasoning for gaining insights on a huge amount of unstructured data of the Web [43]. OntoSenticNet is an ontology built on the basis of SenticNet with a clear definition of the relationship between concepts and sentiment and the ability to associate concepts with external sources like documents [44]. Dragoni et al. also used a combination of WordNet [45] and SenticNet to apply a fuzzy logic for ascertaining concept polarity in a domain, based on the knowledge graph for the domain [46].

12.4.4 Data Integration

Cho et al. [47] worked on enhancing sentiment analysis results by incorporating multiple sentiment lexicons including SenticNet [48], SentiWordNet [49], General Inquirer [50], etc., by standardizing them. They noted that their technique using an integrated dictionary performed better compared to using dictionaries separately for sentiment analysis of review data. Semantic web-based techniques have been used for integrating health data from multiple platforms for building an analytics framework for health, including a component for sentiment mining [51].

12.4.5 Crowdsourcing

Crowdsourcing is a technique using the perception and judgment of people for a task. It is used for annotating sentiment analysis data and also to obtain feedback regarding the performance of the proposed sentiment analysis models [13]. Crowdsourcing through Amazon Mechanical Turk (MTurk) can be used to obtain annotations for data like tweets by appropriate phrasing of questions for the workers [52]. An approach

Table 12.1 The confusion matrix

	Predicted positives	Predicted negatives
Actual positives	Number of true positive cases (TP)	Number of false negative cases (FN)
Actual negatives	Number of false positive cases (FP)	Number of true negative cases (TN)

using crowdsourcing for sentiment analysis was found to give more accuracy when compared to automatic sentiment analysis methods [53]. Crowdsourcing can be either done by paying the participants, as in MTurk, or on a volunteer basis, like in Crowd4u[5] [53].

12.4.6 User Characteristics

Using user profile-related information can be used to enhance sentiment analysis [13]. In their study on factors related to contributions by users on blogging communities, Kim et al. [54] harnessed the characteristics of online identities of the users. They noted that user involvement, social skills, and creativity had an impact on their contributions. User features based on the content of the users' tweets, geographic and graph characteristics can be used to determine spam users as well as 'non-personal' users, whose tweets or posts can be ignored to make sentiment analysis and opinion mining more accurate [55].

12.5 Evaluation Metrics

Here, we discuss metrics that have been used to assess various sentiment classification techniques in terms of their performance. Accuracy, Precision, Recall, and F1-score are four main metrics calculated using the confusion matrix shown in Table 12.1.

1. Accuracy: It is computed as the ratio of all the true predicted cases and all the cases.
$$Accuracy = \frac{TN + TP}{TN + TP + FP + FN}$$
2. Precision: It is computed as the ratio of the true positive predicted cases and all the positive predicted cases.
$$Precision = \frac{TP}{TP + FP}$$

[5] Refer https://crowd4u.org/en/.

3. Recall: It is computed as the ratio of true positive predicted cases and all actual positive cases.

$$Recall = \frac{TP}{TP + FN}$$

4. F1-Score: It is computed by calculating the harmonic average of the precision and the recall.

$$F1 = \frac{2 * Precision * Recall}{Precision + Recall}$$

5. AUC and ROC: Area under the curve (AUC) and receiver operating characteristic (ROC) curve also have been used along with the above-defined metrics [56].
6. Relative Error: Relative error is used to check the prediction efficiency for positive cases. It is defined as

$$RelativeError = \frac{|P - TP|}{TP}$$

7. Kappa criterion: It is also used to show the agreement between two classifiers, and it is computed as

$$Kappa = \frac{P_0 - P_e}{1 - P_e}$$

   where P_0 is the relative agreement of classifiers, and P_e is the hypothetical probability of chance agreement.
8. Jiang et al. [57] proposed a metric to measure the performance of their proposed method. The proposed metric takes emoticons as a benchmark, as given below.

$$e_b^d = \frac{1}{N_e^d} \sum_{i=1}^{N_e^d} e_{e_i}$$

   where N_e^d denotes the number of emoticons, e_i denotes one emoticon of microblog d, e_{e_i} denote the emotion vector, and e_b^d denotes the emotion vector of microblog d.
9. Ranking Loss [58, 59]: It computes the average deviation of the actual sentiment value versus the predicted sentiment value. Given a sentiment classification problem having m sentiment classes and n test data items, it is computed as

$$RankingLoss = \sum_{i=1}^{n} \frac{|y_i - \hat{y}_i|}{m * n}$$

   where y_i and $\hat{y}_i$ are actual and predicted sentiment for data instance i.
10. Mean Absolute Error (MAE): Marcheggiani et al. [60] used MAE and it is defined as

$$MAE(y, \hat{y}) = \frac{1}{m}\sum_{j=1}^{m}\frac{1}{Y_j}\sum_{y_i \in Y_j}|y_i - \hat{y}_i|$$

where y and $\hat{y}$ are the vector of true sentiment and predicted sentiment values, respectively, m is the count of unique sentiment classes in y, and $Y_j = \{y_i | y_i \in y, y_i = j\}$.

11. Least Absolute Errors (LAE): It is a simple metric, i.e., used in [61] and defined as follows:

$$LAE(y, \hat{y}) = \sum_{i=1}^{n}|y_i - \hat{y}_i|$$

12. Mean Squared Error (MSE): MSE is mainly used with regression models [62]. It is defined as

$$MSE(y, \hat{y}) = \frac{1}{n}\sum_{i=1}^{n}(y_i - \hat{y}_i)^2$$

13. Root Mean Square Error (RMSE): It is used in [63], and computed as

$$RMSE(y, \hat{y}) = \sqrt{\frac{1}{n}\sum_{i=1}^{n}(y_i - \hat{y}_i)^2}$$

14. Discounted Cumulative Gain: It measures the performance scores of top-k returned aspects [64]. It is defined as

$$DCG^k = \sum_{i=1}^{k}\frac{2^{rel(i)} - 1}{log_2(i + 1))}$$

where $rel(i)$ represents the relevance score of aspect i.

12.6 Sentiment Analysis Tools

Several sentiment analysis tools have been proposed and we cover some of the tools below.

1. Amplify Analytics: It is a marketing and data solution tool to analyze customer data for predictive analytics and marketing automation for customers' profit (available at https://www.Amplify-Analytics.com).
2. AlchemyAPI [65]: It can recognize text in around 100 different languages and labels the given text with positive, negative, and neutral opinions.

3. AlertRank [66]: Zhao et al. designed AlertRank web-based service for monitoring alerts and identifying severe alerts. The tool first extracts features from the data and then uses the XGBoost ranking algorithm to identify the severe alerts.
4. GATE [67]: General Architecture for Text Engineering (GATE) is an open-access system for text analysis. The storage substructure of GATE is based on TIPSTER architecture [68] that is also a natural engineering tool. It is known for its scalability, reusability, transparency, and robustness and has been used by small to large corporations for text analysis, semantic analysis, or general language processing (refer http://gate.ac.uk/sentiment/).
5. iFeel [69]: iFeel is a free Web application to detect sentiments and can be used for applying seven sentiment analysis methods available in research, including Emoticons, Happiness Index, PANAS-t, SASA, SentiWordNet, SentiStrength, and SenticNet. It is available at [70].
6. Jodange [71]: It is a Web tool that automatically extracts opinions from articles, microblogs, and social media (available at http://jodange.com).
7. LingPipe [72]: It is a java-based tool for text processing, including entity extraction, speech tagging, clustering, classification, etc., using computational linguistics. It automatically classifies tweets into categories and is known for scalability and fast speed.
8. Lithium Social Media Monitoring [73]: It is an integrated social media management system that is used for social publishing and social results. It integrates with Twitter to analyze Net Promoter Score (NPS) and Customer Satisfaction (CSAT) data and helps in improving customers' social media workflow (more details are at https://en.wikipedia.org/wiki/Khoros,_LLC).
9. NLTK [74]: Natural Language Toolkit (NLTK) is a very well-known, free, and open-source tool for natural language processing (NLP) methods, for example, tagging parts-of-speech, classification, parsing, tokenization, stemming, and clustering. It consists of more than 50 corpora and lexical resources that can be used for practice learning. It is also used in other related domains, such as artificial intelligence, information retrieval, cognitive science, and machine learning (refer http://www.nltk.org/).
10. NLP Toolsuite: It is designed by the JULIE Laboratory for applications such as semantic search, text mining, and information extraction [75] (refer https://julielab.de/Resources/).
11. Opinion Observer [76]: Liu et al. proposed this method for analyzing and exploring customers' opinions about a product. It shows the strengths as well as weaknesses of competing products in terms of different features.
12. OpenAmplify [77]: It is a well-known system for large-scale text and NLP analysis. The tool is useful in identifying significant topics, sentiments, perspectives, emotions, attributes, actions, and intentions contained in the provided text. It is available at https://www.crunchbase.com/organization/openamplify.
13. OpenNLP [78]: The Apache OpenNLP library is a tool used for processing natural language text and uses machine learning for supporting NLP tasks.
14. Opinion Finder [79]: This tool determines subjective sentences and also identifies different subjectivity aspects, including the subjectivity source and words

expressing positive or negative sentiments in these sentences (refer http://code.google.com/p/opinionfinder/).

15. OntoGen [80]: This tool is data-driven and is helpful to fill the gap between ontology editors, which can be complex, and the domain experts, who might not have the required skills for ontology engineering. The tool implements unsupervised and supervised methods for concept suggestion and concept visualization.
16. OpenDover: This extracts the semantics-based features from text documents, which could be blogs, news, articles, or websites. It is based on ontology and is useful in multiple domains such as health, economy, education, politics, and law. The labels indicate positive, neutral, or negative sentiments, and there is also a value that indicates the strength of the sentiment (refer http://opendover.nl/).
17. QDA Miner [81, 82]: It is a free software developed by Provalis Research that can be easily used for computer-assisted qualitative analysis of text data including blogs, news articles, journal articles, microblogs, as well as for image analysis. It is used in managing, coding, and analyzing qualitative data and provides GeoTagging (GIS) and Time-Tagging tools. It has been used to find out hidden patterns in text data. The latest version 5 of QDA Miner was released in December 2016. It is available at [83].
18. Social Mention [84]: It is a social media monitoring tool that search blogs, social bookmarks, social networking sites, comments, images, news, videos, microblogs, and the Web to know what people are talking about a brand on the Internet. It is available at https://brandmentions.com/socialmention/.
19. Stanford Parser and Part-of-Speech (POS) Tagger: This software is from the Stanford NLP group and is highly used for sentence parsing, and part-of-speech tagging for multiple languages. It is used in [85, 86] (refer http://nlp.stanford.edu/software/tagger.shtml).
20. SAS Sentiment Analysis Manager: SAS collects real-time data from webpages and social media platforms and extracts sentiment keywords to analyze the sentiments of customers. It is used in [87, 88].
21. Twendz [89]: Twendz is used for identifying the sentiments, topics of the tweet, and the adoption of the emoticons. It is released by Waggener-Edstrom firm and is available at http://twendz.waggeneredstrom.com/default.aspx?q=nodexl.
22. TweetFeel [90, 91]: It is a real-time Twitter search tool and provides a numerical score showing how negative or positive tweets are on a given topic. For example, if you search for a query 'covid-19', it will return to you that the current sentiment is around 80 percent positive (available at http://www.tweetfeel.com/).
23. Twitter sentiment analysis tool [14]: This tool helps in extracting the positive and negative sentiments associated with brands, products, or even topics on Twitter (available at https://twittersentiment.appspot.com/).
24. Twitrratr [92]: Twitrratr builds a set of positive and negative keywords and uses it to identify if a tweet is positive, neutral, or negative respective to the given topic (available at twitrratr.com).

25. TURKSENT [73]: It is an annotation tool to manually annotate sentiments for social media posts and use this corpus to support automatic labeling. It provides easy to use web interface, linguistic annotation capabilities, and detailed sentiment levels.
26. Tawlk/osae [93]: It is a Python library for sentiment classification on social microblog data (refer https://github.com/Tawlk/osae/).
27. Textir [94]: It is a tool for analyzing sentiments in the given text. It consists of (i) the 'mnlm' function that is used for sparse multinomial logistic regression, (ii) 'pls', which is a concise partial least-squares routine, and (iii) 'topics' that is utilized to efficiently estimate and select dimensions in latent topic models.
28. Weka [95]: Waikato Environment for Knowledge Analysis (Weka) is a machine learning software in Java and it was designed at the University of Waikato in New Zealand. It has built-in tools for machine learning tasks such as data preprocessing, classification, clustering, feature selection, regression, visualizing data, and algorithms for data mining and predictive modeling.

12.7 Challenges

Here, we explain some of the challenging issues faced in Sentiment Analysis.

- **Keyword Selection**: In sentiment analysis, most of the techniques use a set of keywords to classify the text. However, identifying the right set of keywords for the given text is not an easy task. One of the main reasons is that a complete sentence and its words, when considered individually, might have a completely different meaning. Some of the recent works in this direction are [96–98]. Similarly, in some cases, a sentiment word may have a completely opposite meaning if it is analyzed with respect to the given topic [99]. These kinds of cases should be handled carefully to improve efficiency.
- **Interrogative and Conditional Sentences**: An interrogative sentence does not necessarily have an explicit positive or negative sentiment; however, the keyword may have positive or negative sentiment in opinion mining [100]. Conditional sentences also create a similar issue in sentiment mining [101].
- **Comparative Opinion**: Similar to conditional sentences, comparative opinion analysis is also tricky [102]. For example, saying "not tasty" doesn't mean that it is not eatable, or "not so good" doesn't mean that it is bad. The current methods consider the negation as a preferred way, though in some cases, it couldn't accurately identify the writer's opinion.
- **Context-dependent sentiment**: Sometimes, the sentiment and polarity of a document or a statement might be completely opposite for two different people. For example, "Honda Activa is a better bike than Bajaj Pulsar" is a comparative sentence. This is a positive statement for Honda company, but not for Bajaj enterprise. In the context-dependent text, the sentiment and polarity vary from person to person [103].

- **Slang Usage and Word Limit**: Twitter have a constraint that each tweet might not be longer than 140 characters. People frequently use short-forms and abbreviations to tweet the complete message. It requires specific pre-processing steps to interpret people's opinions in different contexts. Slang usage and incorrect grammar also need to be taken care of during pre-processing [89, 104].
- **Spam Reviews or Microblogs**: Spam reviews or microblogs can be posted to make or devastate the social reputation of a company, individual, product, or brand. Researchers have proposed methods for spam detection [105]. Spam removal is necessary while quantifying the opinion of users about a product, or an object [106, 107].
- **Domain Specific**: Sentiment interpretation is also dependant on the domain and the meaning of words and sentences might change depending on the context [108]. For example, "It was a blast" is a good movie review but not good if there is any criminal activity. We discuss some related future directions in the next section.
- **Sentiment Analysis in Multi-lingual Setting**: A multi-lingual text comes with its own challenges. The first issue is the language recognition for each sentence, keywords, or phrases in the text. Next, a multi-lingual text might have phrases which have a different interpretation for different kinds of people. In social networks, user profile features, such as gender [109, 110], personality [111, 112], location [113], community [114, 115], and opinion in the past [116] can enhance the quality of SA.
- **Multiple Opinions in a Sentence**: In some cases, one sentence can express multiple opinions, and we need to identify all clauses to estimate the strength of opinion in each clause which are further used to calculate the overall sentiment expressed by the sentence. As an example, the sentence "The picture quality of this camera is great, and it also has a good battery backup, however, the viewfinder is too small for such a good camera" conveys both positive and negative sentiments on the product in the same sentence [108].
- **Negation Handling**: Negation is an operation in grammar. The negative form of a statement represents the opposite or falsity of the given statement. Negation handling is tricky in sentiment analysis as two sentences having completely opposite meaning might just differ by only one token. For example, "I am happy" and "I am not happy" just differ by one word. In some other cases, negation might be expressed using other delicate words like 'avoid' than just using simple words like 'no' and 'never', for example, "It avoids the steep walking path". Negation words affect the polarity of the sentence. Some recent works in this direction are [117–119]. The features for negation modeling can be categorized as (i) negation features, (ii) shifter features, and (iii) polarity modification features [120].
- **Sarcasm Analysis**: In sarcasm, irony or jokes, a few sentences or words transform the meaning of a sentence completely, and even of the document. These are very frequently used on social media. For example, if someone tweets, "I made a genius choice of selling my car just before I chose to move to another city", it is sarcastic as the person is not happy with his action and thinks that it was a bad decision. Humans can easily detect sarcasm or jokes. However, in automatic text

mining, such sentences are difficult to identify and have many challenges while pre-processing [121–123].

– **Review Analysis**: Sometimes there are no sentiment words like good, happy, enjoy, sad, and bad, in a given sentence or a document though the sentences may still convey a positive or negative feedback about some product, service, or policy [124]. A review might also be context-dependent and can have different interpretations for different involved parties.

12.8 Applications

In this section, we explain the major practical applications of sentiment analysis and opinion mining.

1. **Market Prediction:** Determination of certain specific types of public moods on social networking websites. Twitter data has been used to increase the accuracy of Dow Jones Industrial Average (DJIA) predictions [125]. Sentiment analysis of finance-related news has been incorporated to study stock predictions [3, 4, 126].
2. **Business and Market Intelligence:** As part of marketing research, understanding consumers' attitude toward a brand or product is important [2]. With micro-blogging websites being a platform for consumers to express opinions, sentiment classification and opinion summarization can help in making better decisions for marketing products by assessing the competition and identification of opportunities [127]. Sentiment mining on social networks can help in identifying public response to events such as product releases and keep track of user complaints to alleviate the situation before the complaints go viral [17].
3. **Fake News Detection and Mitigation:** Sentiment scores have been used for determining the credibility of news on social media [128, 129], and deception in online reviews [130]. It has also been applied to verify users' emotions toward rumor and non-rumor posts related to an event [131, 132]. Differences in how actual human users and social media bots express sentiment in their tweets can help in detecting bots spreading fake news [133].
4. **Detecting Opinion Spamming:** Sentiment study has been incorporated into determining misleading online reviews posted by spammers to promote or degrade any business [134]. It also discovers communities that are likely to be engaged in opinion spamming in reviews [135].
5. **Online Communities:** People who share similar points of view or beliefs are likely to form online communities, and sentiment detection can help in determining such communities [136]. While community discovery on social networks is usually based on links, sentiment-based community detection can help brands in aspects like segmenting markets [137].

6. **Detection of Abusive accounts on social networking websites:** Heavily negative sentiments gauged from content on social networks can help in detecting and flagging abusive language and the users who engage in it [26].
7. **Applications in IoT:** Internet of Things (IoT) includes smart homes which use devices that are interconnected and manage the systems in the house, such as cooling, heating, and other equipment. An application of sentiment analysis is detecting a person's emotions to accordingly customize the ambience [26].
8. **Chatbots:** Chatbots are often used to interact with consumers online with respect to some service, and automated sentiment analysis can be used to determine the consumer's satisfaction with the chatbot [138]. Improving consumer experience by tailoring the replies by the chatbot to be more consumer-centric based on sentiment analysis is another research direction [139].
9. **Recommendation Systems:** Opinion mining and sentiment analysis have been leveraged to build recommender systems by using the information obtained by grouping users based on their sentiment polarity, and also learning users' preferences with aspect extraction from their reviews to determine the interest of users in the available target products [5, 140, 141]. Sentiment prediction can help in gauging negativity expressed in the feedback, and based on that decision can be made on whether to recommend a product or not [26].
10. **Finding Experts/Influencers:** Sentiment polarity has been used to determine the social influence of social network users [142] and sentimental elements have been used for emotion identification to determine influencers on specific topics on social networks, including opinion leaders, trolls, and controversy makers [143]. As part of determining experts on virtual communities, sentiment mining has been used to detect the sentiment of the members' blogs and the comments on the blogs [144].
11. **Opinion Summarization:** Sentiment analysis can be used for summarizing or aggregating the opinions of consumers on products expressed on social media platforms [145]. The SumView is an example of a tool developed for getting a summary of online reviews and opinions of users on product features [146].
12. **Stance Detection:** Sentiment lexicon features have been used in the stance detection systems for social network posts, in which it is determined if a user opposes, favours, or neutral with respect to an entity [147, 148].
13. **Emotion Detection:** Some other applications of sentiment analysis include detecting emotions in suicide letters, which can be used for forensics and prediction of whether the text indicates a suicide or not [27, 149, 150].
14. **Applications in Health Care:** Twitter sentiment analysis has been used for aiding authorities to track health concerns among the public [151] and spreading of diseases or epidemics [152]. Using sentiment analysis can help in predicting suicidal sentiments or depression thoughts expressed on social media [6]. Emotion analysis of tweets can be used to determine the potential risks of depression for users and can be leveraged for suicide prevention [153].
15. **Application of Sentiment Analysis for Relief during Disasters:** Understanding sentiments in online posts during disasters helps determining public's feelings

and issues, and detect any panic. This situational awareness helps officials in making better decisions for handling disasters [154].

16. **Politics and Government Intelligence:** Sentiment detection empowers the analysis of political mood, or possibly predict election results. It also empowers the government to monitor people's opinions about policies [17]. Sentiment gauged through microblog posts is correlated to opinion derived from polls [155].
17. **Machine Translation:** Sentiment analysis is highly useful in enhancing the quality of machineSentiment analysis is highly useful in enhancing the quality of machine translation by identifying the implicit emotions of the phrases, and sarcasm in the text, as the interpreted meaning might be different [156].
18. **Other Applications in Different Domains:** Sentiment analysis can also be used for clustering of online forums and detecting hotspots [157], and for domains like sports and medicine by tracking emotion trends on social networks [26].

12.9 Conclusion

We have discussed the research methodology followed for sentiment analysis in terms of data collection, data pre-processing, feature extraction, and feature selection. Next, we explained how sentiment analysis results can be further enhanced using techniques, such as ontology, data integration, text categorization, crowdsourcing, and user characteristics. We have also discussed the metrics that are used to assess the quality of sentiment classification. Multiple tools have been created in the past research for aiding the sentiment analysis process, and these have been covered in the chapter. Finally, we discussed various sentiment analysis challenges, applications, and state-of-the-art literature in these directions.

12.10 Further Reading

There are some other surveys and review papers focusing on specific aspects of sentiment analysis and can be looked at further for more details. References [12, 13, 158–164] are some other surveys on sentiment analysis. References [12, 165, 166] explain the sentiment analysis applications in detail. References [27, 167–179] specifically cover the application of sentiment analysis in opinion mining. References [16, 180–185] discuss the sentiment analysis techniques for analyzing the user opinion in product reviews. Readers can look at [159, 186] for more details regarding the challenges for sentiment analysis. References [20, 26, 187–189] surveys focus on sentiment analysis on Twitter datasets. Asghar et al. [190] present a survey on techniques to analyze users' opinion on a YouTube video.

Text Mining is highly related to sentiment analysis, and readers can go through [191–195] for a better understanding. Erion and Morisio [25] presents a systematic review on word embedding and the impact of several other factors. You [196] covers related works to the sentiment analysis on multimedia datasets. References [197–199] review literature on aspect-level sentiment analysis, its challenges, and open problems. References [200–203] review the computational approaches to model negation in sentiment analysis, its limits, and challenges.

Most of the works we discussed in this chapter focused on sentiment analysis in English text, though there are some works that have focused on other languages. References [204–207] provided a survey on Arabic sentiment analysis and address the open challenges for future studies. Korayem et al. [208] surveyed the sentiment analysis techniques for other languages, including Chinese, Spanish, German, French, Urdu, and Arabic languages. Readers can further check [209–221] for sentiment analysis techniques in other languages.

References

1. Mäntylä, M.V., Graziotin, D., Kuutila, M.: The evolution of sentiment analysis- a review of research topics, venues, and top cited papers. Comput. Sci. Rev. **27**, 16–32 (2018)
2. Rambocas, M., Pacheco, B.G.: Online sentiment analysis in marketing research: a review. J. Res. Interact. Mark (2018)
3. Nassirtoussi, A.K., Aghabozorgi, S., Wah, T.Y., Ngo, D.C.L.: Text mining of news-headlines for forex market prediction: a multi-layer dimension reduction algorithm with semantics and sentiment. Expert. Syst. Appl. **42**(1), 306–324 (2015)
4. Li, X., Xie, H., Chen, L., Wang, J., Deng, X.: News impact on stock price return via sentiment analysis. Knowl.-Based Syst. **69**, 14–23 (2014)
5. Musto, C., de Gemmis, M., Semeraro, G., Lops, P.: A multi-criteria recommender system exploiting aspect-based sentiment analysis of users' reviews. In: Proceedings of the Eleventh ACM Conference on Recommender Systems, pp. 321–325 (2017)
6. Birjali, M., Beni-Hssane, A., Erritali, M.: Machine learning and semantic sentiment analysis based algorithms for suicide sentiment prediction in social networks. Procedia Comput. Sci. **113**, 65–72 (2017)
7. Clavel, C., Callejas, Z.: Sentiment analysis: from opinion mining to human-agent interaction. IEEE Trans. Affect. Comput. **7**(1), 74–93 (2015)
8. Mao, H., Bollen, J., Pepe, A.: Modeling public mood and emotion: twitter sentiment and socio-economic phenomena. In: Fifth International AAAI Conference on Weblogs and Social Media (2011)
9. Oberlander, J., Nowson, S.: Whose thumb is it anyway? Classifying author personality from weblog text. In: Proceedings of the COLING/ACL Main Conference Poster Sessions, pp. 627–634 (2006)
10. Rodrigues, R.G., et al.: Sentihealth-cancer: a sentiment analysis tool to help detecting mood of patients in online social networks. Int. J. Med. Inform. **85**(1), 80–95 (2016)
11. Tan, C. et al.: User-level sentiment analysis incorporating social networks. In: Proceedings of the 17th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, pp. 1397–1405 (2011)
12. Medhat, W., Hassan, A., Korashy, H.: Sentiment analysis algorithms and applications: a survey. Ain Shams Eng. J. **5**(4), 1093–1113 (2014)
13. Ahmed, K., El Tazi, N., Hossny, A.H.: Sentiment analysis over social networks: an overview. In: 2015 IEEE International Conference on Systems, Man, and Cybernetics, pp. 2174–2179. IEEE (2015)
14. Go, A., Bhayani, R., Huang, L.: Twitter sentiment classification using distant supervision. CS224N Project Report Stanford **1**(12), 2009 (2009)

15. Villena Román, J., Lana Serrano, S., Martínez Cámara, E., González Cristóbal, J.C.: Tass-workshop on sentiment analysis at sepln (2013)
16. Joshi, N.S., Itkat, S.A.: A survey on feature level sentiment analysis. Int. J. Comput. Sci. Inf. Technol. **5**(4), 5422–5425 (2014)
17. Kumar, A., Teeja, M.S.: Sentiment analysis: a perspective on its past, present and future. Int. J. Intell. Syst. Appl. **4**(10), 1 (2012)
18. Govindarajan, M., Romina, M.: A survey of classification methods and applications for sentiment analysis. Int. J. Eng. Sci. (IJES) **2**(12), 11–15 (2013)
19. Khan, F.H., Bashir, S., Qamar, U.: TOM: twitter opinion mining framework using hybrid classification scheme. Decis. Support. Syst. **57**, 245–257 (2014)
20. Desai, M., Mehta, M.A.: Techniques for sentiment analysis of twitter data: a comprehensive survey. In: 2016 International Conference on Computing, Communication and Automation (ICCCA), pp. 149–154. IEEE (2016)
21. Javadoc. http://twitter4j.org/en/javadoc.html. Accessed 22 Dec 2020
22. The opennlp project. https://developers.facebook.com/docs/graph-api/overview. Accessed 22 Dec 2020
23. Abbasi, A., Chen, H., Salem, A.: Sentiment analysis in multiple languages: feature selection for opinion classification in web forums. ACM Trans. Inf. Syst. (TOIS) **26**(3), 1–34 (2008)
24. Singh, T., Kumari, M.: Role of text pre-processing in twitter sentiment analysis. Procedia Comput. Sci. 89(Supplement C), 549–554 (2016)
25. Erion, Ç, Morisio, M: Word embeddings for sentiment analysis: a comprehensive empirical survey (2019). arXiv:1902.00753
26. Kharde, V., Sonawane, P., et al. Sentiment analysis of twitter data: a survey of techniques (2016). arXiv:1601.06971
27. Ravi, K., Ravi, V.: A survey on opinion mining and sentiment analysis: tasks, approaches and applications. Knowl.-Based Syst. **89**, 14–46 (2015)
28. Pang, B., Lee, L., Vaithyanathan, S.: Thumbs up? Sentiment classification using machine learning techniques. In: Proceedings of the 2002 Conference on Empirical Methods in Natural Language Processing (EMNLP 2002), pp. 79–86. Association for Computational Linguistics (2002)
29. Mikolov, T., Chen, K., Corrado, G., Dean, J.: Efficient estimation of word representations in vector space (2013). arXiv:1301.3781
30. Gimpel, K., Schneider, N., O'Connor, B., Das, D., Mills, D., Eisenstein, J., Heilman, M., Yogatama, D., Flanigan, J., Smith, N.A.: Part-of-speech tagging for twitter: annotation, features, and experiments. Technical report, Carnegie-Mellon Univ Pittsburgh Pa School of Computer Science (2010)
31. Law, M.H.C., Figueiredo, M.A.T., Jain, A.K.: Simultaneous feature selection and clustering using mixture models. IEEE Trans. Pattern Anal. Mach. Intell. **26**(9), 1154–1166 (2004)
32. Chandrashekar, G., Sahin, F.: A survey on feature selection methods. Comput. Electr. Eng. **40**(1), 16–28 (2014)
33. Haddi, E., Liu, X., Shi, Y.: The role of text pre-processing in sentiment analysis. Procedia Comput. Sci. **17**, 26–32 (2013)
34. Pudil, P., Novovičová, J., Kittler, J.: Floating search methods in feature selection. Pattern Recognit. Lett. **15**(11), 1119–1125 (1994)
35. Reunanen, J.: Overfitting in making comparisons between variable selection methods. J. Mach. Learn. Res. **3**(Mar), 1371–1382 (2003)
36. Goldenberg, D.E.: Genetic algorithms in search, optimization and machine learning (1989)
37. Ya-Han, H., Chen, Y.-L., Chou, H.-L.: Opinion mining from online hotel reviews-a text summarization approach. Inf. Process. Manag. **53**(2), 436–449 (2017)
38. Shetty, A., Bajaj, R.: Auto text summarization with categorization and sentiment analysis. Int. J. Comput. Appl. **130**(7), 57–60 (2015)
39. Pang, B., Lee, L.: A sentimental education: sentiment analysis using subjectivity summarization based on minimum cuts (2004). arXiv:cs/0409058

40. Bhargava, R., Sharma, Y.: MSATS: multilingual sentiment analysis via text summarization. In: 2017 7th International Conference on Cloud Computing, Data Science & Engineering-Confluence, pp. 71–76. IEEE (2017)
41. Cambria, E., Hussain, A.: Sentic computing. Marketing **59**(2), 557–577 (2012)
42. Kontopoulos, E., Berberidis, C., Dergiades, T., Bassiliades, N.: Ontology-based sentiment analysis of twitter posts. Expert. Syst. Appl. **40**(10), 4065–4074 (2013)
43. Cambria, E., Speer, R., Havasi, C., Hussain, A.: Senticnet: a publicly available semantic resource for opinion mining. In: 2010 AAAI Fall Symposium Series (2010)
44. Dragoni, M., Poria, S., Cambria, E.: Ontosenticnet: a commonsense ontology for sentiment analysis. IEEE Intell. Syst. **33**(3), 77–85 (2018)
45. Miller, G.A.: WordNet: An Electronic Lexical Database. MIT Press (1998)
46. Dragoni, M., Tettamanzi, A.G.B., da Costa Pereira, C.: Propagating and aggregating fuzzy polarities for concept-level sentiment analysis. Cogn. Comput. **7**(2), 186–197 (2015)
47. Cho, H., Kim, S., Lee, J., Lee, J.-S.: Data-driven integration of multiple sentiment dictionaries for lexicon-based sentiment classification of product reviews. Knowl.-Based Syst. **71**, 61–71 (2014)
48. Cambria, E., Havasi, C., Hussain, A.: Senticnet 2: a semantic and affective resource for opinion mining and sentiment analysis. In: Twenty-Fifth International FLAIRS Conference (2012)
49. Baccianella, S., Esuli, A., Sebastiani, F.: Sentiwordnet 3.0: an enhanced lexical resource for sentiment analysis and opinion mining. In: Lrec, vol. 10, pp. 2200–2204 (2010)
50. Stone, P.J., Hunt, E.B.: A computer approach to content analysis: studies using the general inquirer system. In: Proceedings of the May 21-23, 1963, Spring Joint Computer Conference, pp. 241–256 (1963)
51. Ji, X.: Social data integration and analytics for health intelligence. In: Proceedings VLDB PhD Workshop (2014)
52. Machedon, R., Rand, W., Joshi, Y.: Automatic crowdsourcing-based classification of marketing messaging on twitter. In: 2013 International Conference on Social Computing, pp. 975–978. IEEE (2013)
53. Borromeo, R.M., Toyama, M.: Automatic vs. crowdsourced sentiment analysis. In: Proceedings of the 19th International Database Engineering & Applications Symposium, pp. 90–95 (2015)
54. Kim, H.W., Zheng, J.R., Gupta, S.: Examining knowledge contribution from the perspective of an online identity in blogging communities. Comput. Hum. Behav. **27**(5), 1760–1770 (2011)
55. Guo, D., Chen, C.: Detecting non-personal and spam users on geo-tagged twitter network. Trans. GIS **18**(3), 370–384 (2014)
56. Lv, Y., Liu, J., Chen, H., Mi, J., Liu, M., Zheng, Q.: Opinioned post detection in sina weibo. IEEE Access **5**, 7263–7271 (2017)
57. Jiang, D., Luo, X., Xuan, J., Zheng, X.: Sentiment computing for the news event based on the social media big data. IEEE Access **5**, 2373–2382 (2016)
58. Crammer, K., Singer, Y.: Pranking with ranking. In: Advances in Neural Information Processing Systems, pp. 641–647 (2002)
59. Moghaddam, S., Ester, M.: Opinion digger: an unsupervised opinion miner from unstructured product reviews. In: Proceedings of the 19th ACM International Conference on Information and Knowledge Management, pp. 1825–1828 (2010)
60. Marcheggiani, D., Täckström, O., Esuli, A., Sebastiani, F.: Hierarchical multi-label conditional random fields for aspect-oriented opinion mining. In: European Conference on Information Retrieval, pp. 273–285. Springer (2014)
61. Lu, B., Ott, M., Cardie, C., Tsou, B.K.: Multi-aspect sentiment analysis with topic models. In: 2011 IEEE 11th International Conference on Data Mining Workshops, pp. 81–88. IEEE (2011)
62. Wang, H., Lu, Y., Zhai, C.X.: Latent aspect rating analysis without aspect keyword supervision. In: Proceedings of the 17th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, pp. 618–626 (2011)

63. Yang, D., Zhang, D., Yu, Z., Wang, Z.: A sentiment-enhanced personalized location recommendation system. In: Proceedings of the 24th ACM Conference on Hypertext and Social Media, pp. 119–128 (2013)
64. Järvelin, K., Kekäläinen, J.: Cumulated gain-based evaluation of IR techniques. ACM Trans. Inf. Syst. (TOIS) **20**(4), 422–446 (2002)
65. Turian, J.: Using alchemyapi for enterprise-grade text analysis. In: Denver, CO, USA, AlchemyAPI (2013)
66. Zhao, N., Jin, P., Wang, L., Yang, X., Liu, R., Zhang, W., Sui, K., Pei, D.: Automatically and adaptively identifying severe alerts for online service systems
67. Cunningham, H.: Gate, a general architecture for text engineering. Comput. Humanit. **36**(2), 223–254 (2002)
68. TIPSTER Architecture Committee et al. Tipster text phase ii architecture concept. Technical report, TIPSTER working paper (1994). http://www.cs.nyu.edu/tipster
69. Araújo, M., Gonçalves, P., Cha, M., Benevenuto, F.: ifeel: a system that compares and combines sentiment analysis methods. In: Proceedings of the 23rd International Conference on World Wide Web, pp. 75–78 (2014)
70. A sentiment analysis framework. http://www.ifeel.dcc.ufmg.br/. Accessed 22 Dec 2020)
71. Carter, S., Chen, F., Muralidharan, A.S., Pickens, J.: Dig: a task-based approach to product search. In: Proceedings of the 16th International Conference on Intelligent User Interfaces, pp. 303–306 (2011)
72. Carpenter, B.: Lingpipe for 99.99% recall of gene mentions. In: Proceedings of the Second BioCreative Challenge Evaluation Workshop, vol. 23, pp. 307–309. BioCreative (2007)
73. Eryiğit, G., Cetin, F.S., Yanık, M., Temel, T., Ciçekli, I.: Turksent: a sentiment annotation tool for social media. In: Proceedings of the 7th Linguistic Annotation Workshop and Interoperability with Discourse, pp. 131–134 (2013)
74. Bird, S. et al.: Multidisciplinary instruction with the natural language toolkit. In: Association for Computational Linguistics (2008)
75. Tomanek, K., Buyko, E., Hahn, U.: An uima-based tool suite for semantic text processing. In: UIMA Workshop at the GLDV, vol. 11 (2007)
76. Liu, B., Hu, M., Cheng, J.: Opinion observer: analyzing and comparing opinions on the web. In: Proceedings of the 14th International Conference on World Wide Web, pp. 342–351 (2005)
77. Kotelevskaya, V.: Evaluating and Improving An Automatic Sentiment Analysis System. Uppsala University, Uppsala, Sweden (2011)
78. Baldridge, J.: The opennlp project, p. 1, (2005). http://opennlp.apache.org/index.html. Accessed 27 July 2020
79. Wilson, T., Hoffmann, P., Somasundaran, S., Kessler, J., Wiebe, J., Choi, Y., Cardie, C., Riloff, E., Patwardhan, S.: Opinionfinder: a system for subjectivity analysis. In: Proceedings of HLT/EMNLP 2005 Interactive Demonstrations, pp. 34–35 (2005)
80. Fortuna, B., Grobelnik, M., Mladenic, D.: Ontogen: semi-automatic ontology editor. In: Symposium on Human Interface and the Management of Information, pp. 309–318. Springer (2007)
81. Mostafa, M.M.: More than words: social networks' text mining for consumer brand sentiments. Expert. Syst. Appl. **40**(10), 4241–4251 (2013)
82. Lewis, R.B., Maas, S.M.: QDA miner 2.0: mixed-model qualitative data analysis software. Field Methods **19**(1), 87–108 (2007)
83. Free qualitative data analysis software: QDA miner lite. https://provalisresearch.com/products/qualitative-data-analysis-software/freeware/. Accessed 22 Dec 2020
84. Al-Kabi, M., Al-Qudah, N.M., Alsmadi, I., Dabour, M., Wahsheh, H.: Arabic/english sentiment analysis: an empirical study. In: The Fourth International Conference on Information and Communication Systems (ICICS 2013), pp. 23–25 (2013)
85. Go, M.P., Nocon, N.: Using Stanford part-of-speech tagger for the morphologically-rich Filipino language. In: Proceedings of the 31st Pacific Asia Conference on Language, Information and Computation, pp. 81–88 (2017)

86. Bajwa, I.S, Lee, M., Bordbar, B.: Resolving syntactic ambiguities in natural language specification of constraints. In: International Conference on Intelligent Text Processing and Computational Linguistics, pp. 178–187. Springer (2012)
87. Spinczyk, D., Nabrdalik, K., Rojewska, K.: Computer aided sentiment analysis of anorexia nervosa patients' vocabulary. Biomed. Eng. Online **17**(1), 19 (2018)
88. Sarkar, M.K., Chakraborty, G.: Opinion mining and geo-positioning of textual feedback from professional drivers. In: Proceedings of SAS Global Forum Data Mining Text Analaysis, pp. 1–11. Citeseer (2013)
89. Barbosa, L., Feng, J.: Robust sentiment detection on twitter from biased and noisy data. In: Coling 2010: Posters, pp. 36–44 (2010)
90. Kouloumpis, E., Wilson, T., Moore, J.: Twitter sentiment analysis: the good the bad and the omg! In: Fifth International AAAI Conference on Weblogs and Social Media (2011)
91. Cadilhac, A., Benamara, F., Aussenac-Gilles, N.: Ontolexical resources for feature-based opinion mining: a case-study. In: Proceedings of the 6th Workshop on Ontologies and Lexical Resources, pp. 77–86 (2010)
92. Cui, A., Zhang, M., Liu, Y., Ma, S.: Emotion tokens: bridging the gap among multilingual twitter sentiment analysis. In: Asia Information Retrieval Symposium, pp. 238–249. Springer (2011)
93. Shaik, A., Dey, N.S., Purnachand, K., Babu, C.M.: Sentiment extraction and analysis using machine learning tools-survey. In: IOP Conference Series: Materials Science and Engineering, vol. 594, p. 012022. IOP Publishing (2019)
94. Taddy, M.A.: Inverse regression for analysis of sentiment in text (2010). arXiv:1012
95. Anta, A.F., Chiroque, L.N., Morere, P., Santos, A.: Sentiment analysis and topic detection of Spanish tweets: a comparative study of of NLP techniques. Procesamiento del Lenguaje Nat. **50**, 45–52 (2013)
96. Bharti, S.K., Babu, K.S., Pradhan, A., Devi, S., Priya, T.E., Orhorhoro, E., Orhorhoro, O., Atumah, V., Baruah, E., Konwar, P., et al.: Automatic keyword extraction for text summarization in multi-document e-newspapers articles. Eur. J. Adv. Eng. Technol. **4**(6), 410–427 (2017)
97. Campos, R., Mangaravite, V., Pasquali, A., Jorge, A.M., Nunes, C., Jatowt, A.: A text feature based automatic keyword extraction method for single documents. In: European Conference on Information Retrieval, pp. 684–691. Springer (2018)
98. Biswas, S.K.: Keyword extraction from tweets using weighted graph. In: Cognitive Informatics and Soft Computing, pp. 475–483. Springer (2019)
99. Bhadane, C., Dalal, H., Doshi, H.: Sentiment analysis: measuring opinions. Procedia Comput. Sci. **45**, 808–814 (2015)
100. Jeyapriya, A., Kanimozhi Selvi, C.S.: Extracting aspects and mining opinions in product reviews using supervised learning algorithm. In: 2015 2nd International Conference on Electronics and Communication Systems (ICECS), pp. 548–552. IEEE (2015)
101. Jeevanandam Jotheeswaran, D.R., Kumaraswamy, Y.S.: Opinion mining using decision tree based feature selection through Manhattan hierarchical cluster measure. J. Theor. Appl. Inf. Technol. **58**(1), 72–80 (2013)
102. Varathan, K.D., Giachanou, A., Crestani, F.: Comparative opinion mining: a review. J. Assoc. Inf. Sci. Technol. **68**(4), 811–829 (2017)
103. Ravichandran, M., Kulanthaivel, G.: Twitter sentiment mining (TSM) framework based learners emotional state classification and visualization for e-learning system. J. Theor. Appl. Inf. Technol. **69**(1) (2014)
104. Matsumoto, K., Ren, F., Matsuoka, M., Yoshida, M., Kita, K.: Slang feature extraction by analysing topic change on social media. CAAI Trans. Intell. Technol. **4**(1), 64–71 (2019)
105. Chen, C., Zhang, J., Chen, X., Xiang, Y., Zhou, W.: 6 million spam tweets: a large ground truth for timely twitter spam detection. In: 2015 IEEE International Conference on Communications (ICC), pp. 7065–7070. IEEE (2015)
106. Martinez-Romo, J., Araujo, L.: Detecting malicious tweets in trending topics using a statistical analysis of language. Expert. Syst. Appl. **40**(8), 2992–3000 (2013)

107. Sedhai, S., Sun, A.: Semi-supervised spam detection in twitter stream. IEEE Trans. Comput. Soc. Syst. **5**(1), 169–175 (2017)
108. Pang, B., Lee, L.: Opinion mining and sentiment analysis. Found. Trends Inf. Retr. **2**(1–2), 1–135 (2008)
109. Miller, Z., Dickinson, B., Hu, W.: Gender prediction on twitter using stream algorithms with N-gram character features (2012)
110. You, Q., Bhatia, S., Sun, T., Luo, J.: The eyes of the beholder: gender prediction using images posted in online social networks. In: 2014 IEEE International Conference on Data Mining Workshop, pp. 1026–1030. IEEE (2014)
111. Majumder, N., Poria, S., Gelbukh, A., Cambria, E.: Deep learning-based document modeling for personality detection from text. IEEE Intell. Syst. **32**(2), 74–79 (2017)
112. Xue, D., Lifa, W., Hong, Z., Guo, S., Gao, L., Zhiyong, W., Zhong, X., Sun, J.: Deep learning-based personality recognition from text posts of online social networks. Appl. Intell. **48**(11), 4232–4246 (2018)
113. Galal, A., El-Korany, A.: Enabling semantic user context to enhance twitter location prediction. ICAART **1**, 223–230 (2016)
114. Li, C., Bai, J., Zhang, L., Tang, H., Luo, Y.: Opinion community detection and opinion leader detection based on text information and network topology in cloud environment. Inf. Sci. **504**, 61–83 (2019)
115. Gargi, U., Lu, W., Mirrokni, V., Yoon, S.: Large-scale community detection on youtube for topic discovery and exploration. In: Fifth International AAAI Conference on Weblogs and Social Media (2011)
116. Jedrzejewski, K., Morzy, M.: Opinion mining and social networks: a promising match. In: 2011 International Conference on Advances in Social Networks Analysis and Mining, pp. 599–604. IEEE (2011)
117. Farooq, U., Mansoor, H., Nongaillard, A., Ouzrout, Y., Qadir, M.A.: Negation handling in sentiment analysis at sentence level. JCP **12**(5), 470–478 (2017)
118. Krebs, F., Lubascher, B., Moers, T., Schaap, P., Spanakis, G.: Social emotion mining techniques for facebook posts reaction prediction (2017). arXiv:1712.03249
119. Amalia, R., Bijaksana, M.A., Darmantoro, D.: Negation handling in sentiment classification using rule-based adapted from Indonesian language syntactic for Indonesian text in twitter. In: Journal of Physics: Conference Series (International Conference on Data Information Science), vol. 971 (2017)
120. Jain, T.I., Nemade, D.: Recognizing contextual polarity in phrase-level sentiment analysis. Int. J. Comput. Appl. **7**(5), 12–21 (2010)
121. González-Ibánez, R., Muresan, S., Wacholder, N.: Identifying sarcasm in twitter: a closer look. In: Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies, pp. 581–586 (2011)
122. Maynard, D.G., Greenwood, M.A.: Who cares about sarcastic tweets? investigating the impact of sarcasm on sentiment analysis. In: LREC 2014 Proceedings. ELRA, 2014
123. Bamman, D., Smith, N.A.: Contextualized sarcasm detection on twitter. In: Ninth International AAAI Conference on Web and Social Media (2015)
124. Mukherjee, S.: Sentiment Analysis of Reviews, pp. 1–10. Springer, New York (2017)
125. Bollen, J., Mao, H., Zeng, X.: Twitter mood predicts the stock market. J. Comput. Sci. **2**(1), 1–8 (2011)
126. Qiu, L., Rui, H., Whinston, A.: Social network-embedded prediction markets: the effects of information acquisition and communication on predictions. Decis. Support. Syst. **55**(4), 978–987 (2013)
127. Li, Y.-M., Li, T.-Y.: Deriving market intelligence from microblogs. Decis. Support. Syst. **55**(1), 206–217 (2013)
128. Castillo, C., Mendoza, M., Poblete, B.: Predicting information credibility in time-sensitive social media. Internet Res. **23**(5), 560–588 (2013)
129. Gupta, A., Kumaraguru, P.: Credibility ranking of tweets during high impact events. In: Proceedings of the 1st Workshop on Privacy and Security in Online Social Media, pp. 2–8 (2012)

130. Yoo, K.-H., Gretzel, U.: Comparison of deceptive and truthful travel reviews. Inf. Commun. Technol. Tour. **2009**, 37–47 (2009)
131. Kwon, S., Cha, M., Jung, K.: Rumor detection over varying time windows. PloS One **12**(1) (2017)
132. Saxena, A., Hsu, W., Lee, M.L., Leong Chieu, H., Ng, L., Teow, L.N.: Mitigating misinformation in online social network with top-k debunkers and evolving user opinions. Companion Proc. Web Conf. **2020**, 363–370 (2020)
133. Dickerson, J.P., Kagan, V., Subrahmanian, V.S.: Using sentiment to detect bots on twitter: are humans more opinionated than bots? In: 2014 IEEE/ACM International Conference on Advances in Social Networks Analysis and Mining (ASONAM 2014), pp. 620–627. IEEE (2014)
134. Peng, Q., Zhong, M.: Detecting spam review through sentiment analysis. JSW **9**(8), 2065–2072 (2014)
135. Choo, E., Yu, T., Chi, M.: Detecting opinion spammer groups through community discovery and sentiment analysis. In: IFIP Annual Conference on Data and Applications Security and Privacy, pp. 170–187. Springer (2015)
136. Deitrick, W., Hu, W.: Mutually enhancing community detection and sentiment analysis on twitter networks (2013)
137. Xu, K., Li, J., Liao, S.S.: Sentiment community detection in social networks. In: Proceedings of the 2011 iConference, pp. 804–805 (2011)
138. Feine, J., Morana, S., Gnewuch, U.: Measuring service encounter satisfaction with customer service chatbots using sentiment analysis (2019)
139. Schreuder, A., Schreuder, A., van Wyk, J.: Customer centric artificial intelligence-using text and sentiment analysis & deep neural network learning to make chatbots reply in a more customer centric fashion (2017)
140. Li, Y.-M., Shiu, Y.-L.: A diffusion mechanism for social advertising over microblogs. Decis. Support. Syst. **54**(1), 9–22 (2012)
141. García-Cumbreras, M., Montejo-Ráez, A., Díaz-Galiano, M.C.: Pessimists and optimists: Improving collaborative filtering through sentiment analysis. Expert. Syst. Appl. **40**(17), 6758–6765 (2013)
142. Lahuerta-Otero, E. Cordero-Gutiérrez, R.: Looking for the perfect tweet. The use of data mining techniques to find influencers on twitter. Comput. Hum. Behav. **64**, 575–583 (2016)
143. Gao, L., Yue, W., Xiong, X., Tang, J.: Discriminating topical influencers based on the user relative emotion. IEEE Access **7**, 100120–100130 (2019)
144. Kumar, A., Ahmad, N.: Comex miner: expert mining in virtual communities. Int. J. Adv. Comput. Sci. Appl. (IJACSA) **3**(6) (2012)
145. Soleymani, M., Garcia, D., Jou, B., Schuller, B., Chang, S.-F., Pantic, M.: A survey of multimodal sentiment analysis. Image Vis. Comput. **65**, 3–14 (2017)
146. Wang, D., Zhu, S., Li, T.: Sumview: A web-based engine for summarizing product reviews and customer opinions. Expert. Syst. Appl. **40**(1), 27–33 (2013)
147. Mohammad, S., Kiritchenko, S., Sobhani, P., Zhu, X., Cherry, C.: Semeval-2016 task 6: detecting stance in tweets. In: Proceedings of the 10th International Workshop on Semantic Evaluation (SemEval-2016), pp. 31–41 (2016)
148. Aldayel, A., Magdy, W.: Assessing sentiment of the expressed stance on social media. In: International Conference on Social Informatics, pp. 277–286. Springer (2019)
149. Desmet, B., Hoste, V.R.: Emotion detection in suicide notes. Expert. Syst. Appl. **40**(16), 6351–6358 (2013)
150. Mohammad, S.M.: From once upon a time to happily ever after: tracking emotions in mail and books. Decis. Support. Syst. **53**(4), 730–741 (2012)
151. Ji, X., Chun, S.A., Wei, Z., Geller, J.: Twitter sentiment classification for measuring public health concerns. Soc. Netw. Anal. Min. **5**(1), 13 (2015)
152. Ji, X., Chun, S.A., Geller, J.: Monitoring public health concerns using twitter sentiment classifications. In: 2013 IEEE International Conference on Healthcare Informatics, pp. 335–344. IEEE (2013)

153. De Choudhury, M., Gamon, M., Counts, S., Horvitz, E.: Predicting depression via social media. In: Seventh International AAAI Conference on Weblogs and Social Media (2013)
154. Beigi, G., Hu, X., Maciejewski, R., Liu, H.: An overview of sentiment analysis in social media and its applications in disaster relief. In: Sentiment Analysis and Ontology Engineering, pp. 313–340. Springer (2016)
155. O'Connor, B., Balasubramanyan, R., Routledge, B.R., Smith, N.A.: From tweets to polls: linking text sentiment to public opinion time series. In: Fourth International AAAI Conference on Weblogs and Social Media (2010)
156. Hossny, A., Shaalan, K., Fahmy, A.: Machine translation model using inductive logic programming. In: 2009 International Conference on Natural Language Processing and Knowledge Engineering, pp. 1–8. IEEE (2009)
157. Li, N., Wu, D.D.: Using text mining and sentiment analysis for online forums hotspot detection and forecast. Decis. Support. Syst. **48**(2), 354–368 (2010)
158. Hussein, D.M.E.D.M.: A survey on sentiment analysis challenges. J. King Saud Univ.-Eng. Sci. **30**(4), 330–338 (2018)
159. Mukherjee, S., Bhattacharyya, P.: Sentiment analysis: a literature survey (2013). arXiv:1304.4520
160. Routray, P., Swain, C.K., Mishra, S.P.: A survey on sentiment analysis. Int. J. Comput. Appl. **76**(10) (2013)
161. Kaur, H., Mangat, V., et al.: A survey of sentiment analysis techniques. In: 2017 International Conference on I-SMAC (IoT in Social, Mobile, Analytics and Cloud)(I-SMAC), pp. 921–925. IEEE (2017)
162. Abirami, A.M., Gayathri, V.: A survey on sentiment analysis methods and approach. In: 2016 Eighth International Conference on Advanced Computing (ICoAC), pp. 72–76. IEEE (2017)
163. Jain, S.K., Singh, P.: Systematic survey on sentiment analysis. In: 2018 First International Conference on Secure Cyber Computing and Communication (ICSCCC), pp. 561–565. IEEE (2018)
164. Yue, L., Chen, W., Li, X., Zuo, W., Yin, M.: A survey of sentiment analysis in social media. Knowl. Inf. Syst. **1–47** (2019)
165. Liu, R., Shi, Y., Ji, C., Jia, M.: A survey of sentiment analysis based on transfer learning. IEEE Access **7**, 85401–85412 (2019)
166. Rathan, M., Hulipalled, V.R., Murugeshwari, P., Sushmitha, H.M.: Every post matters: a survey on applications of sentiment analysis in social media. In: 2017 International Conference on Smart Technologies For Smart Nation (SmartTechCon), pp. 709–714. IEEE (2017)
167. Liu, B., Zhang, L.: A survey of opinion mining and sentiment analysis. In: Mining Text Data, pp. 415–463. Springer (2012)
168. Vinodhini, G., Chandrasekaran, R.M.: Sentiment analysis and opinion mining: a survey. Int. J. **2**(6), 282–292 (2012)
169. Kaur, A., Gupta, V.: A survey on sentiment analysis and opinion mining techniques. J. Emerg. Technol. Web Intell. **5**(4), 367–371 (2013)
170. Hemmatian, F., Sohrabi, M.K.: A survey on classification techniques for opinion mining and sentiment analysis. Artif. Intell. Rev. 1–51 (2019)
171. Varghese, R., Jayasree, M.: A survey on sentiment analysis and opinion mining. Int. J. Res. Eng. Technol. **2**(11), 312–317 (2013)
172. Pradhan, V.M., Vala, J., Balani, P.: A survey on sentiment analysis algorithms for opinion mining. Int. J. Comput. Appl. **133**(9), 7–11 (2016)
173. Li, Z., Fan, Y., Jiang, B., Lei, T., Liu, W.: A survey on sentiment analysis and opinion mining for social multimedia. Multimed. Tools Appl. **78**(6), 6939–6967 (2019)
174. Sadegh Hajmohammadi, M., Ibrahim, R., Othman, Z.A. : Opinion mining and sentiment analysis: a survey. Int. J. Comput. Technol. **2**(3), 171–178 (2012)
175. Padmaja, S., Fatima, S.S.: Opinion mining and sentiment analysis-an assessment of peoples' belief: a survey. Int. J. Ad Hoc, Sens. Ubiquitous Comput. **4**(1), 21 (2013)
176. Guellil, I., Boukhalfa, K.: Social big data mining: a survey focused on opinion mining and sentiments analysis. In: 2015 12th International Symposium on Programming and Systems (ISPS), pp. 1–10. IEEE (2015)

177. ChandraKala, S., Sindhu, C.: Opinion mining and sentiment classification: a survey. ICTACT J. Soft Comput. **3**(1), 420–425 (2012)
178. Liu, B.: Sentiment analysis and opinion mining. Synth. Lect. Hum. Lang. Technol. **5**(1), 1–167 (2012)
179. Patel, V., Prabhu, G., Bhowmick, K.: A survey of opinion mining and sentiment analysis. Int. J. Comput. Appl. **131**(1), 24–27 (2015)
180. Jebaseeli, A.N., Kirubakaran, E.: A survey on sentiment analysis of (product) reviews. Int. J. Comput. Appl. **47**(11) (2012)
181. Buche, A., Chandak, D., Zadgaonkar, A.: Opinion mining and analysis: a survey (2013). arXiv:1307.3336
182. Tsytsarau, M., Palpanas, T.: Survey on mining subjective data on the web. Min. Knowl. Discov. **24**(3), 478–514 (2012)
183. Khan, K., Baharudin, B.B., Khan, A., et al.: Mining opinion from text documents: a survey. In: 2009 3rd IEEE International Conference on Digital Ecosystems and Technologies, pp. 217–222. IEEE (2009)
184. Chaturvedi, I., Cambria, E., Welsch, R.E., Herrera, F.: Distinguishing between facts and opinions for sentiment analysis: survey and challenges. Inf. Fusion **44**, 65–77 (2018)
185. Ghode, M., Bere, S., Kamale, M., Moholkar, M.: Sentiment analysis over online product reviews: a survey. Int. J. Recent. Innov. Trends Comput. Commun. **2**(11), 3766–3774 (2014)
186. Pozzi, F.A., Fersini, E., Messina, E., Liu, B.: Challenges of sentiment analysis in social networks: an overview. In: Sentiment Analysis in Social Networks, pp. 1–11. Elsevier (2017)
187. Silva, N.F.F.D., Coletta, L.F.S., Hruschka, E.R.: A survey and comparative study of tweet sentiment analysis via semi-supervised learning. ACM Comput. Surv. (CSUR) **49**(1), 1–26 (2016)
188. Giachanou, A., Crestani, F.: Like it or not: a survey of twitter sentiment analysis methods. ACM Comput. Surv. (CSUR) **49**(2), 1–41 (2016)
189. Wagh, R., Punde, P.: Survey on sentiment analysis using twitter dataset. In: 2018 Second International Conference on Electronics, Communication and Aerospace Technology (ICECA), pp. 208–211. IEEE (2018)
190. Asghar, M.Z., Ahmad, S., Marwat, A., Kundi, F.M.: Sentiment analysis on youtube: a brief survey (2015). arXiv:1511.09142
191. Aggarwal, C.C., Zhai, C.: Mining Text Data. Springer Science & Business Media (2012)
192. Berry, M.W., Castellanos, M.: Survey of text mining. Comput. Rev. **45**(9), 548 (2004)
193. Gupta, V., Lehal, G.S., et al.: A survey of text mining techniques and applications. J. Emerg. Technol. Web Intell. **1**(1), 60–76 (2009)
194. Allahyari, M., Pouriyeh, S., Assefi, M., Safaei, S., Trippe, E.D., Gutierrez, J.B., Kochut, K.: A brief survey of text mining: classification, clustering and extraction techniques (2017). arXiv:1707.02919
195. Hotho, A., Nürnberger, A., Paaß, G.: A brief survey of text mining. In: Ldv Forum, vol. 20, pp. 19–62. Citeseer (2005)
196. You, Q.: Sentiment and emotion analysis for social multimedia: methodologies and applications. In: Proceedings of the 24th ACM International Conference on Multimedia, pp. 1445–1449 (2016)
197. Schouten, K., Frasincar, F.: Survey on aspect-level sentiment analysis. IEEE Trans. Knowl. Data Eng. **28**(3), 813–830 (2015)
198. Rana, T.A., Cheah, Y.-N.: Aspect extraction in sentiment analysis: comparative analysis and survey. Artif. Intell. Rev. **46**(4), 459–483 (2016)
199. Laskari, N.K., Sanampudi, S.K.: Aspect based sentiment analysis survey. IOSR J. Comput. Eng. (IOSR-JCE) **18**(2), 24–28 (2016)
200. Wiegand, M., Balahur, A., Roth, B., Klakow, D., Montoyo, A.: A survey on the role of negation in sentiment analysis. In: Proceedings of the Workshop on Negation and Speculation in Natural Language Processing, pp. 60–68 (2010)
201. Dadvar, M., Hauff, C., de Jong, F.: Scope of negation detection in sentiment analysis. In: Proceedings of the Dutch-Belgian Information Retrieval Workshop (DIR 2011), pp. 16–20. Citeseer (2011)

202. Asmi, A., Ishaya, T.: Negation identification and calculation in sentiment analysis. In: The Second International Conference on Advances in Information Mining and Management, pp. 1–7 (2012)
203. Lapponi, E., Read, J., Øvrelid, L.: Representing and resolving negation for sentiment analysis. In: 2012 IEEE 12th International Conference on Data Mining Workshops, pp. 687–692. IEEE (2012)
204. Al-Ayyoub, M., Khamaiseh, A.A., Jararweh, Y., Al-Kabi, M.N.: A comprehensive survey of arabic sentiment analysis. Inf. Process. Manag. **56**(2), 320–342 (2019)
205. Korayem, M., Crandall, D., Abdul-Mageed, M.: Subjectivity and sentiment analysis of arabic: a survey. In: International Conference on Advanced Machine Learning Technologies and Applications, pp. 128–139. Springer (2012)
206. Alowisheq, A., Alhumoud, S., Altwairesh, N., Albuhairi, T.: Arabic sentiment analysis resources: a survey. In: Social Computing and Social Media: 8th International Conference, SCSM 2016, Held as Part of HCI International 2016, Toronto, ON, Canada, July 17–22, 2016. Proceedings 8, pp. 267–278. Springer (2016)
207. Biltawi, M., Etaiwi, W., Tedmori, S., Hudaib, A., Awajan, A.: Sentiment classification techniques for arabic language: a survey. In: 2016 7th International Conference on Information and Communication Systems (ICICS), pp. 339–346. IEEE (2016)
208. Korayem, M., Aljadda, K., Crandall, D.: Sentiment/subjectivity analysis survey for languages other than english. Soc. Netw. Anal. Min. **6**(1), 75 (2016)
209. Syed, A.Z., Aslam, M., Martinez-Enriquez, A.M.: Sentiment analysis of urdu language: handling phrase-level negation. In: Mexican International Conference on Artificial Intelligence, pp. 382–393. Springer (2011)
210. Sharma, R., Nigam, S., Jain, R.: Opinion mining in hindi language: a survey (2014). arXiv:1404.4935
211. Wei, W., Xiang, Y., Chen, Q.: Survey on Chinese text sentiment analysis. Jisuanji Yingyong/J. Comput. Appl. **31**(12), 3321–3323 (2011)
212. Panigrahi, P.K., Bele, N.: A review of recent advances in text mining of Indian languages. Int. J. Bus. Inf. Syst. **23**(2), 175–193 (2016)
213. Zehe, A., Becker, M., Jannidis, F., Hotho, A.: Towards sentiment analysis on German literature. In: Joint German/Austrian Conference on Artificial Intelligence (Künstliche Intelligenz), pp. 387–394. Springer (2017)
214. Brooke, J., Tofiloski, M., Taboada, M.: Cross-linguistic sentiment analysis: From English to Spanish. In: Proceedings of the International Conference RANLP-2009, pp. 50–54 (2009)
215. Khattak, A., Asghar, M.Z., Saeed, A., Hameed, I.A., Hassan, S.A., Ahmad, S.: A survey on sentiment analysis in urdu: a resource-poor language. Egypt. Inform. J. (2020)
216. Pereira, D.A.: A survey of sentiment analysis in the Portuguese language. Artif. Intell. Rev. **1–29** (2020)
217. Kumar, S.S., Kumar, M.A., Soman, K.P.: Identifying sentiment of Malayalam tweets using deep learning. In: Digital Business, pp. 391–408. Springer (2019)
218. Vīksna, R., Jēkabsons, G.: Sentiment analysis in Latvian and Russian: a survey. Appl. Comput. Syst. **23**(1), 45–51 (2018)
219. Basile, V., Nissim, M.: Sentiment analysis on Italian tweets. In: Proceedings of the 4th Workshop on Computational Approaches to Subjectivity, Sentiment and Social Media Analysis, pp. 100–107 (2013)
220. Alwakid, G., Osman, T., Hughes-Roberts, T.: Challenges in sentiment analysis for Arabic social networks. Procedia Comput. Sci. **117**, 89–100 (2017)
221. Peng, H., Cambria, E., Hussain, A.: A review of sentiment analysis research in Chinese language. Cogn. Comput. **9**(4), 423–435 (2017)

Chapter 13
Recent Developments in Sentiment Analysis on Social Networks: Techniques, Datasets, and Open Issues

Akrati Saxena, Harita Reddy, and Pratishtha Saxena

Abstract In recent years, sentiment analysis has been highly used on social media datasets to get conclusive information, opinions of users about different topics, such as politics, events, and products, and predicting users' mood or emotions. Marketing companies are very interested in finding out people's opinions and consider that in designing marketing strategies. However, filtering and monitoring the sentiments remain a formidable task due to the proliferation of diverse websites. Researchers have proposed numerous methods for automating the process of filtering and summarizing users' sentiment and opinion on social media. In this chapter, we present a systematic review of sentiment analysis techniques, including machine learning-based, deep learning-based, lexicon-based, graph-based, and hybrid approaches. We further provide a concise list of openly available datasets that have been used for research studies. Finally, we cover future directions and the respective challenges that should be looked at for further enhancing the sentiment analysis for microblog datasets of social networks.

13.1 Introduction

Social media websites furnish a platform for building online social relationships and sharing news, opinions, as well as multimedia to a global audience. With a huge volume of posts containing opinions of people across the world on various topics, the content has been used for insights by both academic researchers and companies

A. Saxena (✉)
Department of Mathematics and Computer Science, Eindhoven University of Technology, Eindhoven, The Netherlands
e-mail: a.saxena@tue.nl

H. Reddy
Surat, Gujarat, India

P. Saxena
G. L. Bajaj Institute of Technology and Management, Greater Noida, India
e-mail: pratistha.saxena@glbitm.ac.in

A. Biswas et al. (eds.), *Principles of Social Networking*,
Smart Innovation, Systems and Technologies 246,
https://doi.org/10.1007/978-981-16-3398-0_13

for opinion mining [1]. Sentiment analysis involves analyzing people's sentiment, opinion, or attitudes toward a particular entity, which might include products, services, policies, or celebrities [2]. A common application of sentiment analysis in the industry is to plan marketing strategies based on public sentiment and gauge public sentiment in response to a new product launch. Researchers have shown great interest in proposing novel techniques for classifying the sentiment of online content.

The sentiment classification approaches in previous works can be classified into machine learning-based and lexicon-based methods. However, inadequate contextual information in small paragraphs and individual sentences could pose challenges despite these methods. The availability of abundant data and computational power has paved the way for exploring deep learning solutions that utilize prior information for sentiment analysis. Tasks requiring complex features for sentiment classification and contextual polarity disambiguation in tweets make use of deep convolutional neural networks [3, 4]. In this chapter, we will discuss all these categories of sentiment analysis techniques in depth. In a data-driven model, it becomes imperative to analyze how data is collected, perceived, and communicated. We will also cover various datasets that have been used for training and evaluating sentiment analysis models in the past research. Furthermore, there are multiple directions on which researchers can focus in the future, which we discuss toward the end of the chapter.

The chapter is structured in the following manner. In Sect. 13.2, we present a systematic review of sentiment analysis techniques, including machine learning-based techniques, lexicon-based techniques, hybrid techniques, graph-based techniques, and other techniques. In Sect. 13.3, we list the available datasets. In Sect. 13.4, we discuss the future directions of sentiment analysis in the context of large-scale data-rich social networks. The chapter is concluded in Sect. 13.5.

13.2 Sentiment Analysis Techniques

An important task in sentiment analysis is the selection of the technique for achieving high accuracy in classifying the sentiment of the content. There are three broad categories of techniques, i.e., machine learning (ML)-based, lexicon-based, and hybrid techniques, as shown in Fig. 13.1. Using these approaches, a given text document is assigned a suitable sentiment class: positive, negative, or neutral. ML techniques involve the usage of ML algorithms, for example, Naive Bayes Classifier, k-Nearest Neighbors Classifier, and Support Vector Machines (SVM), along with the linguistic features of the document. This class of techniques can be subdivided into supervised, weakly supervised, semi-supervised, and unsupervised learning techniques. For supervised learning, a dataset containing the text documents annotated with the ground truth sentiment labels is required for training the classification model before the model can identify the sentiment of new documents. Unsupervised learning is used when labeled documents are not available. Lexicon-based techniques involve the use of a sentiment dictionary or lexicon containing opinion words to identify the sentiment toward which a document is oriented. Examples of lexicon-based tech-

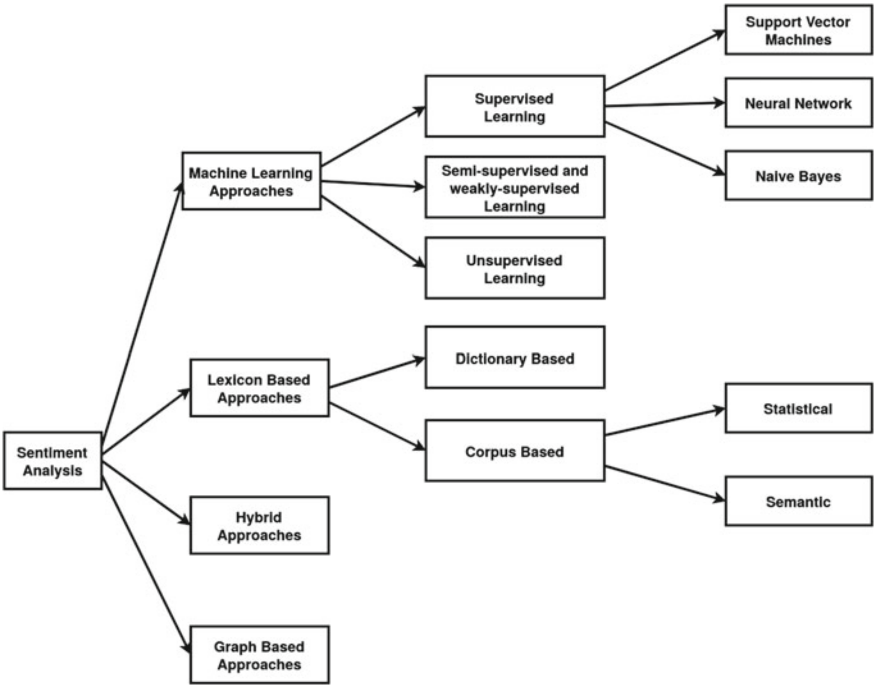


Fig. 13.1 Overview of Sentiment Analysis Techniques

niques are dictionary-based and corpus-based techniques. In the dictionary-based techniques, a pre-existing dictionary containing a sentiment word list is used to find the presence of such words within the text and accordingly determine the sentiment expressed in the document based on counting. The corpus-based approaches additionally rely on contextual information with statistical and semantic methods. Hybrid approaches are usually a fusion of the discussed techniques. Another class of techniques that we discuss is graph-based approaches. In the upcoming sections, we will discuss the sentiment analysis techniques in detail.

13.2.1 Machine Learning Approaches

Machine Learning (ML)-based approaches use an existing dataset for learning how to determine the class, with the dataset being split into training and test sets. The training set comprises features of input data samples annotated with appropriate labels indicating the class to which the sample belongs. The classification model is trained using this training set, such that it learns the required features based on the expected output class labels given the feature vectors of the input samples. Based on the learning on the training set, the model should be able to predict the class labels for new data samples that were not a part of the training set. The performance of the classifier on unseen samples is validated using the test set. The test set samples are given as input, and the classifier predicts the output class, which is verified against the ground truth. For sentiment analysis, the input samples are text documents whose sentiment needs to be determined. ML-based approaches do not rely on a prior lexicon, but leverage the classification algorithms, including the Naive Bayes Classifier (NB) and Support Vector Machine (SVM) [5]. Some of the common features used as an input to ML classifiers are part-of-speech (POS), presence of terms, frequency of terms, n-grams, and negation [6]. In ML approaches, the model learns by using either supervised, unsupervised, or hybrid techniques. These techniques are discussed in the following subsections.

Supervised Learning. Several works in the literature have used supervised learning for sentiment classification. These classifiers are developed with the use of labeled training documents considering sentiment classification as a standard statistical classification. The predictions on new data are made on the basis of the patterns observed in the training set. In the following sections, we will discuss some of the mainly used classification methods.

A. Support Vector Machines (SVM). SVM is an ML technique that performs binary classification by determining the hyper-plane that achieves the best separation of the data that belongs to either of the two classes. The SVM classifier involves the representation of the input samples in the space as points, such that the samples belonging to different classes are separated as much as possible [7]. This algorithm separates text documents into positive and negative sentiment classes. A number of studies on sentiment classification of social media texts with supervised learning have used the SVM algorithm [8–12]. The SVM becomes suitable because the features of a text document are correlated and are linearly separable [13].

Sentiment classification with SVM has been used with a variety of features. Shein et al. [14] proposed a combined ontology-based and Natural Language Processing (NLP)-based approach in which a 3-class classification was done using a linear SVM. Luo et al. [7] compared the results obtained through SVM with the results of naive Bayes (NB) and k-nearest neighbors (KNN) algorithms in the 2-class classification of text documents represented in a vector space model. They noted that SVM outperformed the other two classifiers irrespective of the technique used for feature selection. The one-vs-one (OVO) technique is a common way to convert a multi-class classification problem to multiple binary classification problems. Liu et al. [15] used this strategy along with SVM to obtain better results for multi-class classification compared to the existing techniques.

Balahur [16] proposed a framework using unigram and bigram features from Twitter data with SVM Sequential Minimal Optimization (SVM SMO) [17]. Using SVM SMO helps in overcoming the risk of overfitting on data. The framework was extensible for use in other languages due to less processing of linguistic features and was also suitable for real-time processing. Kiritchenko et al. [18] designed a linear kernel SVM-based architecture to analyze the sentiment of short texts because the linear-kernel-based SVM performed better relative to the Radial Basis Function (RBF)-based kernel. They extracted sentiment features from the posts containing sentiment-word, hashtags, and emoticons. Their framework obtained the first rank in SemEval-2013 task 2.

The performance of SVM for the sentiment analysis task can be enhanced with the help of the grid search technique for optimizing hyper-parameters [19]. This involves the evaluation of multiple models with varying parameters. Ahmad et al. [19] noted the improved performance of the SVM classifier through this optimization on Twitter and IMDB datasets. Sharma et al. [20] enhanced the performance of SVM by combining it with Boosting algorithm, which is an ensemble algorithm. It can help in improving the ability to generalize due to the selection of the best-performing features at every step. Based on their evaluation of movie and hotel review datasets, it was shown that using an ensemble of boosting along with SVM outperforms an individual SVM classifier.

B. Naive Bayes. The Naive Bayes classifier is frequently applied, especially due to the simplicity in its assumption that the input features are independent. Despite this assumption, this classifier performs reasonably well as compared to other classifiers and has been proved to be useful in use cases, such as text classification [21]. The family of classifiers relies on the Bayes theorem that for a given sample, the class that is most likely to satisfy the input vector is assigned to the sample.

Naive Bayes classifier has been used for real-time sentiment classification of tweets due to its speed [22]. Pak and Paroubek [23] used a multinomial Naive Bayes classifier for classifying Twitter data into three classes: positive, neutral, and negative and it outperformed SVM. They used both n-gram-based features and POS tagging-based features. They also noted that some of the POS tags are good for inferring emotion. Troussas et al. [24] used the Naive Bayes classifier for sentiment analysis of status updates of Facebook users and obtained high accuracy. They showed that Naive Bayes had comparable performance with the Rocchio classifier but with a lower recall. In another work, it was shown that the Naive Bayes classifier outperformed the Maximum Entropy model for sentiment classification of updates on Twitter by users [25]. Kang and Yoo [26] proposed an enhanced Naive Bayes algorithm to address the problem of imbalance between the positive and negative classification accuracy and noted that the algorithm decreased the gap between both accuracies on restaurant review data.

Gamallo et al. [27] proposed two variations of the classifier for their work on tweets, in which the first variant included a classifier trained for three-class classification. The second variant included training the classifier on a corpus after eliminating neutral tweets and thus doing only binary classification. If the tweet has no word that belongs to a polarity lexicon, the tweet was termed as neutral. This binary classifier achieved above 80% precision on the data with two classes.

In another novel approach, Tan et al. [28] adapted Naive Bayes to a new domain for sentiment classification. They used both the previous domain knowledge along with the knowledge from the new domain data, which is unlabeled, thus treating the problem as a domain-transfer problem. They proposed a measure, called Frequently

Co-occurring Entropy, which extracts features occurring commonly in the old as well as the new domain. The features obtained from the old domain are generalizable features. They further constructed an Adapted Naïve Bayes classifier that can leverage the information from a new domain. It increases the weight given to the new domain and reduces the weight given to the old domain during the iteration process. They have shown that this modified classifier outperforms the base classifier and transfer-learning methods.

C. Artificial Neural Networks (ANN). ANNs are somewhat motivated by the neurological connections in the human brain. They contain artificial neurons, which, analogous to a brain neuron, are processing units and are interconnected to each other. The neurons in a particular layer receive the output of the neurons in the previous layer as input, and these inputs carry different weights. After processing the input, the neurons give an output to the neurons in the next layer. The weights associated with the connections among the neurons are fine-tuned based on the data that is being used to train the network. The training dataset, for which the final expected output labels are known, is used for making the network learn the appropriate weights for the connections between the neurons. Neural networks often contain multiple layers, which are also called 'multi-layer' neural networks that consist of hidden intermediate layers which are not directly connected to the input that we provide and the output that the neural network provides. These multi-layer networks are commonly used for non-linear classification problems.

Moraes et al. [29] compared ANNs and SVM for assigning the appropriate sentiment class to a text document. Using the traditional bag-of-words model for word vectorization of text coupled with ANNs resulted in an equal or better performance as compared to SVM when the datasets were not unbalanced. Duncan and Zhang [30] used a feedforward neural network in their work on Twitter sentiment analysis and noted that they faced memory issues when they tried to train the network with large vocabulary derived from a large training set of tweets. They suggested that feature reduction methods can be used for reducing the vocabulary used for training the network. However, advances in computational hardware and the increased availability of large datasets for training ANNs had rekindled the attention given to ANN architectures, especially with a focus on deep neural networks that have multiple layers [2]. Deep networks usually take the vectorized form of the text documents as input, also called word embeddings. These embeddings denote the latent features of the words in the document, and can be learned using ANNs or matrix factorization.

There are several classes of ANNs, a common one being Convolutional Neural Networks (CNN). CNNs are feedforward ANNs that comprise multiple layers with convolving filters that capture the features locally and are heavily used for image classification [31]. Bhargava et al. [32] used ANNs for the sentiment identification of Indian languages' tweets, namely, Hindi, Bengali, and Tamil. Their aim was to assess if neural network architectures outperformed the conventional machine learning models and how the hidden layers affect the performance of neural networks. They created a number of sequential models using a combination of layers based on CNNs, Recurrent Neural Networks (RNNs), and Long Short-Term Memory (LSTM). It was noted that the increase in the complexity of the language used in the text resulted in reduced accuracy and more hidden layers resulted in enhanced accuracy.

Some works focus on classifying the sentiment of individual sentences, also called sentence-level classification. As in the case of document-level sentiment analysis, ANNs are used to generate representations of sentences. However, since sentences contain lesser information as compared to a document due to their short lengths, additional features such as parse trees and part of speech tags may be incorporated to get good results [2]. Instead of relying on parse trees for syntax and semantics, newer models based on CNNs and RNNs leverage word embeddings that encode this information. Socher et al. [33] used recursive autoencoders based on vector embeddings for variable-length phrases. There are various implementations of ANNs for the sentence-level sentiment classification [33–39].

Short text blogs, which are common on social media, often have limited context and features, which makes it challenging to classify their sentiment. CNNs have been incorporated for sentiment analysis to overcome such challenges. Santos and Gatti [3] used CNNs for obtaining character, word, and subsequently, sentence-based features for movie reviews and tweets. Wang et al. [40] worked on a combined architecture with CNN and RNN for identifying the sentiment in short texts. While CNNs can capture local-level features, RNNs are suitable for capturing long-distance dependencies in data modeled as a sequence. Deep learning methods generally need a large amount of training data for learning accurately. To overcome this limitation, Guan et al. [41] proposed a weak supervised learning method for learning the initial sentence representations followed by the supervised approach based on labeled data for sentiment classification of sentences.

D. Other Supervised Learning Methods. Another class of algorithms is genetic algorithms, which are inspired by the evolution process in biology [42]. Govindarajan proposed an approach using a hybrid machine learning classification model through an ensemble of genetic algorithm and the naive Bayes classifier to obtain a more robust model with a better accuracy [43]. The hybrid classifier gives a better classification accuracy in sentiment analysis on a dataset of movie reviews. The coupling of the methods is done through an arcing classifier. The choice of the two classification techniques was based on their heterogeneity and strengths, and the hybrid approach leveraged the advantages of both the classifiers. While individually, the genetic classifier outperforms Naive Bayes, the combined classifier outperforms the individual classifiers.

Semi-Supervised and Weakly Supervised Learning. In supervised learning techniques, an essential step is the collection of data annotated with their class labels for training. However, for some domains, it is not possible to get an adequate volume of labeled data for training the classifier. In such cases, a practical solution is to use a small set of labeled documents along with unlabeled documents, which may be easily available in large numbers.

Li et al. [44] used a semi-supervised approach to account for use cases where the data is unbalanced between the two sentiment classes. They used iterative under-sampling, and dynamic random subspace generation [45]. Their proposed methods successfully used the unlabeled data, and dynamically generating the subspaces did better than a static generation. The method also outperformed baseline techniques in data pertaining to four domains. Sintsova et al. [46] used the knowledge obtained from an emotion lexicon to create a classifier for sports-related tweets on Twitter. In the semi-supervised approach, a classifier is initially used to derive pseudo-labels from the unlabeled data, after which the labels are further refined. They used the balanced weighted voting for handling unbalanced labels in the data, and this was shown to enhance the classifier performance.

The Expectation-Maximization (EM) algorithm is also commonly used in semi-supervised learning, especially, for cases where the data has missing labels [47, 48]. Zhai et al. [48] used the EM algorithm for determining the groups to which product features belong. In this approach, the labeled data is first used to train the classifier for obtaining probabilistic labels for the unlabeled samples. Soft-labels created for the set of unlabeled samples using constraints are also used for the initial classifier. After the completion of this step, another classifier is trained with labeled data, as well as unlabeled data with probabilistic labels.

Weak supervision is the broad topic that covers three types of learning: (i) Incomplete supervision, in which only a small number of samples are labeled in the training dataset, (ii) Inexact supervision, in which the labels in the dataset are more abstract or coarse-grained, and (iii) Inaccurate supervision, in which some samples may have wrong labels [49]. He and Zhou [50] proposed a framework in which a pre-existing sentiment lexicon containing words and their respective sentiment is used for training the initial classifier with generalized expectation. The classifier is then applied to unlabeled documents to get sentiment annotations. Even without using labeled data, their proposed approach was shown to perform as good or better compared to the already present weak supervision techniques.

Unsupervised Learning. Unsupervised learning, in which there are no class labels available for input data, has also been adopted for sentiment analysis. Since we do not have pre-existing class labels, the evaluation of unsupervised learning models is relatively difficult, and the decision of choosing a particular model cannot be easily explained. The aim of unsupervised learning techniques is to determine any regularities present in input samples. The prediction is done based on the structures of the input space in which certain patterns occur more often than others. In other words, these techniques utilize sentiment-based patterns to determine the labels for words/phrases.

Hu et al. [51] incorporated emotional signals for unsupervised sentiment analysis. The authors gave examples of ratings and emoticons as emotional signals in posts made by users expressing their sentiment. They work with two types of signals, (i) emotion indication and (ii) correlation. Emotion indication includes signals, such as emoticons, ratings, and stars, which give a strong indication of the conveyed sentiment. The basic idea of emotion correlation is that words often co-occurring together are more probable to have similar emotions. Using such emotional signals, the authors implemented matrix factorization method for unsupervised learning approach on publicly available Twitter and debate datasets. They noted that their unsupervised framework performed better than the baselines on these two datasets.

13.2.2 Lexicon-Based Approaches

Lexicon-based approaches utilize a sentiment lexicon for sentiment classification. This is usually done by taking a count or weighting the opinion words that are present in the text that is being analyzed [52]. Such techniques rely on the lexical resource and do not require to be trained on a training dataset. A lexicon can contain words, idioms, or phrases annotated with their sentiment polarities. A neutral or objective polarity can also be included in a lexicon. The opinion words lexicon can be constructed using a manual approach but is not time-effective. Usually, manual approaches may be used to complement automated techniques for any correction of mistakes. The common approaches used for getting the opinion words are dictionary-based and corpus-based approaches. The quality of these lexical resources contributes to the effectiveness of the sentiment classification.

Natural languages can be complex, and therefore lexicon-based approach cannot account for some characteristics of language such as slang, sarcasm, and negation [53]. Some of the difficulties of using such approaches are (i) a word's meaning can depend on their application in the sentence, (ii) sentiment words in certain contexts may not express polarity, and (iii) sentences may not include any sentiment words when expressing an opinion [54]. However, the simplicity of the lexicon-based approaches is one advantage. Enumerating the positive and negative words is easy, and the approaches also give the flexibility of adjusting to different languages and giving fast results. Such approaches are suitable when the structure of the data is complex, the data is less, and there is limited time to obtain the results. However, we can enhance the quality of sentiment analysis by using lexicon along with machine learning techniques. One way of creating an opinion lexicon is using a manual approach, but it is inefficient. Therefore, usually, automated approaches are used first, and then manual approaches are used to finally check any mistakes. The automated approaches are described in the upcoming subsections.

Dictionary-Based Approaches. In dictionary-based approaches, a seed list is initially created with opinion words through a manual approach. The sentiment polarity of these words is determined, and these words can be matched against data to be analyzed to find the text polarity [55]. The collection is then extended by extracting antonyms and synonyms of these words using a lexical resource, such as WordNet [56], SentiWordNet, and a thesaurus [57], and appropriately ascertaining the polarity of newly discovered words [58]. The new words are added to the collection through an iterative process, and finally, any errors may need to be corrected manually. An example of lexical resource is SentiWordNet, which has different versions and contains annotations for WordNet synsets based on the polarity [59].

Li et al. [60] proposed a framework related to stock predictions, which included sentiment analysis of news articles for which they used sentiment dictionaries. The sentiment dictionary is used to map the news articles into the suitable sentiment space. First, the news is vectorized in the form of term frequencies. A separate matrix that represents the sentiment dictionary is multiplied with the vectorized news, after which the news is labeled as positive, neutral, or negative based on thresholds. In another work, Filho et al. [61] evaluated the LIWC dictionary for sentiment determination of Brazilian Portuguese texts. They compared the performance against other Portuguese sentiment analysis lexical resources, including SentiLex and Opinion Lexicon and noted that LIWC performed better in detecting positive texts, however, SentiLex worked better with negative ones.

Other works using sentiment dictionary include the one by Qiu and He [62], in which they proposed an approach for advertising that leverages sentiment analysis to determine users' dissatisfaction toward a certain topic. In this approach, a pre-existing dictionary containing positive and negative words has been used to determine opinionated sentences that consist of sentiment words. Based on this, the topic words toward which the users express negative polarity are extracted. This enables us to know the product features of a brand toward which the users' have expressed negative sentiment and then display rival products' advertisements accordingly. One disadvantage of the dictionary-based approach is the inability to capture opinion words that have polarity specific to a certain domain or context.

Corpus-Based Approaches. The collection of positive and negative polarity words can also be generated using a text corpus based on an existing seed list of polarity words and the syntactic and co-occurrence patterns present in the chosen corpus [63]. This approach is helpful in incorporating domain-specific sentiment words and informal slang terms used online but which are not present in a normal dictionary [64]. A statistical or semantic approach is usually implemented with the corpus-based approach. We first discuss the work on corpus-based lexicon creation, followed by a detailed discussion on the semantic and statistical approaches.

Moreno-Ortiz and Fernández-Cruz [65] utilized both the specialized language corpus and the generic corpus to obtain domain-specific opinion words. They gave special attention to those words that have a different semantic orientation in a domain as compared to their orientation in a generic language. With this approach, they were able to obtain specialized terms with domain-specific polarity. One approach to constructing a lexicon is leveraging graph propagation. Velikovich et al. [66] used a graph propagation algorithm in which the graph nodes are the possible candidates for the polarity lexicon and the edges between representing their semantic similarity. The graph is also initialized with a seed list of polarity words with the aim of propagating the information to other nodes from the seeds. The constructed graph was based on n-grams obtained from 4 billion pages on the Web. They evaluated their resultant sentiment lexicon and noted that the lexicon contained phrases, such as spelling variants and slang, and also performed better compared to the existing lexicons for classification of the polarity of a sentence. In another approach, Hatzivassiloglou and McKeown [67] leveraged the conjunctions between the adjectives to extract semantic orientation of adjectives. Usually, conjunctions connect adjectives with similar orientations, such as ‘beautiful and intelligent’, except for the case of ‘but’ where the orientations of the adjectives are dissimilar. They used a Wall Street Journal corpus and log-linear regression to extract the orientation similarity or dissimilarity between the adjectives to construct a graph. Finally, clustering is used to separate the adjectives into the polarity classes.

In a given context, sentences with similar polarities occur near each other. Kanayama and Nasukawa [68] proposed a framework to determine entities that convey the sentiment in a given clause, for which they used intra-sentential context and inter-sentential context on a Japanese corpus. Their framework enabled enlarging of a sentiment lexicon for domain-specific sentiment analysis by identifying polar phrases specific to a domain. Sentiment lexicon can be generated by leveraging social media, which is becoming a common platform where people reveal their sentiments on different topics. Feng et al. [69] compared the usage of Twitter, Wikipedia, and the Web corpora for calculating sentiment similarity, which is a crucial step for determining polarity of words. They noted that Twitter is a good resource for measuring sentiment similarity and that the incorporation of emoticons in the seed list can enhance performance.

The following sections discuss statistical and semantic approaches in detail.

A. Statistical Approaches. Statistical techniques are used to determine co-occurrences of words/phrases. The polarity of a given adjective can depend on the noun it is being used with and the context around it. Instead of fixing the polarities of all the adjectives from the beginning, only certain adjectives are taken with a prior polarity. After this, the targets within the domain are determined. The polarity of other adjectives, also referred to as target-specific adjectives, is computed based on the occurrence of their combination with adjectives and prior polarity in a large corpus [70]. For aspect-level sentiment analysis, after the aspect detection stage, a co-occurrence algorithm [71] is used to determine aspect categories [72]. The algorithm involves a co-occurrence matrix capturing the co-occurrence count between the words. For the sentiment classification part, a sentiment lexicon is created based on the idea that words occurring near aspects with known polarity possibly are of the same polarity.

A large corpus can be used to identify words that commonly occur in positive contexts and the words that commonly occur in negative contexts, based on which the polarity of those opinion words can be determined. In short, words that often appear together in text documents are likely to share similar polarity. The Pointwise Mutual Information (PMI) is used to indicate the probability that two words will co-occur and also to determine how a pair of words are lexically related to each other. The sentiment orientation of a text can be ascertained by measuring the similarity of the words present in the document with the words that have a known polarity [73]. Turney [74] leveraged PMI [75] for determining whether a review recommends a product or does not recommend it. It is based on the idea of extracting phrases containing two words, with one of them being an adjective or an adverb. The overall orientation of an extracted phrase is calculated as the difference of the PMI of the phrase containing 'excellent', a positive word, and the PMI of the phrase with 'poor', a negative word. Finally, the average orientation of all the phrases present in the text is determined to indicate whether the review recommends the product or not. When a significantly large corpus is used for such word similarity-based approaches, sentiment analysis performance across multiple domains becomes more consistent [73].

In a star rating problem, Hogenboom et al. [76] used statistical approaches for sentiment analysis with a five-class classification. They used a vectorized representation of features in the form of sentiment words constructed using a sentiment lexicon. Their basis was that the sentiment expressed by a document is more in terms of the count of distinct words that have a semantically same orientation, and therefore, they used a binary representation. The statistical approaches were then applied to the vectorized representations. For comparing the similarity of documents, they used Jaccard similarity and cosine similarity measures. In an alternate approach, the similarity is measured by modeling classes as probability distributions based on word distributions.

Latent Semantic Analysis (LSA) is a popular technique in information retrieval that is based on the measurement of the semantic closeness of the documents based on the presence of words with similar meanings [77]. The evaluation of the latent semantic structure involves the creation of a term-document matrix, and Singular Value Decomposition (SVD) [78]. In their work on the analysis of hotel reviews given by customers to understand customer satisfaction, Xu et al. [79] used Term Frequency Inverse Document Frequency (TF-IDF) for transforming the frequency matrix and SVD for dimension reduction. In another work, Phu and Tran [80] used the Dennis Coefficient (DNC) with LSA, and the combined LSA and DNC model is used to separate documents into positive or negative classes. Their model can process a high number of documents in a short time and is also applicable to other languages.

Probabilistic LSA has also been used to extract the latent semantic topic in text documents represented using the bag-of-words model [81]. This approach constitutes the computation of the documents' probability distributions using Probabilistic LSA and then utilizes the Z-vector obtained for the sentiment classification. The authors noted that the probabilistic LSA caused a significant improvement in the results of sentiment determination, and it was also suited in the clustering scenario where multiple topics are present in a document. They tested their model on textual data from sources such as Weibo and showed that the model predicted sentiment with greater accuracy compared to the word histogram approach.

B. Semantic Approaches. Semantic approaches to sentiment analysis are proposed on the premise that words that are semantically closer have alike sentiments [82]. If a word's synonyms lean toward a particular polarity, it is likely that the word will be of the same polarity. WordNet is a popular lexical database that explains the semantic relationships between words [56]. Kim and Hovy [83] used WordNet to obtain the synonyms of a word with unknown polarity and determined a word's sentiment class by ascertaining the extent to which the word's synonyms are associated with a particular class. Their work includes the identification of opinion holders pertaining to specific topics and the classification of their sentiment on the basis of the words expressing polarity presented in the opinionated sentences.

The semantic similarity of words used by users in their posts on social networking websites with the words in a lexical database like WordNet can help us determine the polarity expressed by the users. In sentiment detection for suicide prevention based on Twitter data, WordNet was used to compare the semantic similarity of the words presented in the training data tweets with those in a test tweet [84]. In another approach for classifying tweets, Madani et al. [85] computed the semantic similarity of the tweet with the representative documents of the classes positive, neutral, and negative. The representative documents contain words that belong to a specific sentiment class, and the class with which the tweet has the most semantic similarity is its assigned polarity. Their methods gave positive results in terms of error rate and F1 measure.

Saif et al. [86] performed sentiment classification on the Twitter dataset by incorporating semantic concepts of the entities presented in the text. They evaluated the methods for obtaining entities from tweets and mapping the entities to concepts and noted that AlchemyAPI was more accurate in the mapping of entities to concepts. They proposed three ways to use the semantic information for the classifier, namely, (i) replacement of entities with their corresponding concepts, (ii) semantic augmentation, and (iii) semantic interpolation. They concluded that semantic interpolation outperformed the other two techniques and also showed that semantic techniques are suitable for large datasets spanning multiple topics.

Expert systems to perform automated analysis of weaknesses of a product are very useful as all the reviews given by multiple customers cannot be read manually. Zhang et al. [87] identified polarity associated with each product aspect or attribute for their Weakness Finder. They determined different feature words associated with the aspects of the product. A basic approach to this would be on the premise that explicit feature words belonging to the same category are usually synonyms or antonyms, which can be determined using a dictionary. However, this approach has vocabulary limitations and does not take into consideration domain knowledge. To handle this, they incorporated semantic approaches by taking into consideration the ideographic nature of Chinese characters, which gave them the opportunity to leverage word structures to find words expressing the same concept. They also used Hownet to calculate similarity scores between words [88] and thus expand the set of feature words.

13.2.3 Hybrid Approaches

Using a hybrid of lexicon and machine learning-based approaches enables us to leverage stability of lexicon approaches and the accuracy of machine learning approaches [89]. Mukwazvure and Supreethi [90] used a hybrid approach with the use of sentiment lexicon, followed by training with machine learning algorithms. On a dataset of comments derived from *The Guardian* (www.theguardian.com), the sentiment lexicon was used to decide the sentiment annotation, and machine learning classifiers were then trained using this data. The authors also proposed future work in the form of a domain-specific lexicon and the incorporation of opinion rules for getting more robust labels.

Kolchyna et al. [64] showed that the inclusion of lexicon score and some other handcrafted features enhanced the performance of SVM, Naive Bayes, and decision tree classifiers. Feature selection based on information gain showed that the lexicon sentiment score was the top input feature for the classifier. The pSenti, a hybrid of lexicon and learning-based method, is another concept-level system [91]. This approach outperformed systems such as SentiStrength when tested on datasets containing software and movie reviews [92]. The authors highlighted the advantage of their system over an only-lexicon-based approach in terms of sentiment classification accuracy.

13.2.4 Graph-Based Approaches

Approaches involving graph-based techniques have been applied for sentiment analysis. The graph-based techniques are used to obtain the sentiment from a graph-structured framework of opinions with the extracted features. The nodes in the graph are selected based on the features that occurred in the input, such as reviews and stock prices data.

Bordoloi and Biswas [93] created a graph with tokenized words for sentiment analysis of product reviews data, which helped in extracting semantic relationships. The nodes of the graph signify the tokens, and the edges represent the co-occurrence of the corresponding tokens. Main keywords are determined by assigning the degree centrality measure to the nodes in the graph. After polarity is assigned to the keywords, sentiment analysis of the datasets is done, and the model performs better than some of the existing methods. A similar approach was used by Castillo et al. [94] who constructed a co-occurrence graph and used centrality measures [95] for extracting the most important words for supervised learning-based sentiment analysis of documents.

The Potts model is a probability graph model that comprises a network of variables that can take multiple values and the values of the variable are not ordinal [96]. Graph-based approaches can be useful for the polarity classification of individual sentences, especially when the sentence does not have enough features to make a decision about its polarity. By incorporating features outside of the sentence, sentiment classification can be improved. Zhao et al. [97] used features of sentences within the same documents as well as features of sentences in other correlated documents in their graph propagation model. To determine the sentiment of an ambiguous sentence, they incorporated (i) the sentiment features of the neighboring sentences of that sentence and (ii) the sentiment of sentences in other documents that share semantic similarity with that sentence. They constructed a graph in which nodes are the sentences and connections between them represent an intra-document or inter-document relationship as mentioned above. The probability that a node belongs to a particular sentiment class in the graph is determined using the Potts model [96]. They noted that their proposed model outperformed approaches to sentiment classification that use only features pertaining to that particular sentence. Takamura et al. [98] also used the Potts model in their work on the classification of adjective-noun pairs ('phrases') based on polarity. They constructed a network of words that are interconnected based on the presence of one of the words in another word's gloss. There are two types of connections, the first type indicating the words have similar orientation while the second one indicates dissimilar orientation. They observed that their proposed approach was also successful in classifying phrases containing words that were not known before.

Goldberg and Zhu [99] presented a graph-based technique that leverages a semi-supervised algorithm for inferring the rating of a document using the sentiment analysis, especially focusing on the use case of limited availability of data annotated with ratings. Their undirected graph is comprised of documents as nodes, where an unlabeled document node is connected with the nearest labeled and unlabeled document nodes based on a similarity measure. This approach was shown to be better than the approaches that do not leverage unlabeled data in situations where labeled documents are limited. Wang et al. [100] worked on hashtag-level sentiment classification instead of analyzing the sentiment polarity of each tweet pertaining to a given topic. They improved the hashtag-level sentiment classification by constructing a graph model that captures the co-occurrence relationship between the hashtags, and also used boosting approach to further enhance the results. Through the evaluation of the proposed method on a tweet corpus, they showed that their approach improved the results over the baseline.

13.2.5 Other Approaches

In this section, we discuss the approaches that did not fall under any of the above categories. Formal Concept Analysis (FCA) is a mathematical technique that is used for the representation of knowledge and involves visualizing of concepts and any dependencies between them [101, 102]. The concept is based on lattice and set theory. The input is a matrix with boolean values where a row is an object and the columns refer to attributes. The application of FCA creates a concept lattice with a hierarchy of relationships between concepts, as well as the dependencies between attributes. With FCA allowing the analysis of complicated structures and dependencies, Shein proposed a sentiment classification approach for text documents leveraging domain ontology used with FCA [103]. After the POS tagging and FCA-based ontology steps, SVM was used for sentiment classification.

Fuzzy Formal Concept Analysis (FFCA) is used to create a conceptual hierarchy using fuzzy formal concept [104]. FFCA uses fuzzy logic, and it differs from FCA, in which a binary value is used to indicate the relationship between an object and attribute. In FFCA, the degree of relationship is indicated with a value between 0 and

1 [105, 106]. Li and Tsai [106] in their work on polarity classification used FFCA to extract the internal insights from a corpus and also to understand the abstract concepts of the documents. Their proposed approach is versatile, i.e., it can be used in multiple domains and is also less sensitive to noise. Park et al. [104] used FFCA to get a hierarchy of sentiment words, which was not provided by WordNet [56], and thus provide an extension of hierarchy-based feature-level sentiment analysis.

Cambria [107] explains concept-based sentiment analysis as textual analysis using concepts and affective knowledge through ontology and other semantic networks. This allows an understanding of subtly expressed sentiments by analyzing the interlinking of a concept with another concept that reveals emotions. The Sentic computing approach to sentiment analysis includes the usage of artificial intelligence and semantic web for representing knowledge and allows for analyzing documents at the level of concepts by understanding their dependencies [108]. Cambria et al. [108] proposed SenticNet2, which serves as a resource for all-round analysis of textual language by furnishing both the sentics and the semantics related to more than 14,000 concepts. It can be utilized in real-life applications, such as social data mining, and deal with structured as well as unstructured data.

13.3 Datasets

Here, we discuss publicly available social networking datasets that are annotated for sentiment analysis, extracted from websites such as Twitter and Facebook.

1. CINLP [109]: Tromp and Pechenizkiy collected this Twitter dataset as (i) search tweets having happy smileys like :), :-), :D, etc., for 30 min, (ii) search tweets having sad smileys like :(, :-(, :'(, etc., for another 30 min, and (iii) for neutral messages, tweets are extracted from news pages, for example, BBC, CNN (English), and EenVandaag (Dutch) for 30 min. The dataset contains 11,778 training and 859 testing English tweets, and 4,805 training and 1,057 testing Dutch tweets. This dataset is available at [110].
2. COST [111]: This dataset is extracted from Twitter using positive emoticons, such as :D, :), and XD, and negative emoticons such as :(, :-/, and D:. After preprocessing and removing tweets having emoticons from both the polarities, the dataset contains 17,317 positive and 17,317 negative tweets. A tweet's polarity is set to 1 if it is collected using positive emoticons and 0 if retrieved using negative ones. The dataset can be obtained by emailing the authors.
3. COVID-19 Tweets Dataset [112]: This dataset contains tweets related to the Covid-19 pandemic and was extracted using 54 related keywords. Each tweet is assigned a sentiment score. The dataset is available at [113].
4. EMOT [114]: This English dataset was collected by Go et al. for positive emotion and negative emotion using happy ':)' and sad ':(' emoticons. The authors used 800,000 positive and 800,000 negative emoticon tweets for training. The test data

was manually annotated having 182 positive emoticon tweets and 177 negative emoticon tweets. It is available at [115].

5. EmIroGeFb [116]: This dataset is extracted from Facebook having comments in Spanish on three different topics (i) politics, (ii) football, and (iii) celebrities. The authors selected four Facebook pages on each topic and extracted 400 comments for each topic, 200 from each category, males and females. The dataset is labeled using three kinds of tags, (i) emotions used in the text, (ii) irony usage, and (iii) gender category. It also contains meta-information for each comment like Facebook ID, topic, annotator set, etc. This can be downloaded from http://ow.ly/uQWEs.
6. Facebook Comments v1.0 [117]: Zhang et al. collected 2000 Facebook comments and manually labeled them. The comments are labeled as follows: (i) if a comment has positive and objective sentences, then it is a positive message, (ii) if it has negative and objective sentences, then it is negative, (iii) if it contains only objective sentences, it is objective, (iv) if it consists of more positive sentences than the negative sentences, it is positive mixed, and (v) finally if it has both types of sentences but negative sentences are more, it is negative mixed. The dataset is available at http://cucis.ece.northwestern.edu/projects/Social/sentiment_data.html.
7. FB Sentiment [118]: Tran and Shcherbakov collected this dataset on the 2016 United States presidential elections from two popular news channels CNN and BBC on Facebook. It contains around 100,000 comments on 200 posts. The dataset is available in CSV format at https://raw.githubusercontent.com/saodem74/Sentiment-Analysis/master/Data/comment_data.csv. The CSV file of the dataset has datetime stamp value, topic, post title, comment text, and positive and negative sentiment score (float value).
8. Facebook Status Dataset [119]: The authors collected 62,202 Facebook statuses using *iFeel*, a Facebook Connect application. They collected 25 most recent status updates from a logged-on Facebook user and their friends. After preprocessing, data is labeled in two ways. First, they labeled data with positive and negative labels containing 4,320 statuses. Next, they did multi-class labeling having four labels: unhappy, happy, skeptical, and playful; this dataset has 3,612 labeled status. As we know, the dataset link is not provided by the authors.
9. HASH [120]: This tweet dataset is compiled from the Edinburgh Twitter corpus.[1] It contains 31,861 positive, 64,850 negative, and 125,859 neutral tweets in English.
10. Health Care Reform (HCR) [121]: This dataset was collected in March 2010 by retrieving the tweets having hashtag '#hcr'. It was manually annotated with five labels (neutral, negative, positive, irrelevant, and unsure(other)) and contained 1,381 negative, 541 positive, and 470 neutral tweets.
11. InterTASS corpus [122]: This dataset was released in the 2017 TASS workshop edition. It contains 3,413 tweets (1,008 training, 506 development, and 1,899

[1] http://demeter.inf.ed.ac.uk.

testing) in Spanish. Three annotators labeled the tweet polarity as positive, negative, neutral, and none. The dataset is available on the TASS website.

12. ISIEVE [120]: This English tweet dataset was collected and hand-annotated by the iSieve Corporation (www.i-sieve.com). It contains approximately 4,000 tweets annotated with negative, positive, or neutral sentiment related to the topic of the tweet.
13. LIGA_Benelearn11 [123]: This dataset is extracted by collecting tweets in six languages, (i) Dutch, (ii) French, (iii) English, (iv) German, (v) Italian, and (vi) Spanish, and six accounts were selected for each language. This dataset is available at [110].
14. New Delhi political elections [124]: The authors collected Facebook data related to New Delhi political elections from October 4 to December 10, 2013. The dataset contained around 103,000 posts and over 30,000 users that wrote at least 5 posts on around 50 different topics. The dataset is not freely available; authors can be contacted for the details.
15. Obama-McCain Debate (OMD) [125]: It contains 3,238 tweets collected during the first U.S. presidential TV debate that happened in September 2008. This dataset was labeled using Amazon Mechanical Turk (AMT).
16. O'Connor's Corpus (OC) [126]: It consists of one billion tweets collected over from 2008 to 2009 using Twitter API. It was annotated for sentiments using hashtags and emoticons.
17. RepLab 2013 [127]: It contains 142,527 English and Spanish tweets (63,442 positive, 16,415 negative, and 30,493 neutral), out of which 28,983 are in Spanish. The tweets were labeled with the polarity to analyze their potential impact (positive or negative) on the reputation of a company. The dataset was released by RepLab and is available at http://nlp.uned.es/replab2013/, and can be used for testing using the EvAll service.[2]
18. SAB/MAS corpus [128]: This dataset contains 4546 tweets mentioning brands from different sectors, such as automotive, beverages, food, banking, retail, sports, and telecoms. The dataset is in Spanish. The tweets are labeled for four positive ((i) happiness, (ii) trust, (iii) love, and (iv) satisfaction) emotions, and four negative ((i) sadness, (ii) dissatisfaction, (iii) hate, and (iv) fear) emotions. It is available at http://sabcorpus.linkeddata.es/.
19. Sentiment Strength Twitter Dataset (SS-Tweet) [92]: Thelwall et al. collected 4,242 tweets and labeled them with negative and positive sentiment strengths. A negative strength is labeled from -1 (as not negative) to -5 (as highly negative); similarly, positive strength is labeled from 1 (as not positive) to 5 (as highly positive). It is available at [129]. [130] labeled this dataset again with three labels as positive, negative, and neutral.
20. Sanders Twitter Dataset [131]: This dataset contains 5,512 tweets on four topics (i) Apple, (ii) Twitter, (iii) Google, and (iv) Microsoft). It was manually annotated and has 570 positive, 2,503 neutral, 654 negative, and 1,786 irrelevant tweets. It is available at http://www.sananalytics.com/lab.

[2] http://www.evall.uned.es/.

21. Tweets v1.0 [117]: This contains 1,000 tweets labeled in the same way as the Facebook Comments v1.0 dataset. The dataset is available at http://cucis.ece.northwestern.edu/projects/Social/sentiment_data.html.
22. Twitter US Airline [132]: This sentiment analysis dataset contains 14,640 tweets related to major US airlines. Tweets are labeled as positive, negative, or neutral.
23. TWITA [133]: The authors collected around 100 million Italian tweets for the time period Feb 2012–Feb 2013. The authors applied a language detection process to make sure that the collected tweets are in Italian. Two subsets having 1,000 tweets are annotated by three independent native-speakers. Each tweet is assigned a label as positive, neutral, or negative.
24. Twitter1 [134]: Agarwal et al. used 11,875 manually annotated tweets in English. The data was acquired from NextGen Invent (NGI) Corporation; please contact the authors for obtaining the dataset.
25. Twitter2 [135]: This is a collection of 667 English tweets. It is available at http://goo.gl/UQvdx.
26. The Dialogue Earth Twitter Corpus [136]: This tweet corpus has three subsets, (i) WA contains 4,490 tweets about weather, (ii) WB contains 8,850 tweets related to weather, and (iii) GASP having 12,770 tweets related to gas prices. The tweets were manually labeled with 5 labels, (i) positive, (ii) negative, (iii) neutral, (iv) not related, and (v) other (cannot tell), with respect to the topic. Refer www.dialogueearth.org for more details.
27. TASS general corpus [137]: This dataset was released in 2012 for the first edition of the TASS challenge. It contains 68,017 tweets in Spanish, collected from Twitter accounts of 154 celebrities, and covers topics such as music, sports, films, politics, entertainment, economy, soccer, technology, literature, and so on. Each tweet is labeled with either of five polarity levels as (i) P+, (ii) P, (iii) NEU, (iv) N, and (v) N+, or with 'NONE' sentiment tag. The dataset is freely available at [138] after signing a non-commercial user agreement.
28. TASS politics corpus [139]: This dataset is also from TASS workshop 2013. It contains 2,500 tweets in Spanish related to four major political parties (IU, PP, UPyD, and PSOE). The tweets are tagged at entity and tweet level with three polarity levels as neutral, positive, negative, and None. The dataset is available at [140].
29. TASS social-TV corpus [141]: This dataset is from TASS 2014 edition and was used in subsequent editions. It contained 2,773 tweets in Spanish related to a football match in 2014.
30. TASS STOMPOL corpus [142]: This dataset was collected during the Spanish general election, 2015. It contains 1,284 Spanish tweets on politics-related topics such as economics, political party, education, and health system from the main political parties (IU, PP, PSOE, Cs, UPyD, and Podemos). Two people annotated the tweets for each user's opinion about each aspect as negative, positive, and neutral; the disagreement was resolved by a third annotator. The dataset is available at [143].

13.4 Future Directions

- Real-time Analysis: In applications of SA such as marketing, fake news detection [144, 145], and opinion mining, the data should be processed in real time to produce and update the results constantly. Real-time analysis is crucial, especially in use cases such as election results prediction, where people's opinions might change quickly. Social media continuously generates a large amount of data, and the information can go viral in just a few seconds. If a person posts a negative review, companies need to act fast to minimize its impact. One can therefore propose fast real-time processing tools for SA [22]. The tools should also consider the data coming from different sources; for example, an election campaign might be running on different websites (Facebook, Twitter) at the same time.
- Building Datasets: Most of the works in sentiment analysis are for English language datasets. There is still a lack of lexicon datasets and benchmark datasets for other languages. Social network data can be collected using API; however, data labeling is a concern. Some of the datasets have been annotated by authors, or they have used crowdsourced portals for the labeling. In crowdsourcing, each text is annotated by 2–3 people, and the final label is decided based on the majority [125]. Though recently researchers have focused on building resources for other languages, the creation of more datasets will be useful to expand sentiment analysis to other languages.
- Preprocessing: Real-world data is noisy and unstructured. It needs preprocessing steps for spelling correction, grammar correction, abbreviation handling, and removal of unnecessary characters and strings, such as URLs from tweets [119]. One can therefore work on designing fast preprocessing steps. There are very few works proposing optimization techniques for feature selection. One can also propose methods for automatic feature selection using machine learning and optimization techniques.
- Domain-Independent Models: The accuracy of most of the proposed models has been verified using domain-specific datasets as the training and testing for a domain-specific corpus gives a higher edge to the accuracy. Cross-domain SA was introduced to reduce the manual effort in training machine learning models, but it is still a major issue. For example, "The screen is curved" is a good review for television but a bad review for a smartphone. Sentiment vocabularies and the strength of sentiment words differ across domains. In some cases, domain-specific models are preferred, such as the review analysis for a certain kind of product, but in some cases, general domain-independent models will be preferred if the incoming text is from different kinds of sources. Based on the demand, we need to build both kinds of models having better accuracy [73].
- Multimodal Reviews: People also post reviews or opinions in the multimedia format on social networking or blogging websites. An example of this is a spoken review of a movie posted on Facebook. Sentiment analysis techniques need to consider features other than text, such as facial expressions, voice base, and emotion,

for multimodal data [146]. The recent trends and availability of multimodal data make it a potential research area.

- Reduce Computational Cost: One major challenge is the reduction of the computational cost of models, especially for real-time analysis [22]. Machine learning methods having lower training costs are in demand. Additionally, the time-taking annotation process causes less availability of datasets. Researchers, therefore, can design semi-supervised or unsupervised methods to learn significant features [46].
- Generalized Models: Researchers have proposed neural network-based models, and their results show higher accuracy for the given dataset. However, the accuracy of different methods in different cases might not be the same, and it will take researchers time and effort to assess the efficiency for the given application dataset. Therefore, more generalized methods might be preferred [147].
- Temporal data analysis: On social networking websites, people constantly keep posting content that represents their opinion, mood, and emotion for a topic. This data can be helpful in the sentiment analysis of future posts of a user. For example, if a person doesn't like a political party, then there is a very high probability that if the user posts about that party it might seem positive but will be a sarcasm. Most of the methods mainly consider lexicons or linguistic features for SA; however, considering the past data can improve the accuracy. Temporal data analysis for opinion mining is still in a fancy state and can be looked at for further research [148].
- Homophily and SA: In real-world networks, the opinion of a user is influenced by its neighbors. It is observed that similar people respond to a situation in a similar manner. One potential direction is to consider the impact of homophily, and spatial-temporal patterns in SA [149].
- Sarcasm and Irony: These are frequently used on social media and it is hard to detect sarcasm in microblog posts. In the last decade, a few researchers have explored this direction [150]. Future research works can consider exploring this further as (i) sarcastic words differ in different languages, and (ii) there is a lack of universal datasets for this type of analysis.
- Recommendation Models: In social networking websites, a user prefers to see the posts confirming their belief and opinion. Hsu et al. [151] have shown that posts containing strong sentiments receive more clicks than the posts containing neutral sentiments, and posts containing negative polarity attract more views than posts containing positive polarity. The feed recommendation model based on SA and the social network of the user will be appreciated [152].
- Bias: Challenges associated with gender, racial, regional, and age-related biases are bound to arise in sentiment analysis systems. Researchers have detected significant biases originating from the language resources in [153]. The designing of fair systems will be highly appreciated and better to apply in real-life applications.

13.5 Conclusion

This chapter has mainly focused on different types of techniques used for sentiment classification of content. These techniques include machine learning-based, lexicon-based, hybrid, and graph-based. Machine learning-based approaches leverage machine learning algorithms to learn from the available data to be able to classify the content as negative, positive, or neutral in nature. Lexicon-based techniques are further categorized as dictionary-based and corpus-based techniques and rely on a sentiment lexicon to determine the polarity of a given document. Hybrid techniques are a combination of both machine learning and lexicon-based techniques. Another novel class of techniques is the graph-based approaches, which leverage the graph-based structures among features or components of the content. Next, we discussed the datasets which are available for the research, extracted from social networking websites. Finally, we discussed open directions for this field of research, ranging from real-time analysis and interdisciplinary research.

References

1. Ya-Han, H., Chen, Y.-L., Chou, H.-L.: Opinion mining from online hotel reviews-a text summarization approach. Inf. Process. Manag. **53**(2), 436–449 (2017)
2. Zhang, L., Wang, S., Liu, B.: Deep learning for sentiment analysis: a survey. Wiley Interdiscip. Rev.: Data Min. Knowl. Discov. **8**(4), e1253 (2018)
3. Santos, C.D., Gatti, M.: Deep convolutional neural networks for sentiment analysis of short texts. In: Proceedings of COLING, the 25th International Conference on Computational Linguistics: Technical Papers, pp. 69–78 (2014)
4. Santos, C.N.D.: Think positive: towards twitter sentiment analysis from scratch. In: SemEval@COLING (2014)
5. Vinodhini, G., Chandrasekaran, D.: Sentiment analysis and opinion mining: a survey. Int. J. Adv. Res. Comput. Sci. Technol. **2**, 06 (2012)
6. Pang, B., Lee, L., Vaithyanathan, S.: Thumbs up? Sentiment classification using machine learning techniques. In: Proceedings of the 2002 Conference on Empirical Methods in Natural Language Processing (EMNLP 2002), pp. 79–86. Association for Computational Linguistics (2002)
7. Luo, F., Li, C., Cao, Z.: Affective-feature-based sentiment analysis using SVM classifier. In: 2016 IEEE 20th International Conference on Computer Supported Cooperative Work in Design (CSCWD), pp. 276–281. IEEE (2016)
8. Wu, C.E., Tsai, R.T.H.: Using relation selection to improve value propagation in a conceptnet-based sentiment dictionary. Knowl.-Based Syst. **69**, 100–107 (2014)
9. Nadia, F., Hruschka, E., Hruschka, E.: Tweet sentiment analysis with classifier ensembles. Decis. Support Syst. **66**, 170–179 (2014)
10. Smailović, J., Grčar, M., Lavrac, , N., Žnidaršič, M.: Stream-based active learning for sentiment analysis in the financial domain. Inf. Sci. **285**, 181–203 (2014)
11. Yan, G., He, W., Shen, J., Tang, C.: A bilingual approach for conducting Chinese and English social media sentiment analysis. Comput. Netw. **75** (2014)
12. Balahur, A., Perea-Ortega, J.: Sentiment analysis system adaptation for multilingual processing: the case of tweets. Inf. Process. Manag. **51**, 547–556 (2015)
13. GMedhat, W., Hassan, A., Korashy, H.: Sentiment analysis algorithms and applications: a survey. Ain Shams Eng. J. **5**(4), 1093–1113 (2014)

14. Shein, K.K.P, Nyunt, T.T.S.: Sentiment classification based on ontology and SVM classifier. In: 2010 Second International Conference on Communication Software and Networks, pp. 169–172. IEEE (2010)
15. Liu, Y., Bi, J.-W., Fan, Z.-P.: A method for multi-class sentiment classification based on an improved one-vs-one (OVO) strategy and the support vector machine (SVM) algorithm. Inf. Sci. **394**, 38–52 (2017)
16. Balahur, A.: Sentiment analysis in social media texts. In: WASSA@NAACL-HLT (2013)
17. Platt, J.: Sequential minimal optimization: a fast algorithm for training support vector machines (1998)
18. Kiritchenko, S., Zhu, X., Mohammad, S.: Sentiment analysis of short informal text. JAIR **50** (2014)
19. Ahmad, M., Aftab, S., Bashir, M.S., Hameed, N., Ali, I., Nawaz, Z.: SVM optimization for sentiment analysis. Int. J. Adv. Comput. Sci. Appl **9**(4), 393–398 (2018)
20. Sharma, A., Dey, S.: A boosted SVM based sentiment analysis approach for online opinionated text. In: Proceedings of the 2013 Research in Adaptive and Convergent Systems, pp. 28–34 (2013)
21. Rish, I. et al.: An empirical study of the naive Bayes classifier. In: IJCAI 2001 Workshop on Empirical Methods in Artificial Intelligence, vol. 3, pp. 41–46 (2001)
22. Goel, A., Gautam, J., Kumar, S.: Real time sentiment analysis of tweets using naive Bayes. In: 2016 2nd International Conference on Next Generation Computing Technologies (NGCT), pp. 257–261. IEEE (2016)
23. Pak, A., Paroubek, P.: Twitter as a Corpus for Sentiment Analysis and Opinion Mining, vol. 10 (2010)
24. Troussas, C., Virvou, M., Espinosa, K.J., Llaguno, K., Caro, J.: Sentiment analysis of facebook statuses using naive Bayes classifier for language learning. In: IISA 2013, pp. 1–6. IEEE (2013)
25. Parikh, R., Movassate, M.: Sentiment analysis of user-generated twitter updates using various classification techniques (2009)
26. Kang, H., Yoo, S. J., Han, D.: Senti-lexicon and improved naive Bayes algorithms for sentiment analysis of restaurant reviews. Expert Syst. Appl. **39**, 6000–6010 (2012)
27. Gamallo, P., Garcia, M.: Citius: A naive-Bayes strategy for sentiment analysis on English tweets, pp. 171–175 (2014)
28. Tan, S., Cheng, X., Wang, Y., Xu, H.: Adapting naive Bayes to domain adaptation for sentiment analysis. In: European Conference on Information Retrieval, pp. 337–349. Springer (2009)
29. Moraes, R., Valiati, J., Neto, W.G.:. Document-level sentiment classification: an empirical comparison between SVM and ANN. Expert Syst. Appl. **40**, 621–633 (2013)
30. Duncan, B., Zhang, Y.: Neural networks for sentiment analysis on twitter. In: 2015 IEEE 14th International Conference on Cognitive Informatics & Cognitive Computing (ICCI* CC), pp. 275–278. IEEE (2015)
31. LeCun, Y., Bottou, L., Bengio, Y., Haffner, P.: Gradient-based learning applied to document recognition. Proc. IEEE **86**(11), 2278–2324 (1998)
32. Rupal, B., Shivangi, A., Yashvardhan, S.: Neural network-based architecture for sentiment analysis in Indian languages. J. Intell. Syst. **28** (2018)
33. Socher, R., Pennington, J., Huang, E., Ng, A., Manning, C.: Semi-supervised recursive autoencoders for predicting sentiment distributions, pp. 151–161, 01 (2011)
34. Kalchbrenner, N., Grefenstette, E., Blunsom, P.: A convolutional neural network for modelling sentences. In: 52nd Annual Meeting of the Association for Computational Linguistics, ACL 2014 - Proceedings of the Conference, p. 1, 04 (2014)
35. Dos Santos, C., Gatti de Bayser, M.: Deep convolutional neural networks for sentiment analysis of short texts, p. 08 (2014)
36. Poria, S., Cambria, E., Gelbukh, A.: Aspect extraction for opinion mining with a deep convolutional neural network. Knowl.-Based Syst. **108**, 06 (2016)
37. Wang, X., Liu, Y., Sun, C., Wang, B., Wang, X.: Predicting polarities of tweets by composing word embeddings with long short-term memory, pp.1343–1353, 01 (2015)

38. Huang, M., Qian, Q., Zhu, X.: Encoding syntactic knowledge in neural networks for sentiment classification. ACM Trans. Inf. Syst. **35**, 1–27, 06 (2017)
39. Akhtar, M., Kumar, A., Ghosal, D., Ekbal, A. and Bhattacharyya, P.: A multilayer perceptron based ensemble technique for fine-grained financial sentiment analysis, pp. 540–546, 01 (2017)
40. Wang, X., Jiang, W., Luo, Z.: Combination of convolutional and recurrent neural network for sentiment analysis of short texts. In: Proceedings of COLING 2016, the 26th International Conference on Computational Linguistics: Technical Papers, pp. 2428–2437. Osaka, Japan, December 2016. The COLING 2016 Organizing Committee
41. Guan, Z., Chen, L., Zhao, W., Zheng, Y., Tan, S., Cai, D.: Weakly-supervised deep learning for customer review sentiment classification. In: IJCAI (2016)
42. Srinivas, M., Patnaik, L.M.: Genetic algorithms: a survey. Computer **27**(6), 17–26 (1994)
43. Govindarajan, M.: Sentiment analysis of movie reviews using hybrid method of naive Bayes and genetic algorithm (2014)
44. Li, S., Wang, Z., Zhou, G., Lee, S.Y.M.: Semi-supervised learning for imbalanced sentiment classification. In: Twenty-Second International Joint Conference on Artificial Intelligence. Citeseer (2011)
45. Tin Kam Ho: The random subspace method for constructing decision forests. IEEE Trans. Pattern Anal. Mach. Intell. **20**(8), 832–844 (1998)
46. Sintsova, V., Musat, C., Pu, P.: Semi-supervised method for multi-category emotion recognition in tweets. In: 2014 IEEE International Conference on Data Mining Workshop, pp. 393–402 (2014)
47. Dempster, A.P., Laird, N.M., Rubin, D.B.: Maximum likelihood from incomplete data via the EM algorithm. J. R. Stat. Soc.: Ser. B (Methodol.) **39**(1), 1–22 (1977)
48. Zhai, Z., Liu, B., Xu, H., Jia, P.: Grouping product features using semi-supervised learning with soft-constraints. In: Proceedings of the 23rd International Conference on Computational Linguistics, pp. 1272–1280 (2010)
49. Zhou, Z.-H.: A brief introduction to weakly supervised learning. Natl. Sci. Rev. **5**(1), 44–53 (2018)
50. He, Y., Zhou, D.: Self-training from labeled features for sentiment analysis. Inf. Process. Manag. **47**(4), 606–616 (2011)
51. Hu, X., Tang, J., Gao, H., Liu, H.: Unsupervised sentiment analysis with emotional signals. In: Proceedings of the 22nd International Conference on World Wide Web, pp. 607–618 (2013)
52. Hailong, Z., Wenyan, G., Bo, J.: Machine learning and lexicon based methods for sentiment classification: a survey. In: 2014 11th Web Information System and Application Conference, pp. 262–265. IEEE (2014)
53. Khan, T., Durrani, M., Ali, A., Inayat, I., Khalid, S., Khan, K.: Sentiment analysis and the complex natural language. Complex Adapt. Syst. Model. **4**, 02 (2016)
54. Akter, S., Aziz, M.: Sentiment analysis on facebook group using lexicon based approach, pp. 1–4, 09 (2016)
55. Hardeniya, T., Borikar, D.A.: Dictionary based approach to sentiment analysis-a review. Int. J. Adv. Eng. Manag. Sci. **2**(5), 239438 (2016)
56. Miller, G., Beckwith, R., Fellbaum, C., Gross, D., Miller, K.: Wordnet: an on-line lexical database. Commun. ACM **38**, 07 (2008)
57. Mohammad, S., Dunne, C., Dorr, B.: Generating high-coverage semantic orientation lexicons from overtly marked words and a thesaurus, p. 08 (2009)
58. Hu, M., Liu, B.: Mining and summarizing customer reviews, pp. 168–177, 08 (2004)
59. Baccianella, S., Esuli, A., Sebastiani, F.: Sentiwordnet 3.0: an enhanced lexical resource for sentiment analysis and opinion mining. In: Lrec, vol. 10, pp. 2200–2204 (2010)
60. Li, X., Xie, H., Chen, L., Wang, J., Deng, X.: News impact on stock price return via sentiment analysis. Knowl.-Based Syst. **69**, 14–23 (2014)
61. Balage Filho, P., Pardo, T.A.S., Aluísio, S.: An evaluation of the brazilian portuguese liwc dictionary for sentiment analysis. In *Proceedings of the 9th Brazilian Symposium in Information and Human Language Technology*, 2013

62. Qiu, G., He, X., Zhang, F., Shi, Y., Bu, J., Chen, C.: DASA: dissatisfaction-oriented advertising based on sentiment analysis. Expert. Syst. Appl. **37**, 6182–6191, 09 (2010)
63. Darwich, M., Mohd Noah, S.A., Omar, N., Osman, N.A.: Corpus-based techniques for sentiment lexicon generation: a review. J. Digit. Inf. Manag. **17**(5), 297 (2019)
64. Kolchyna, O., Souza, T.T.P., Treleaven, P., Aste, T.: Twitter sentiment analysis: Lexicon method, machine learning method and their combination (2015). arXiv:1507.00955
65. Moreno-Ortiz, A., Fernández-Cruz, J.: Identifying polarity in financial texts for sentiment analysis: a corpus-based approach. Procedia - Soc. Behav. Sci. **198**, 330–338 (2015)
66. Velikovich, L., Blair-Goldensohn, S., Hannan, K., McDonald, R.: The viability of web-derived polarity lexicons (2010)
67. Hatzivassiloglou, V., McKeown, K.: Predicting the semantic orientation of adjectives, 05 (2002)
68. Kanayama, H., Nasukawa, T.: Fully automatic lexicon expansion for domain-oriented sentiment analysis. In: Proceedings of the 2006 Conference on Empirical Methods in Natural Language Processing, pp. 355–363 (2006)
69. Feng, S., Zhang, L., Li, B., Wang, D., Yu, G., Wong, K.-F.: Is twitter a better corpus for measuring sentiment similarity? In: Proceedings of the 2013 Conference on Empirical Methods in Natural Language Processing, pp. 897–902 (2013)
70. Fahrni, A., Klenner, M.: Old wine or warm beer: target-specific sentiment analysis of adjectives, 04 (2008)
71. Schouten, K., Frasincar, F.: Finding implicit features in consumer reviews for sentiment analysis. In: International Conference on Web Engineering, pp. 130–144. Springer (2014)
72. Schouten, K., Frasincar, F., De Jong, F.: Commit-p1wp3: a co-occurrence based approach to aspect-level sentiment analysis. In: Proceedings of the 8th International Workshop on Semantic Evaluation (SemEval 2014), pp. 203–207 (2014)
73. Read, J., Carroll, J.: Weakly supervised techniques for domain-independent sentiment classification. In: CIKM 2009 (2009)
74. Turney, P.: Thumbs up or thumbs down? Semantic orientation applied to unsupervised classification of reviews. In: Computing Research Repository - CORR, pp. 417–424, 12 (2002)
75. Turney, P.D.: Mining the web for synonyms: PMI-IR versus LSA on TOEFL. In: European Conference on Machine Learning, pp, 491–502. Springer (2001)
76. Hogenboom, A., Boon, F., Frasincar, F.: A statistical approach to star rating classification of sentiment. In: Management Intelligent Systems, pp. 251–260. Springer (2012)
77. Wang, L., Wan, Y.: Sentiment classification of documents based on latent semantic analysis. In: International Conference on Computer Education, Simulation and Modeling, pp. 356–361. Springer (2011)
78. Deerwester, S., Dumais, S., Landauer, T., Furnas, G., Harshman, R.: Indexing by latent semantic analysis. J. Am. Soc. Inf. Sci. **41**, 391–407 (1990)
79. Xu, X., Wang, X., Li, Y., Haghighi, M.: Business intelligence in online customer textual reviews: understanding consumer perceptions and influential factors. Int. J. Inf. Manag
80. Phu Vo Ngoc: Latent semantic analysis using a dennis coefficient for english sentiment classification in a parallel system. Int. J. Comput. Commun. Control. **13**(3), 408–428 (2018)
81. Ren, W., Han, K.: Sentiment detection of web users using probabilistic latent semantic analysis. J. Multimed. **9**(10), 1194–1200 (2014)
82. Medhat, W., Hassan, A., Korashy, H.: Sentiment analysis algorithms and applications: a survey. Ain Shams Eng. J. **5**(4), 1093–1113 (2014)
83. Kim, S.-M., Hovy, E.: Determining the sentiment of opinions, 01 (2004)
84. Birjali, M., Beni-Hssane, A., Erritali, M.: Machine learning and semantic sentiment analysis based algorithms for suicide sentiment prediction in social networks. Procedia Comput. Sci. **113**, 65–72 (2017)
85. Madani, Y., Erritali, M., Bengourram, J.: Sentiment analysis using semantic similarity and Hadoop MapReduce. Knowl. Inf. Syst. **59**(2), 413–436 (2019)
86. Saif, H., He, Y., Alani, H.: Semantic sentiment analysis of twitter. In: International Semantic Web Conference, pp. 508–524. Springer (2012)

87. Zhang, W., Hua, X., Wan, W.: Weakness finder: find product weakness from Chinese reviews by using aspects based sentiment analysis. Expert. Syst. Appl. **39**(11), 10283–10291 (2012)
88. Dong, Z., Dong, Q., Hao, C.: Hownet and the computation of meaning (2006)
89. El Alaoui, I., Youssef, G., Messoussi, R., Chaabi, Y., Todoskoff, A., Kobi, A.: A novel adaptable approach for sentiment analysis on big social data. J. Big Data **5**, 12 (2018)
90. Mukwazvure, A., Supreethi, K.P.: A hybrid approach to sentiment analysis of news comments. In: 2015 4th International Conference on Reliability, Infocom Technologies and Optimization (ICRITO)(Trends and Future Directions), pp. 1–6. IEEE (2015)
91. Mudinas, A., Zhang, D., Levene, M.: Combining lexicon and learning based approaches for concept-level sentiment analysis. In: WISDOM '12 (2012)
92. Thelwall, M., Buckley, K., Paltoglou, G.: Sentiment strength detection for the social web. J. Am. Soc. Inf. Sci. Technol. **63**(1), 163–173 (2012)
93. Bordoloi, M., Biswas, S.K.: E-commerce sentiment analysis using graph based approach. In: 2017 International Conference on Inventive Computing and Informatics (ICICI), pp. 570–575. IEEE (2017)
94. Castillo, E., Cervantes, O., Vilarino, D., Báez, D., Sánchez, A.: Udlap: sentiment analysis using a graph-based representation. In: Proceedings of the 9th International Workshop on Semantic Evaluation (SemEval 2015), pp. 556–560 (2015)
95. Saxena, A., Iyengar, S.: Centrality measures in complex networks: a survey (2020). arXiv:2011.07190
96. Wu, F.Y.: The potts model. Rev. Mod. Phys. **54**, 235–268 (1982). Jan
97. Zhao, Y., Qin, B., Liu, T.: Integrating intra- and inter-document evidences for improving sentence sentiment classification: integrating intra- and inter-document evidences for improving sentence sentiment classification. Acta Autom. Sin. **36**, 1417–1425 (2010)
98. Takamura,, H., Inui, T., Okumura, M.: Extracting semantic orientations of phrases from dictionary. In: HLT-NAACL (2007)
99. Goldberg, A., Zhu, X.: Seeing stars when there aren't many stars: graph-based semi-supervised learning for sentiment categorization, 01 (2006)
100. Wang, X., Wei, F., Liu, X., Zhou, M., Zhang, M.: Topic sentiment analysis in twitter: a graph-based hashtag sentiment classification approach, pp. 1031–1040, 10 (2011)
101. Wille, R.: Restructuring lattice theory: an approach based on hierarchies of concepts. In: ICFCA (2009)
102. Škopljanac-Mačina, F., Blašković, B.: Formal concept analysis-overview and applications. Procedia Eng. **69**, 1258–1267 (2014)
103. Shein, K.P.P.: Ontology based combined approach for sentiment classification. In: Proceedings of the 3rd International Conference on Communications and Information Technology, pp. 112–115. World Scientific and Engineering Academy and Society (WSEAS) (2009)
104. Park, S.M., Kim, Y.-G., Baik, D.-K.: Sentiment root cause analysis based on fuzzy formal concept analysis and fuzzy cognitive map. J. Comput. Inf. Sci. Eng. **16**(3) (2016)
105. Quan, T.T., Hui, S.C., Cao, T.H.: A fuzzy FCA-based approach for citation-based document retrieval. In: IEEE Conference on Cybernetics and Intelligent Systems, 2004., vol. 1, pp. 578–583. IEEE (2004)
106. Li, S.-T., Tsai, F.-C.: A fuzzy conceptualization model for text mining with application in opinion polarity classification. Knowl. Based Syst. **39**, 23–33 (2013)
107. Cambria, E.: An introduction to concept-level sentiment analysis. In: Mexican International Conference on Artificial Intelligence, pp. 478–483. Springer (2013)
108. Cambria, E., Havasi, C., Hussain, A.: Senticnet 2: a semantic and affective resource for opinion mining and sentiment analysis. In: FLAIRS Conference (2012)
109. Tromp, E., Pechenizkiy, M.: Rbem: a rule based approach to polarity detection. In: Proceedings of the Second International Workshop on Issues of Sentiment Discovery and Opinion Mining, pp. 1–9 (2013)
110. Databases and hypermedia group, tu/e. https://www.win.tue.nl/~mpechen/projects/smm/#Datasets. Accessed 22 Dec 2020

111. Martínez-Cámara, E., Martín-Valdivia, M.T., Ureña-López, L.A., Mitkov, R.: Polarity classification for Spanish tweets using the cost corpus. J. Inf. Sci. **41**(3), 263–272 (2015)
112. Lamsal, R.: Coronavirus (covid-19) tweets dataset (2020)
113. Coronavirus (covid-19) tweets dataset. https://ieee-dataport.org/open-access/coronavirus-covid-19-tweets-dataset Accessed 22 Dec 2020
114. Go, A., Bhayani, R., Huang, L.: Twitter sentiment classification using distant supervision. CS224N project report, Stanford **1**(12), 2009 (2009)
115. Training and testing emoticon data. http://cs.stanford.edu/people/alecmgo/trainingandtestdata.zip. Accessed 22 Dec 2020
116. Rangel, F., Hernández, I., Rosso, P., Reyes, A.: Emotions and irony per gender in facebook. In: Proceedings of Workshop ES3LOD, LREC, pp. 1–6 (2014)
117. Zhang, K., Cheng, Y., Xie, Y., Honbo, D., Agrawal, A., Palsetia, D., Lee, K., Liao, W.K., Choudhary, A.: SES: sentiment elicitation system for social media data. In: 2011 IEEE 11th International Conference on Data Mining Workshops, pp. 129–136. IEEE (2011)
118. Tran, H., Shcherbakov, M.: Detection and prediction of users attitude based on real-time and batch sentiment analysis of facebook comments. In: International Conference on Computational Social Networks, pp. 273–284. Springer (2016)
119. Ahkter, J.K., Soria, S.: Sentiment analysis: facebook status messages. Unpublished master's thesis, Stanford, CA, 2010
120. Kouloumpis, E., Wilson, T., Moore, J.: Twitter sentiment analysis: the good the bad and the omg! In: Fifth International AAAI Conference on Weblogs and Social Media (2011)
121. Speriosu, M., Sudan, N., Upadhyay, S., Baldridge, J.: Twitter polarity classification with label propagation over lexical links and the follower graph. In: Proceedings of the First Workshop on Unsupervised Learning in NLP, pp. 53–63 (2011)
122. Martínez-Cámara, E., Díaz-Galiano, M.C., García-Cumbreras, M.A., García-Vega, M., and Villena-Román, J.: Overview of TASS 2017. In: Proceedings of TASS, pp. 13–21 (2017)
123. Tromp, E., Pechenizkiy, M.: Graph-based n-gram language identification on short texts. In: Proceedings of 20th Machine Learning conference of Belgium and The Netherlands, pp. 27–34 (2011)
124. Rastogi, S.S.K., Singhal, R., Kumar, R.: A sentiment analysis based approach to facebook user recommendation. Int. J. Comput. Appl. **90**(16), 21–25 (2014)
125. Diakopoulos, N.A., Shamma, D.A.: Characterizing debate performance via aggregated twitter sentiment. In: Proceedings of the SIGCHI Conference on Human Factors in Computing Systems, pp. 1195–1198 (2010)
126. O'Connor, B., Balasubramanyan, R., Routledge, B.R., Smith, N.A.: From tweets to polls: linking text sentiment to public opinion time series. In: Fourth International AAAI Conference on Weblogs and Social Media (2010)
127. Amigó, E., De Albornoz, J.C., Chugur, I., Corujo, A., Gonzalo, J., Martín, T., Meij, E., De Rijke, M., Spina, D.: Overview of replab 2013: evaluating online reputation monitoring systems. In: International Conference of the Cross-language Evaluation Forum for European Languages, pp. 333–352. Springer (2013)
128. Navas-Loro, M., Rodrıguez-Doncel, V.: Oeg at TASS 2017: Spanish sentiment analysis of tweets at document level, pp. 43–49 (2017)
129. Index of /documentation. http://sentistrength.wlv.ac.uk/documentation/. Accessed 22 Dec 2020
130. Saif, H., Fernandez, M., He, Y., Alani, H.: Evaluation datasets for twitter sentiment analysis: a survey and a new dataset, the STS-gold (2013)
131. Bravo-Marquez, F., Mendoza, M., Poblete, B.: Combining strengths, emotions and polarities for boosting twitter sentiment analysis. In: Proceedings of the Second International Workshop on Issues of Sentiment Discovery and Opinion Mining, pp. 1–9 (2013)
132. Rustam, F., Ashraf, I., Mehmood, A., Ullah, S., Choi, G.S.: Tweets classification on the base of sentiments for us airline companies. Entropy **21**(11), 1078 (2019)
133. Basile, V., Nissim, M.: Sentiment analysis on Italian tweets. In: Proceedings of the 4th Workshop on Computational Approaches to Subjectivity, Sentiment and Social Media Analysis, pp. 100–107 (2013)

134. Agarwal, A., Xie, B., Vovsha, I., Rambow, O., Passonneau, R.J.: Sentiment analysis of twitter data. In: Proceedings of the Workshop on Language in Social Media (LSM 2011), pp. 30–38 (2011)
135. Kontopoulos, E., Berberidis, C., Dergiades, T., Bassiliades, N.: Ontology-based sentiment analysis of twitter posts. Expert. Syst. Appl. **40**(10), 4065–4074 (2013)
136. Asiaee, A., Tepper, T.M., Banerjee, A., Sapiro, G.: If you are happy and you know it... tweet. In: Proceedings of the 21st ACM International Conference on Information and Knowledge Management, pp. 1602–1606 (2012)
137. Román, J.V., Lana Serrano, S., Martínez Cámara, E., González Cristóbal, J.C.: TASS-workshop on sentiment analysis at sepln (2013)
138. TASS: Workshop on semantic analysis at sepln. http://www.sepln.org/workshops/tass/tass_data/download.php. Accessed 22 Dec 2020
139. Villena-Román, J., García-Morera, J., Lana-Serrano, S., González-Cristóbal, J.C.: TASS 2013-a second step in reputation analysis in Spanish. Procesamiento del Lenguaje Natural **52**, 37–44 (2014)
140. Tass 2013 @ sepln. http://www.sepln.org/workshops/tass/2013/corpus.php. Accessed 22 Dec 2020
141. Román, J.V., Cámara, E.M., Morera, J.G., Zafra, S.M.J.: Tass 2014-the challenge of aspect-based sentiment analysis. Procesamiento del Lenguaje Natural **54**, 61–68 (2015)
142. García Cumbreras, M.Á., Martínez Cámara, E., Villena Román, J., García Morera, J.: Tass 2015–the evolution of the Spanish opinion mining systems (2016)
143. Tass 2015. http://www.sepln.org/workshops/tass/2015/tass2015.php#corpus. Accessed 22 Dec 2020
144. Kwon, S., Cha, M., Jung, K.: Rumor detection over varying time windows. PloS one **12**(1), e0168344 (2017)
145. Saxena, A., Hsu, W., Lee, M.L., Leong Chieu, H., Ng, L., Teow, L.N.: Mitigating misinformation in online social network with top-k debunkers and evolving user opinions. In: Companion Proceedings of the Web Conference 2020, pp. 363–370 (2020)
146. Soleymani, M., Garcia, D., Jou, B., Schuller, B., Chang, S.-F., Pantic, M.: A survey of multimodal sentiment analysis. Image Vis. Comput. **65**, 3–14 (2017)
147. Shelke, N., Deshpande, S., Thakare, V.: Domain independent approach for aspect oriented sentiment analysis for product reviews. In: Proceedings of the 5th International Conference on Frontiers in Intelligent Computing: Theory and Applications, pp. 651–659. Springer (2017)
148. Paul, D., Li, F., Teja, M.K., Yu, X., Frost, R.: Compass: spatio temporal sentiment analysis of us election what twitter says! In: Proceedings of the 23rd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, pp. 1585–1594 (2017)
149. Beigi, G., Hu, X., Maciejewski, R., Liu, H.: An overview of sentiment analysis in social media and its applications in disaster relief. In: Sentiment Analysis and Ontology Engineering, pp. 313–340. Springer (2016)
150. Reyes, A., Rosso, P.: Making objective decisions from subjective data: detecting irony in customer reviews. Decis. Support. Syst. **53**(4), 754–760 (2012)
151. Hsu, P.-Y., Lei, H.-T., Huang, S.-H., Liao, T.H., Lo, Y.-C., Lo, C.-C.: Effects of sentiment on recommendations in social network. Electron. Mark. **29**(2), 253–262 (2019)
152. Aivazoglou, M., Roussos, A.O., Margaris, D., Vassilakis, C., Ioannidis, S., Polakis, J., Spiliotopoulos, D.: A fine-grained social network recommender system. Soc. Netw. Anal. Min. **10**(1), 8 (2020)
153. Kiritchenko, S., Mohammad, S.M.: Examining gender and race bias in two hundred sentiment analysis systems (2018). arXiv:1805.04508

Chapter 14
Misinformation, Fake News and Rumor Detection

Naveen Arora

Abstract Misinformation, fake news and rumors have been an issue of concern for societies and nations. Societies, countries and even organizations experience the negative impact of misinformation, fake news and rumors. These are the forms of information or news that are unverified and could be false. That is why these have immense potential to harm the social system and beliefs. The use of the internet and social media is very common in the dissemination of misinformation. Fake accounts, bots-operated accounts or semi-automated accounts are predominantly used to spread misinformation, fake news and rumors. Some websites are also engaged in the process of creation and aggregation of unverified information. This chapter aims to define different categories of false and unverified information. The impact of misinformation, fake news and rumors and the causes of their dissemination have been explored. This chapter also analyzes the methods, tools and techniques that could be used to detect false information and discourage its dissemination. It is observed that there is a need to create awareness and educate internet users to spot and detect misinformation. Social media platforms and the government authorities are also using algorithms and other technological frameworks to detect and eliminate misinformation, fake news and rumors from the web. The findings of the study might be useful for internet users, academicians, policymakers, entrepreneurs and managers of news and social media industry. The outcomes of the study can be helpful in predicting, explaining and controlling the process of creation and dissemination of misinformation, fake news and rumors.

14.1 Introduction

An increase in internet penetration, availability of new technologies, growth in economy and education level has enhanced the average daily time per user spent online. Individuals search online for any information they need. Mobile, the internet and social media have become popular sources of information. In the past, people

N. Arora (✉)
Jagran Institute of Management, Kanpur, Uttar Pradesh, India

A. Biswas et al. (eds.), *Principles of Social Networking*,
Smart Innovation, Systems and Technologies 246,
https://doi.org/10.1007/978-981-16-3398-0_14

had to wait for a newspaper or had to search TV news channels at home or at the office to get news updates. But now, individuals do not wait for newspaper or TV news, they just go to the web and try to find the news updates. Individuals, who do not actively search for news and information, receive it as they log in to their social media accounts or simply browse the web. Newsfeeds from different social media platforms are also displayed as notifications on mobile phones and other gadgets. Social media has emerged as an important source for sharing information. This change in news consumption behavior has disrupted conventional models of the journalism business. As a result, many news outlets have either discontinued their operations or have scaled-down traditional businesses and are struggling to adapt to new environmental conditions. Old and reliable media outlets are now weaker, while new channels of news and information distribution are growing faster than anyone's ability to understand and regulate them. Some of the news and information distributed by emerging media channels are correct, while the remaining are false or incorrect.

14.1.1 Classification of False, Incorrect and Unverified Information

False, incorrect and unverified information can be classified into different categories. Some of the important categories are fake news, misinformation, deepfake and rumor. Fake news or disinformation has been defined as purposely developing and sharing wrong, incorrect or manipulated information with an intention to mislead and deceive the audiences with the aim of causing harm or gaining personal, financial or political advantages. Fake news is fabricated information that looks like the content of genuine news media but the intention and the organizational process are not consistent with genuine news media. Misinformation is described as the unintentional sharing of wrong or false information. Deepfakes include audios and videos that sound and appear like a real person speaking something or acting in a way that the person had never done or spoken [1]. The use of highly advanced software makes it more complicated and challenging to spot deepfakes. Another category of unverified information, rumor can be defined as information or proposition of belief which is not verified and its topic is relevant to the individuals who actively participate in the dissemination process [2]. Rumors are generally not originated from news events. Unlike misinformation and fake news, a rumor could even be true. Rumors are more prevalent when situations are ambiguous or uncertain. Misinformation, fake news and rumors are disseminated using different media and tools. Among online tools, the use of social media and websites is common. Websites and software that collect and display information from different websites or sources are called aggregators. These aggregators and content farms create or aggregate sensational, low-quality and mostly misleading information to maximize web traffic and clicks. However, fake news outlets are websites that deliberately publish false content that seems to be true. The design of these fake news outlets is substantially similar to the mainstream media outlets.

14.1.2 Power of Misinformation, Fake News and Rumors

Misinformation, fake news and rumors might influence the lives of individuals, societies, countries and even the whole world. In the current era, information and communication technology (ICT) has the ability to disseminate any information to the whole world within a few seconds. Deliberately publicized fake news or false information might influence the sincerity of a ballot. The governments and authorities have been showing their concern over misinformation and fake news [3, 4]. Even a single source of misinformation can be substantially harmful. Once an endorser of fake news gets access to press wires, it will be challenging for editors all over the world to resolve the issues. Unlike the local reports, the news that editors receive over the wire cannot be verified [5]. Current internet and social media systems have empowered everyone, who has political, social and technical skills, to disseminate large volumes of misinformation, fake news and rumors. Misleading statements and false information about organizations, products and brands can have significant financial and non-financial consequences. Rumors within organizations can result in confusion, low productivity and sometimes even violent conflicts between groups. Misinformation, fake news and rumors have also interfered with medical decisions. People in many countries have refused to vaccinate themselves and their family members as there was some false news about vaccines. Researches suggest that even if misinformation and fake news disseminators might not be of much influence over the media landscape at large, they can set the agenda for partisan news outlets. The power to set agenda is important as it decides which topics get the public's attention [6]. The domestic news agenda varies largely depending on states and nations. In the U.S. and U.K., misinformation and fake news emphasize political issues; however, in countries like Austria and Germany, sensational and appealing content about refugees is more prevalent [7].

14.1.3 Role of Social Media Platforms in Dissemination of Misleading Information

Misleading information and its dissemination is not a new issue for society and organizations. Countries have been suffering from the problems of fake news and rumors that sometimes have even resulted in communal riots and violence against some groups of societies. In past, disseminators of misleading information used telephones, personal meetings, posters and handbills to disseminate fake news, misinformation and rumors. However, in the current system, the availability of the internet and social media platforms provides the fertile ground for easier and faster dissemination of misleading information which is dangerous for any society. Social media platforms have actually given everyone access to an unlimited number of audiences. Because of a large number of sources and the creation of echo chambers, misleading information on social media platforms is more dangerous and potent. Detection of misinformation, fake news and rumors on social media is challenging because of the rapid growth of information sources. Filter bubbles and echo chambers are created as people tend to follow and connect with like-minded individuals. As a result, no conflicting information is available to counter misinformation and the general consensus within the isolated social groups. This leads to a lack of shared reality that might be divisive and a threat to society [8]. Such situations can create some discriminatory and inflammatory content that can be shared with the general public and might be considered as fact. This divisive content can further be applied to victimize some individuals or communities, to normalize preconceptions, to toughen us-versus-them attitudes, and sometimes to provoke and rationalize violence [9]. The role of social media platforms in the dissemination of misleading information is a matter of concern because of their broader reach to the larger segment of the population that is less informed and does not get information from other reliable sources. That is why these individuals are more vulnerable to get affected by ideologically inclined news [10]. The rise of messaging app WhatsApp and its ability to disseminate misinformation, fake news and rumors through its closed messaging system is being seen as a new threat to societies, states and nations. In particular, countries like India and Brazil face major challenges of social instability because of misleading information disseminated through this messaging app. To date, WhatsApp does not have any concrete mechanism to identify, flag or remove any misleading information. Social media platform Twitter is also one of the important platforms used for the dissemination of misinformation, fake news and rumors. In comparison to verified information, misinformation on Twitter is generally retweeted by much more users and disseminates more rapidly [11]. However, sharing and liking any content on social media platforms do not necessarily mean consumption of that content [12].

14.1.4 Facebook Influenced the U.S. Presidential Elections 2016

Internet and social media platforms have empowered disseminators of misinformation so much that sometimes it presents a serious threat to countries' democracy. As more and more internet users have increasingly started relying on social media as a source of information, the public's vulnerability to misinformation has also grown. Researches suggest that at least 62% of American adults get information and news from social media platforms like Facebook [13]. In the U.S., during the 2016 presidential election, Facebook played a significant role in the spreading of misinformation and fake news [14]. However, Facebook addressed these concerns in

December 2016 and started “Disputed” tags for stories in its news feed that had been found to have false information [15]. Subsequently, in April 2017 and May 2018, Facebook promoted tips for identifying wrong information at the top of the newsfeed [16, 17].

14.1.5 Negative Content and Trust

Articles with misleading information are generated because there is demand for them. Online search engines and social media platforms use some algorithms to enhance user experience and personalize the content, news and advertisements. The content is selected on the basis of users’ past online activities, location and social connections. Social media platforms’ business model is based on the revenues from advertisements and the content that increases revenues would be prioritized by the algorithms [18]. This business model is actually responsible for the evolution of clickbait, i.e. creation of news content with sensational and appealing headlines solely to capture attention and increase users’ clicks [19]. Therefore, negative content always gets priority by algorithms as it is shared more frequently than positive content. Some researchers suggest that because of these algorithms, users generally do not receive messages that contradict their beliefs and prejudices. It is unfortunate that the most important emotions that increase the likelihood of misinformation getting viral are anger, fear and uncertainty. Internet and social media users are more likely to believe the news and stories if they come from friends, relatives, peer groups or other related sources. Those stories and news are more believable that are popular, and to which the users are exposed third or fourth time. Credibility increases with an increase in familiarity.

This study aims to explore the concepts of misinformation, fake news and rumor. The impact of misinformation and the causes of its dissemination have also been examined. The study also intends to explore the ways to detect and diminish the dispersion of misinformation, fake news and rumors.

14.2 Literature Review

Fake news is a news article or piece of information that is deliberately and verifiably false [20]. It can also be defined as news or information that is factually false and designed to mislead the user into believing it is true [21]. It is information or news published and propagated using media that carries false information irrespective of the objectives and means behind it [22]. In comparison to conventional news media such as radio, television and newspaper, the creation and broadcasting of misinformation and fake news online is cheaper and faster [23]. The absence of a physical distance barrier and the ability to share, vote, forward and review enhances users' engagement on social media platforms. This encourages users knowingly or unknowingly to participate in the dissemination of misinformation and fake news [24].

14.2.1 The Psychology of Misinformation, Fake News and Rumors

The role of psychological factors is important in determining how individuals process information. In social media, users find it challenging to determine the authenticity of news sources where news content is sandwiched between photos and videos of friends, relatives and colleagues [25]. Depending on motivation and ability, an individual processes content centrally or peripherally. Individuals do not have direct exposure to the events, incidents, speeches and workings of politics. Their knowledge is actually based on the information and news retrieved, shared and communicated by others. Thus mostly individuals' decision-making does not depend on their own judgment but on shared group-level discussions [26]. It has been observed that the credibility of the source deeply influences the social interpretation of the message or information [27, 28]. People rely on the information that comes from familiar or well-known sources. Further, people are slanted information-seekers. They seek information that supports their existing ideologies. As a result, individuals tend to update biased content on political issues [29]. A new piece of information is accepted uncritically if the source is assumed to be trustworthy or the information supports existing views. However, unfamiliar information or information coming from an unreliable source is ignored. Individuals' beliefs do not necessarily change with the correction of misinformation [3]. In fact, providing individuals with disputing information might have a negative impact and they may entrench in their initial beliefs. Most of the time, the misinformation persists even when the individuals believe in correction. It has been observed that repetition of misinformation is harmful; even in the context of contradicting it [30]. Because of fluency and familiarity biases in individuals' cognitive processing, they tend to believe information if they are familiar with it and they get more familiar with the information that they counter more frequently [31]. Thus frequent exposure to misinformation leads to long-term effects, while the corrections have short-term effects. Social pressure also influences the acceptance of any information. While sharing the information, individuals tend to preserve their reputation and avoid sharing that information, which their social groups might perceive as fake. It is also observed that individuals who tend to avoid conflicts are less likely to share the information they think might be controversial [32]. This tendency can be helpful for fact-checking tools on social media. These tools discourage sharing of disputed information, but might not have a long-term effect on beliefs [31]. While fact-checking tools warn that a user might be sharing fake news, another opportunity to intervene is to shift peer consumption online. Online users can be motivated to interact and communicate with varied social groups. This can also be an effective measure to dilute polarization and misrepresentation of facts around social and political issues. Novelty has also been found to be one of the important drivers in the dissemination of misinformation, fake news and rumors. Individuals tend to share information that they perceive is unique and new. Thus, they try to be "more informed" among their peer group. In a society, highly influenced by social media, individuals actually define their social identity through their likes, shares and posts on social media platforms. Social media platforms are being used to indicate social networks—what an individual believes, endorses, supports and opposes [33]. This approach sometimes motivates individuals to share false information even if they know that it is not true.

14.2.2 Misinformation, Fake News and Rumor Dissemination Process

Social media is one of the most important platforms used by fake news websites for the dispersion of fake news and engagement of users. Misinformation and fakes news are disseminated from origin to consumers through a complicated ecosystem of social media platforms, websites and bots. Ease of sharing and re-connecting social groups make social media engaging. This facilitates individuals and bots to manipulate information and become a powerful source of fake news and misinformation [34]. Because of unfollow/unfriend tools and social influence, the structure of social media is segregated and polarized [35]. This results in highly homogeneous echo chambers [36], which is an ideal condition for confirmation bias and selective exposure. These echo chambers are dense and clustered, so the information disperses efficiently and every member is exposed to the same information from different sources [37]. Misinformation has a higher probability to go viral in these isolated groups [38]. Even if social media users want to share only correct information, factors like information overload and limited attention make it difficult to differentiate between correct and incorrect information at the system level. This helps incorrect information to disseminate as virally as correct information [39]. Social bots are also found to be quite influential in the dissemination of misinformation, fake news and rumors [40]. Bots are designed in such a way that they enhance the reach of misinformation [41]. In addition to bots, "Breaking news" sites also play a key role in the dissemination of misinformation, fake news and rumors [42]. Using Twitter accounts to mimic reliable news outlets, these sites acquire a large credulous follower base. These websites and social media accounts share misinformation and fake news in a definitive tone. As people do not tend to give attention to every news and information, disseminators of misleading information use consumers' psychology to disseminate misinformation more efficiently. Thus information shared during a period of demand has a higher probability to disseminate efficiently [43]. It has been observed that an important channel for the dissemination of fake news and misinformation is those individuals who share lots of information in general. These individuals do not verify and sometimes even do not read the whole message before sharing. Further, individuals with certain characteristics are more likely to share misinformation and fake news: older [44, 45] and politically extremists tend to share misinformation more than others [46]. For a long time, politically motivated organizations, states and nations have been involved in creating and disseminating misinformation, fake news and rumors. It has been found that the impactful dissemination of misinformation due to simple misunderstandings is very rare and misleading information is often the outcome of planned strategic campaigns that help in achieving some particular political and military objectives [47]. Individuals, who are reluctant to accept changes and new ideas, also suspect fact-checking tools and websites [20]. This makes them more vulnerable to fake news and misinformation.

As fake news are created and dispersed with an intention to deceive, those are presented in many different and interesting forms to enhance users' engagement. Besides traditional media format, fake news are also dispersed using memes and other entertaining formats. This content helps in repeat exposure to the fake news and being interesting, this content is self-propagating and lives long on the websites, cell phones and message boards [14]. Social bots and Cyborgs (individuals whose accounts are controlled by apps) are found to be influential in the dissemination of misinformation, fake news and rumors [40]. These bots are meant to increase the dissemination of misinformation [41]. Social bots identify the accounts of influential users on social media and instigate them to share misinformation [48]. Analysis of the online behavior of individuals, who are more likely to share misinformation, shows that individuals who share lots of news online tend to share more misinformation, fake news and rumors than others. Older people and individuals with extreme political and religious approach also appear to share more misinformation [49]. As a result of selective exposure, internet users give preference to that information that supports their preexisting attitudes [29]. This information seems more persuasive because of confirmation bias [23] and as a result of desirability bias, individuals tend to accept the information that pleases them [37].

14.2.3 Identification of Misinformation, Fake News and Rumors

Identification of misinformation, fake news and rumors has been challenging [3]. The uses of advanced technologies, the internet and social media by disseminators of fake news have made this task even more challenging. Different social media platforms have introduced many tools to identify misinformation, fake news and rumors. The use of "disputed" tags for unverified information has become a common practice among social media platforms. Though "disputed" tags are considered to be helpful in reducing the belief in wrong information, they might be ineffective in undermining "exposure effects" over time [50]. "Disputed" tags also tend to develop an "implied truth" effect in which users consider untagged false information as verified and true. In reducing the influence of misinformation and fake news, specific warnings are found to be more effective than generalized warnings [51]. The role of fact-checkers is found to be important in the identification and tagging of misleading information. Social media platforms and many websites are hiring fact-checkers. This process has also been outsourced to third-party fact-checkers. However, the process of fact-checking takes time. For Facebook, it takes over three days to apply the "disputed" tag after fact-checkers found information as misleading or fake [52]. Researchers suggest that, in comparison to misleading information on social media platforms, there is less fact-checking and there is a substantial time delay after the dissemination of the original misleading information [41]. Thus, disseminators of misinformation get enough time to spread it, before it is tagged as "disputed". However, it is also argued that warnings like "Disputed by 3rd party fact-checkers" only partially reduce the perceived accuracy of the misleading information [50]. Fact-checking websites are also not much efficient in influencing the domestic news agenda [6]. Social bots perform some important activities in creating a dense ecosystem of misinformation and fake news [53]. Social media accounts handled by bots can be identified through structural, temporal, content and user features [54]. Reconstructive interviews that intend to determine the processes of how journalists create new stories can also be helpful in identifying misinformation and fake news [55]. These might identify misinformation and fake news from sources like partisan media sources, alternative media sources and mainstream media sources. Misinformation, fake news and rumors posts are designed in such a way that they seem real and correct. Therefore simply emphasizing the text content might not be much helpful in the detection of misleading information [56]. Subsequently, different methods of misinformation and fake news detection have been classified under four categories: content-based misinformation detection, context-based misinformation detection, propagation-based misinformation detection and early detection of misinformation.

14.2.4 Correction of Misinformation, Fake News and Rumor

Individuals have a tendency to evaluate the information and news that they are exposed to. This tendency varies depending on an individual's attitude, culture, education, age, gender and other factors [23]. Information is processed for verification and then incorrect information or misinformation is disregarded and corrected. However, it is observed that corrected misinformation persistently influences individuals' memory and thought processes [57, 58]. In the case of correction of misinformation, if the correction is supported by a more detailed explanation, it leads to a more sustainable change in belief. In comparison to myth retractions, fact affirmations over the course of a week support more sustainable change in belief. However, sometimes corrections are found to be ineffective as corrections generally repeat the misinformation and that makes misinformation more familiar [28]. This familiarity leads to an illusion that the news is correct and subsequently builds up an individual's belief "illusion truth effect" [59–61]. Correction of misinformation is found to be effective if it comes from a co-partisan with whom an individual is more likely to agree [27]. Researchers suggest that refutations are found to be more effective than retractions in the change in belief over long periods of time. While directly addressing the misconception, refutations elaborate on the causes of why particular information is false and where the false belief originated from [62]. By providing more detailed information that can be recollected later in support of correction, refutations identify the inconsistencies between an individual's beliefs and corrective information [63]. It is observed that direct opposition of misinformation could be counterproductive as it is seen as threatening and thus fastens individuals with their beliefs [29].

Misinformation, fake news and rumors have the capability to influence democratic societies and nations drastically. During the 2016 U.S. elections, fake stories and news received more likes and shares than the real stories published by major news agencies [14, 20]. Misinformation could be highly dangerous and divisive when there is no conflicting information to correct the wrong information or when there is general consensus within the isolated religious or social groups [8]. With the predominant growth in the influence of social media, fake news is a global issue in the current political and business environment. Subsequently, detection of fake news and its correction has become extremely important [64, 65].

14.2.5 Combating Misinformation, Fake News and Rumor

To provide an unbiased scientific treatment to the issue and combat misinformation and fake news, more conservatives are required to bring into the deliberation process about misinformation. To make truth "louder", reliable sources of information need to be strengthened and those sources must be supported by different communication channels to enhance the penetration of high-quality content. Emotional connections and repetitions are helpful in building trust in institutions and facts. To build trust among audiences, news outlets need to keep audiences informed by focusing on storytelling, impartial coverage and repetition of facts [66]. Computer-assisted detection of misinformation and fake news is also found to be effective in combating misleading information. Further, in a digital information environment, efforts are required to directly increase the media literacy of the individuals [67]. Facebook's guide on "Tips how to spot false news" also needs to be evaluated to increase media literacy. In addition to "disputed" tag for misleading information, "verified" tag for correct information can also be used in countries/regions where reliability for news media sources is relatively high (such as Germany and Spain). However, this strategy might not work in environments where reliability for news media sources is lower [68]. For stemming dissemination of misinformation, fake news and rumors, it is suggested that social identity should be leveraged at the point of origin instead of at the point of reception, As online information primarily disseminates through fewer but influential spreaders, efforts that restrain those spreaders from sharing misinformation can be more effective [69]. Misinformation, fake news and rumors are contagious.

Just like contagious diseases, they spread from one person to another person. To combat the dissemination of misinformation, researchers explored the immunization of individuals through attitudinal inoculation. The process of attitudinal inoculation is somewhat similar to immunization through vaccination. In this process, individuals are exposed to misinformation which is subsequently refuted. They are then warned that they might be exposed to the information that questions their opinions and thoughts. This approach can better explain and immunize individuals to the same type of misinformation in the future [70]. Social media platforms and major websites have been working on handling the issue of misinformation, fake news and rumors. Facebook, Google and Twitter have their own teams of fact-checkers to combat misinformation and fake news. The use of artificial intelligence (AI), machine learning and some proprietary tools is also found to be helpful in checking misinformation on different platforms. In the wake of claims that Facebook was used to disseminate misinformation to a great extent during the 2016 election in the USA [71], Facebook introduced several tools to control the dissemination of misinformation through its platform [15]. In 2018, Facebook announced its plans to establish two new AI labs that would help in combating misinformation and fake news [72].

It has been observed that the challenge of misinformation, fake news and rumors cannot be handled with any single mechanism, whether technological, legal or psychological. An effective solution to this challenge needs the inclusion of different segments of society, organizations and governments.

14.3 Discussion

Identification of misinformation, fake news and rumors and mitigation of their dissemination is a socially critical problem that is also technically challenging because of various reasons. Misinformation, fake news and rumor have been formally defined and characterized in previous sections. Problems and challenges associated with misinformation, fake news and rumors have also been described. Characterization of fake news can be used for the detection of fake news [23]. An extensive examination of the available tools and methods applicable for detection and mitigation is presented in the following sections. Further, to facilitate researchers and other stakeholders with assessment and comparison of misinformation, fake news and rumors detection, a list of available sources along with their characteristic features has been compiled.

The study suggests two basic approaches for the detection of misinformation, fake news and rumor and mitigation of their dissemination.

1. Educating internet users and
2. Detection by platforms and authorities.

14.3.1 Educating Internet Users

Educating internet and social media users to identify and detect misinformation, fake news and rumors could be critical as it could also help in mitigating the dissemination of misinformation. For this, the users might be provided with feedback that particular news might be fake, and thus users will be discouraged to share it. To help the detection of misinformation, the involvement of all the stakeholders is important. With the increased involvement of conservatives and journalists, the correct information will be "louder". Further, social media users can verify the information by examining the number of links it has. Misinformation and fake news generally receive substantially fewer links than genuine media sources. Internet users must verify the source or the author of the news or information. If the source is trusted, then the users should read the whole story because sometimes the headline might be misleading to attract traffic and clicks. A careful assessment of the website, its URL (Uniform Resource Locator), images or news can also help in detecting misinformation, fake news and rumors. An amateur-looking website, news or article with too many ads and altered/stolen pictures is more likely to spread misinformation. The URL of deceptive websites mimics the URL of legitimate websites. "About us" and "Contact us" sections of deceptive websites are either missing or have incomplete information. As misinformation, fake news and rumors are not created by professionals, grammar, spelling and punctuation mistakes are also common.

Most of the social media accounts that spread fake news or misinformation have been estimated to be semi-automated accounts or operated by robots. Researchers suggest that these accounts can be identified on the basis of some signs. Automated or semi-automated accounts generally have a substantially high posting rate with round-the-clock posts or posts at regular intervals. Most of these accounts share, re-tweet or like the content of some other specific accounts. Automated accounts rarely post any original content. Robots-operated and fake accounts have little or no biographical information with fake profile pictures that generally show non-human pictures. If human pictures are used, those are rarely changed or some celebrities' pictures are used as profile pictures. As a result of a follow-back process, fake and automated accounts follow a large number of accounts and most of them are also fake and automated accounts.

Messages that have links to trusted websites along with text, try to use these links to pretend that the text has been taken from the site. Users must verify the information by clicking the link given with the text because possibly that website might be actually denying the claim being made in the text. Prominent newspapers have also added a regular section in which misinformation, fake news and rumors are corrected with detailed information. Some dedicated websites can also be helpful in detecting misinformation and extracting the correct information. These websites list misinformation, fake news, rumors and deepfakes along with the true facts and figures. Internet users must verify the news and information from other authenticated sources if it largely deviates from facts. Table 14.1 exhibits the detailed information of websites that can be used for fact-checking.

Table 14.1 A comparison among expert-based fact-checking websites

Website	Countries/region	Content analyzed	Characteristic features
https://www.Factcheck.org	The USA and neighboring countries	News, speeches, interviews, debates	Allows users to ask specific questions
https://www.Boomlive.in	India and the rest of the world	News, videos, images	Publishes news in different categories
https://www.Snopes.com	World	News, videos, images	Allows users to submit the topic for information
https://www.Altnews.in	India	News, videos, images, speeches	Publishes news in different categories
https://www.Webdunia.com	India and the rest of the world	News, videos, images, interviews etc	Available in English, Hindi and other regional Indian languages
https://www.Abplive.com	India and the rest of the world	News, images, videos, speeches	Section "Sachchai ka sensex" dedicated to a fact-check
https://www.Exposingtheinvisible.org	World	News, videos, images	Also lists some other fact-checking websites
https://www.Latestly.com	India and the rest of the world	News, videos, images, interviews	"Viral" section for fact-check
https://www.Politifact.com	The USA and neighboring countries	News, videos, Facebook posts, images	Section "Truth-o-meter" classifies news in different categories

Those accounts who share more information or who have shared misinformation earlier should not be trusted, and information shared by these accounts must be verified. Social media users should be encouraged to give feedback for such accounts and information. Users must also try to find cues about the time and place claimed in the shared videos, images or audios. Further, those should be carefully checked for any fabrications and manipulations. For images, users can also use Google reverse image search that could help to detect the origin and context of images. However, for videos, thumbnails of videos can be searched in Google reverse image search (Fig. 14.1).

14.3.2 Detection by Platforms and Authorities

Some algorithms and bots are being used by the platforms like Facebook, Twitter and Instagram to detect fake news and prevent its exposure to web users. As most fake news origin from a few sources, the platforms identify those sources or websites

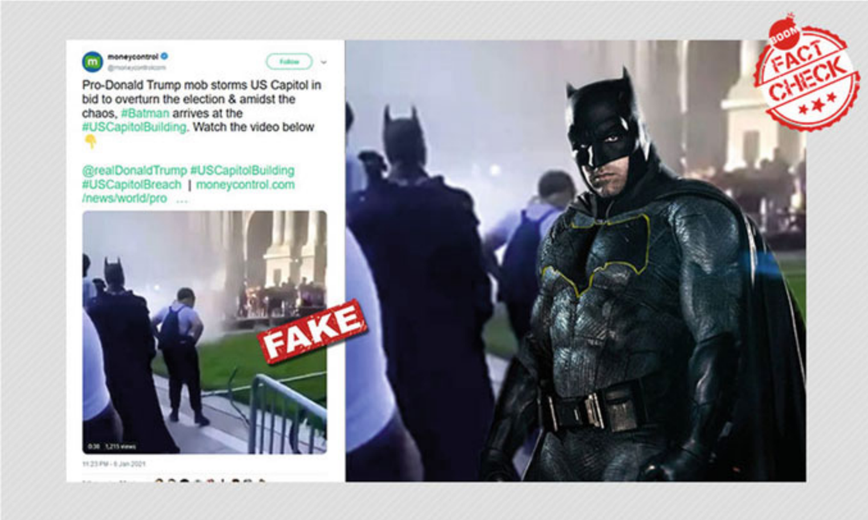


Fig. 14.1 A website showing the result of fact-checking. *Source* https://www.boomlive.in/fact-check/batman-us-capitol-protests-donald-trump-11444

and then block or discourage the dissemination of information from those sources. In order to warn the users against potential misinformation, some platforms flag the information that could be false. On the basis of users' feedback, the platforms also suspend/block suspicious accounts and information. Social media platforms are also hiring fact-checkers who are responsible to verify the information being shared online. Some third-party fact-checkers also provide services to different social media platforms and websites.

The governments and authorities also have a vigilance system for the detection and removal of misinformation, fake news and rumors published online. As local reporting has been a trusted and valuable source of news and information, efforts are being made to encourage and support local reporting. The role of local reporting is crucial in the detection of misinformation and its correction.

14.4 Conclusion

The study explores the concepts of misinformation, fake news and rumors. It is observed that misinformation, fake news and rumors have the ability to influence societies, countries and even larger regions. The social and psychological foundations that encourage or discourage the dissemination of misinformation have been explored. The methods, tools and techniques to detect misinformation, fake news and rumors have also been analyzed. The study concludes that correction of misinformation might not be effective in long-term change in belief. Misinformation, fake news and rumors continue to influence individuals' beliefs even after correction. Educating internet and social media users can be helpful in the detection of misinformation and diminishing its dissemination. The role of social media platforms and governments is also important.

Even if the study explores many tools and techniques to detect fake news and misinformation, individuals tend to not making much effort to check the authenticity of the news. Information that is familiar or that matches with the individual's belief are spread and shared without evaluation. It is recommended that the government have strict laws and rules, like fines and penalties for citizens for sharing and spreading misinformation. This could discourage them to share news without verification. Further, the social pressure of embarrassment for sharing fake news and misinformation could also encourage individuals to verify the correctness of the information before sharing. Rather than accusing social media of diffusion and dissemination of misinformation, it should be used as an effective tool to mitigate the impact of misinformation, fake news and rumors.

References

1. Digital, Culture, Media and Sport Committee: Disinformation and fake news: Final report, House of Commons, UK (2019)
2. Rosnow, R.L., Kimmel, A.J.: Rumor. In: Kazdin, A.E. (ed.) Encyclopedia of Psychology 7, pp. 122–123. Oxford University Press & American Psychological Association, NY (2000)
3. Flynn, D.J., Nyhan, B., Reifler, J.: The nature and origins of misperceptions: understanding false and unsupported beliefs about politics. Polit. Psychol. **38**(S1), 127–150 (2017)
4. Lazer, D.M., Baum, M.A., Benkler, Y., Berinsky, A.J., Greenhill, K.M., Menczer, F., Metzger, M.J., Nyhan, B., Pennycook, G., Rothschild, D., Schudson, M.: The science of fake news. Science **359**(6380), 1094–1096 (2018)
5. McKernon, E.: Fake news and the public: how the press combats rumor, the market rigger, and the propagandist. Harper's Mag. (1925)
6. Vargo, C.J., Guo, L., Amazeen, M.A.: The agenda-setting power of fake news: a big data analysis of the online media landscape from 2014 to 2016. New Media Soc. **20**(5), 2028–2049 (2018)
7. Humprecht, E.: Where 'fake news' flourishes: a comparison across four Western democracies. Inf. Commun. Soc. **22**(13), 1973–1988 (2019)
8. Benkler, Y., Faris, R.Roberts, H., Zuckerman, E.: Study: Breitbart-led right-wing media ecosystem altered broader media agenda. http://www.cjr.org/analysis/breitbart-media-trump-harvard-study.php. Accessed 22 Jan 2020
9. Greenhill, K.M., Oppenheim, B.: Rumor has it: the adoption of unverifies information in conflict zones. Int. Stud. Q. **61**(3) (2017)
10. Benedictis-Kessner, J., Baum, M.A., Berinsky, A.J., Yamamoto, T.: Persuasion in hard places: accounting for selective exposure when estimating the persuasive effects of partisan media. Harvard University and MIT (2017)
11. Vosoughi, S., Roy, D., Aral, S.: The spread of true and false news online. Science **359**(6380), 1146–1151 (2018)
12. Guess, A., Nyhan, B., Reifler, J.: Exposure to untrustworthy websites in the 2016 U.S. election. https://www.dartmouth.edu/~nyhan/fake-news-2016.pdf. Accessed 19 May 2020
13. Gottfried, J., Shearer, E.: News use across social media platforms 2016. https://www.journalism.org/2016/05/26/news-use-across-social-media-platforms-2016/. Accessed 10 Mar 2020

14. Silverman, C.: This analysis shows how viral fake election news stories outperformed real news on Facebook. https://www.buzzfeednews.com/article/craigsilverman/viral-fake-election-news-outperformed-real-news-on-facebook#.qqXqL1AJV. Accessed 22 May 2019
15. Mosseri, A.: Addressing Hoaxes and fake news. https://about.fb.com/news/2016/12/news-feed-fyi-addressing-hoaxes-and-fake-news/. Accessed 13 Feb 2020
16. Constine, J.: Facebook puts links to 10 tips for spotting false news atop feed, https://techcrunch.com/2017/04/06/facebook-puts-link-to-10-tips-for-spotting-false-news-atop-feed/. Accessed 29 June 2020
17. Owen, L.H.: Is your fake news about immigrants and politicians? It all depends on where you live. https://www.niemanlab.org/2018/05/is-your-fake-news-about-immigrants-or-politicians-it-all-depends-on-where-you-live/. Accessed 19 Apr 2020
18. Ciampaglia, G.L., Nematzadeh, A., Menczer, F., Flammini, A.: How algorithmic popularity bias hinders or promotes quality. Sci. Rep. **8**(1), 15951 (2018)
19. Bakir, V., McStay, A.: Fake news and the economy of emotions. Digit. J. **6**(2), 154–175 (2018)
20. Allcott, H., Gentzkow, M.: Social media and fake news in the 2016 election. J. Econ. Perspect. **31**(2), 211–236 (2017)
21. Golbeck, J., Mauriello, M., Auxier, B., Bhanushali, K.H., Bonk, C., Bouzaghrane, M.A., Buntain, C., Chanduka, R., Cheakalos, P., Everett, J.B.: Fake News vs Satire: A dataset and analysis. In: Proceedings of the 10th ACM CONFERENCE 2018 on Web Science, pp. 17–21. ACM (2018)
22. Sharma, K., Qian, F., Jiang, H., Ruchansky, N., Zhang, M., Liu, Y.: Combating fake news: a survey on identification and mitigation techniques. ACM Trans. Intell. Syst. Technol. **37**(4), 111:1–111:41 (2019)
23. Shu, K., Silva, A., Wang, S., Tang, J., Liu, H.: Fake news detection on social media: a data mining perspective. ACM SIGKDD Explorat. Newsl. **19**(1), 22–36 (2017)
24. Zhou, X., Zafarani, R.: A survey of fake news: fundamental theories, detection methods and opportunities. ACM Comput. Surv. https://arxiv.org/pdf/1812.00315.pdf. Accessed 20 July 2020
25. Ecker, U.K.H., Lewandowsky, S., Chang, E.P., Pillai, R.: The effects of subtle misinformation in news headlines. J. Exp. Psychol. Appl. **20**(4), 323–335 (2014)
26. Sloman, S., Fernbach, P.: The knowledge illusion: why we never think alone. Penguin (2017)
27. Berinsky, A.J.: Rumors and health care reforms: Experiments in political misinformation. Brit. J. Pol. Sci. **47**(2), 241–262 (2017)
28. Swire, B., Berinsky, A.J., Lewandowsky, S., Ecker, U.K.H.: Processing political misinformation: comprehending the Trump phenomenon. R. Soc. Open Sci. **4**(3), 160802(2017)
29. Sunstein, C.R., Bobadilla-Suarez, S., Lazzaro, S.C., Sharot, T.: How people update beliefs about climate change: good news and bad news (SSRN Scholarly Paper No. ID 2821919). Social Science Research Network, Rochester, NY (2016)
30. Thorson, E.: Identifying and correcting policy misperceptions. George Washington University. https://www.americanpressinstitute.org/wp-content/uploads/2015/04/Project-2-Thorson-2015-Identifying-Political-Misperceptions-UPDATED-4-24.pdf. Accessed 23 May 2020
31. Schwarz, N., Sanna, L.J., Skurnik, I., Yoon, C.: Metacognitive experiences and the intricacies of setting people straight: implications for debiasing and public information campaigns. Adv. Exp. Soc. Psychol. **39**, 127–161 (2007)
32. Vraga, E.K., Thorson, K., Kligler-Vilenchik, N., Gee, E.: How individual sensitivities to disagreement shape youth political expression on Facebook. Comput. Hum. Behav. **45**, 281–289 (2015)
33. Shin, J., Thorson, K.: Partisan selective sharing: the biased diffusion of fact-checking messages on social media. J. Commun. **67**(2) (2017)
34. Menczer, F.: The spread of misinformation in social media. In: Proceedings of the 25th International Conference Companion World Wide Web. http://gdac.uqam.ca/WWW2016-Proceedings/companion/p717.pdf. Accessed 22 Apr 2020
35. Conover, M., Ratkiewicz, J., Francisco, M., Gonçalves, B., Flammini, A., Menczer, F.: Political polarization on Twitter. In: Proceedings 5th International AAAI Conference on Weblogs and Social Media (2011)

36. Conover, M., Ratkiewicz, J., Gonςalves, B., Flammini, A., Menczer, F.: Predicting the political alignment of Twitter users. In: Proceedings of 3rd IEEE Conference on Social Computing (2011)
37. Conover, M., Gonçalves, B., Flammini, A., Menczer, F.: Partisan asymmetries in online political activity. EPJ Data Sci. **1**, 6 (2012)
38. Tambuscio, M., Oliveira, D.F.M., Ciampaglia, G.L., Ruffo, G.: Network segregation in a model of misinformation and fact-checking. J. Comput. Soc. Sci. **1**, 261–275 (2018)
39. Qiu, X., Oliveira, D.F.M., Sahami Shirazi, A., Flammini, A., Menczer, F.: Limited individual attention and online virality of low-quality information. Nat. Hum. Behav. **1**(7) (2017)
40. Ferrara, E., Varol, O., Davis, C.A., Menczer, F., Flammini, A.: The rise of social bots. Commun. ACM **59**(7), 96–104 (2016)
41. Shao, C., Ciampaglia, G.L., Flammini, A., Menczer, F.: Hoaxy: a platform for tracking online misinformation. In: Proceedings Third Workshop on Social News on the Web (WWW SNOW) (2016)
42. Andrews, C., Fichet, E., Ding, Y., Spiro, E.S., Starbird, K.: Keeping up with tweetdashians: The impact of “official” accounts on online rumoring. In: Proceedings of 19th ACM Conference on Computer-Supported Cooperative Work and Social Computing, pp. 452–465 (2016)
43. Ciampaglia, G.L., Flammini, A., Menczer, F.: The production of information in the attention economy. Sci. Rep. **5**, 9452. https://www.nature.com/articles/srep09452. Accessed 13 Apr 2020
44. Grinberg, N., Joseph, K., Friedland, L., Swire-Thompson, B., Lazer, D.: Fake news on Twitter during the 2016 US presidential election. Science **363**(6425), 374–378 (2019)
45. Guess, A., Nagler, J., Tucker, J.: Less than you think: prevalence and predictors of fake news dissemination on Facebook. Sci. Adv. **5**(1) (2019)
46. Lazer, D.M., Grinberg, N., Friedland, L., Joseph, K., Hobbs, W., Mattsson, C.: Combating fake news: an agenda for research and action. https://shorensteincenter.org/combating-fake-news-agenda-for-research/. Accessed 05 Jan 2020
47. Greenhill, K.M.: Fear and present danger: extra factual sources of threat conception and proliferation. https://ndisc.nd.edu/assets/311237/greenhill_fapd_notre_dame_excerpt_3519.pdf. Accessed 30 May 2020
48. Ratkiewicz, J., Conover, M., Meiss, M., Gonçalves, B., Flammini, A., Menczer, F.: Detecting and tracking political abuse in social media. In: Proceedings 5th International AAAI Conference 2011 on Weblogs and Social Media (ICWSM) (2011)
49. Metzger, M.J., Flanagin, A.J., Medders, R.B.: Social and heuristic approaches to credibility evaluation online. J. Commun. **60**(3), 413–439 (2010)
50. Pennycook, G., Bear, A., Collins, E., Rand, D.G.: The implied truth effect: attaching warnings to a subset of fake news stories increases perceived accuracy of stories without warnings. https://papers.ssrn.com/sol3/papers.cfm?abstract_id=3035384. Accessed 27 Mar 2020
51. Ecker, U.K.H., Lewandowsky, S., Tang, D.T.W.: Explicit warnings reduce but do not eliminate the continued influence of misinformation. Mem. Cognit. **38**(8), 1087–1100 (2010)
52. Silverman, C.: Fake news expert on how false stories spread and why people believe them. NPR. http://www.npr.org/2016/12/14/505547295/fake-newsexpert-on-how-false-stories-spread-and-why-people-believe-them. Accessed 28 Dec 2019
53. Hindman, M., Barash, V.: Disinformation, ‘Fake news’ and influence campaigns on Twitter. Knight Foundation Report. https://kf-site-production.s3.amazonaws.com/media%5Felements/files/000/000/238/original/KF-DisinformationReport-final2.pdf. Accessed 09 Feb 2020
54. Varol, O., Ferrara, E., Davis, C.A., Menczer, F., Flammini, A.: Online human bot interactions: detection, estimation and characterization. In: proceedings International AAAI Conference on Web and Social Media (2017)
55. Hoxha, A., Hanitzsch, T.: How conflict news comes into being: reconstructing ‘reality’ through telling stories. Media, War Confl. **11**(1), 46–64 (2018)
56. Wu, L., Morstatter, F., Carley, K.M., Liu, H.: Misinformation in social media: definition, manipulation and detection. https://www.kdd.org/exploration_files/8._CR.10.Misinformation_in_social_media_-_Final.pdf. Accessed 30 June 2020

57. Ecker, U.K.H., Lewandowsky, S., Swire, B., Chang, D.: Correcting false information in memory: manipulating the strength of misinformation encoding and its retraction. Psychon. Bull. Rev. **18**, 570–578 (2011)
58. Johnson, H.M., Seifert, C.M.: Sources of the continued influence effect: when misinformation in memory affects later inferences. J. Exp. Psychol. Learn. Mem. Cogn. **20**, 1420–1436 (1994)
59. Begg, I.M., Anas, A., Farinacci, S.: Dissociation of processes in belief: source recollection, statement familiarity, and the illusion of truth. J. Exp. Psychol. Gen. **121**, 446–458 (1992)
60. Dechene, A., Stahl, C., Hansen, J., Wanke, M.: The truth about the truth: a metaanalytic review of the truth effect. Pers. Soc. Psychol. Rev. **14**, 238–257 (2010)
61. Wang, W.C., Brashier, N.M., Wing, E.A., Marsh, E.J., Cabeza, R.: On known unknowns: fluency and the neural mechanisms of illusory truth. J. Cogn. Neurosci. **28**, 739–746 (2016)
62. Diakidoy, I.A.N., Mouskounti, T., Fella, A., Ioannides, C.: Comprehension processes and outcomes with refutation and expository texts and their contribution to learning. Learn. Instr. **41**, 60–69 (2016)
63. Guzzetti, B.J.: Learning counter-intuitive science concepts: what have we learned from over a decade of research. Read. Writ. Q. **16**, 89–98 (2000)
64. Connolly, K., Chrisafis, A., McPherson, P., Kirchgaessner, S., Haas, B., Phillips, D., Hunt, E., Safi, M.: Fake news: an insidious trend that's fast becoming a global problem. The Guardian. https://www.theguardian.com/media/2016/dec/02/fake-newsfacebook-us-election-around-the-world. Accessed 15 Jan 2020
65. Lavoipierre, A.: Fake news named 2016 Word of the Year by Macquarie Dictionary. ABC News. http://www.abc.net.au/news/2017-01-25/fake-news-named-2016-word-of-the-year/8211056. Accessed 23 Dec 2019
66. Boaden, H.: BBC—The editors: impartiality is in our genes. https://www.bbc.co.uk/blogs/theeditors/2010/09/impartiality_is_in_our_genes.html. Accessed 28 June 2020
67. Tandoc, E.C.J., Lim, Z.W., Ling, R.: Defining "fake news." Digit. J. **6**(2), 137–153 (2018)
68. Fletcher, R., Nielsen, R.K.: People don't trust news media—and this is key to the global misinformation debate. Understanding and addressing the disinformation ecosystem. https://firstdraftnews.org/wp-content/uploads/2018/03/The-Disinformation-Ecosystem-20180207-v2.pdf. Accessed 17 June 2020
69. Goel, S., Anderson, A., Hofman, J., Watts, D.J.: The structural virality of online diffusion. Manage. Sci. **62**(1) (2015)
70. Linden, S., Leiserowitz, A., Rosenthal, S., Maibach, E: Inoculating the public against misinformation about climate change. Wiley Online Library. https://onlinelibrary.wiley.com. https://doi.org/10.1002/gch2.201600008. Accessed 22 Mar 2020
71. Allcott, H., Gentzkow, M., Yu, C.: Trends in the diffusion of misinformation on social media. Res. Pol. 1–8 (2019)
72. Metz, C.: Facebook adds A.I. labs in Seattle and Pittsburgh, pressuring local universities. New York Times. https://www.nytimes.com/2018/05/04/technology/facebook-artificial-intelligence-researchers.html. Accessed 22 Apr 2020

Chapter 15
Fake News Detection Techniques for Social Media

Akrati Saxena, Pratishtha Saxena, and Harita Reddy

Abstract The advent of numerous social networking websites in the twenty-first century has provided an easy outlet for people across the globe through widely available devices such as smartphones. While this has empowered people belonging to different walks of society to post content on topics ranging from current affairs to history, it is not easy to ascertain the content's veracity. Traditional news media has experts in the domain who have the ability to fact-check the content presented in the news. However, given the enormous amount of social media posts every day, an average human being who is exposed to the content faces difficulty in differentiating false information from real. This has brought researchers' interest in the automated detection of fake news. In this chapter, we will discuss the features that are used to identify fake news and different categories of fake news detection techniques. We also outline the datasets available for fake news detection and provide the directions for further reading.

15.1 Introduction

Social media has transformed the way users communicate by enabling users to post content that can be visible to selected users or the entire world. Social networking websites, such as *Twitter*, have millions of users posting a variety of content every day. In fact, Mendoza et al. [1] were able to index around 4.7 million local posts pertaining to the 2010 Chilean earthquake based on Twitter hashtags. While the ease

A. Saxena (✉)
Department of Mathematics and Computer Science, Eindhoven University of Technology, Eindhoven, The Netherlands
e-mail: a.saxena@tue.nl

P. Saxena
G. L. Bajaj Institute of Technology and Management, Greater Noida, India
e-mail: pratistha.saxena@glbitm.ac.in

H. Reddy
Surat, Gujarat, India

A. Biswas et al. (eds.), *Principles of Social Networking*,
Smart Innovation, Systems and Technologies 246,
https://doi.org/10.1007/978-981-16-3398-0_15

of using social media can be essential for spreading information in situations such as disaster response, the question that arises is how much of the content posted online is verified and factually correct. False information can have severe negative outcomes, such as the creation of panic.

Fake news is commonly defined as untrue content that is created with the intention of misleading people and convincing them to believe it [2]. In their definition of fake news, Shu et al. [3] highlighted two features of fake news, (i) the news contains content that can be verified as false, and (ii) it is intentionally created for misleading readers. False information, in general, can be categorized into two types (i) opinion-based, in which there is no single ground truth that a person is providing false opinions, and this mainly happens in the case of fake online reviews, and (ii) fact-based, in which there is a ground truth which has been contradicted by the information, and this covers fake news, rumors, and misinformation [4].

In the nineteenth century, the availability of cheap printed news did lead to an avenue for the spread of partisan news. However, with the advent of social media today, content can be easily shared with no editorial judgment, fact-verification, or checking by any third-party [5]. This, in addition to the increased consumption of news from social media, increases people's exposure to false information. A study on Canadian social network users almost a decade back showed that two-fifths of the users consumed news from the people they followed on platforms such as Facebook to keep up with a variety of news [6]. Another study[1] in 2017 showed that around two-thirds of the adults in the United States consumed news from social media platforms.

With news sources going online, fast sharing of the news is incentivized by revenue through views and clicks on news articles by the readers, possibly bypassing proper verification of the information [7]. The 'Clickbait' headlines are eye-catchy headlines created for news articles with the intention of enticing users to click on the link and read more [3]. Some types of clickbait headlines exaggerate the content of the news articles or are even associated with factually incorrect articles. Another motivation for generating fake news is to promote an ideology, mainly during elections, by spreading the news in favor of a particular candidate or party. Bots or automated social media accounts can speed up the spread of fake news. Bots were used for sharing a significant volume of content related to politics during the USA presidential elections 2016 and France elections, 2017 [8, 9].

In this chapter, we will discuss techniques for detecting fake news, which can be divided based on three aspects. The first aspect is using information from the news and the news source to determine if the news is fake. Several methods exist, including ascertaining which news sources or users on social media are likely to post fake or biased news [10, 11], understanding social networks of fake accounts [12], and analyzing linguistic [13, 14] and multimedia [15, 16] differences of fake news content from that of real news. The second aspect is understanding the response to the news, including the expression of doubts by readers and how fake news is propagated

[1] https://www.reuters.com/article/us-usa-internet-socialmedia/two-thirds-of-american-adults-get-news-from-social-media-survey-idUSKCN1BJ2A8.

differently compared to legitimate news. The third aspect is checking claims using knowledge sources to verify them. Fake news detection can be challenging in many ways. Social media posts or news websites may contain accompanying images and other multimedia, and it is easy to tamper and photoshop images [2, 7]. On social media, it is possible to create fake accounts impersonating human users.

The chapter is structured such that in Sect. 15.2, we describe techniques to identify features and detect fake news, including classification models and datasets. This is followed by the conclusion and future directions for research in this domain. In Sect. 15.4, we suggest some further readings for the readers interested in exploring more about this domain.

15.2 Fake News Detection

Rubin et al. [17] suggested three classes of fake news detection based on three types of fake news: (i) dishonest reporting by journalists through fabrication, seen in tabloid and yellow journalism, (ii) Internet hoaxes, and (iii) fake news created for humor and satire, often in the form of news parody. While humor is different from serious fabrication, and the readers might identify the intent of satire, identifying other intentionally misleading news online as quickly as possible is essential to prevent people from being misled into believing them as the truth. An example of a tool for automated monitoring of false information online is *Hoaxy*, which crawls data from fact-checking websites and tracks Twitter to extract instances of sharing of the news [18]. In the upcoming section, we discuss the categories of features used for differentiating fake news from legitimate news.

15.2.1 Fake News Detection Features

One of the important features for determining the veracity of news is the content in the original piece of news and the responses by social media users to it. The features of the news source, including the website that published the news or the social media user who posted the news, as well as the features of users who disseminate the news, are other useful indicators for detecting fake information.

15.2.1.1 Content-Based Features

A news article contains a headline, which though short, may carry cues to indicate that the content is misleading. The headline may be written in a catchy or sensationalist manner in the form of a clickbait to attract users to the website [19]. The main body of the article contains a detailed explanation of the news or the claim and provides longer text than the headline for linguistic analysis. A news piece can also contain multimedia such as images, and social media users can attach multimedia to their microblog posts. It is observed that in the case of misinformation, users mostly use images that are not exactly from the news event [20]. The news articles are shared on social media through external links, or the news content can directly be posted in a microblog in brief. The sharing of information on social media further adds challenges as the user can share the information in the form of short posts and might not include the link to the published article. However, the microblog content features can be supplemented with the analysis of comments and replies from other users to obtain the supporting or conflicting opinions offered on the shared news.

Linguistic features include textual features based on characters, words, sentences, and the document as a whole. Commonly used linguistic features in natural language processing (NLP) are lexical and syntactic features [3]. Examples of generic linguistic features include frequency of stop-words, number of exclamation and question marks, parts of speech (POS) tagging, readability of text, usage of function words, and so on. There are certain social networking platform-specific features, including the number or proportion of external links (URLs) and hashtags (which are trending topic names prefixed with a '#'). Some other textual features include *polarity*, which captures the positive or negative sentiment in the text, *subjectivity*, and *disagreement*. There are visible linguistic differences in the writing style of deceptive content [21], and fake news content can also lack objectivity [22]. Discourse markers are used to ascertain the amount of confidence in a news piece [23]. In Table 15.1, we outline some of the linguistic features used in document classification research.

15.2.1.2 User-Based Features

User-based or account-level features include the features of the user who creates a particular post and the users who react to it further by liking, sharing, commenting, or replying. These features include how long the user is registered on the platform, the number of microblogs posted by the user, follower and friend count, is the user verified, does the user have a bio, and so on [28]. Vosoughi et al. [30] identified six user features for users engaged in rumor propagation, (i) how controversial the user is, (ii) originality of the user's tweets, (iii) the user's account is verified or not, (iv) user's engagement, (v) user's role, and (vi) user's influence. User-level features are also used to identify the groups of people spreading misinformation on the platform. The group-level users' features include individual-level features aggregated over a group of users, based on the premise that groups of users who spread fake news can be identified through some unique features [3].

15.2.1.3 Network-Based Features

Network- or graph-based features consist of the network of connections of users engaged with the news on social media, as well as the *diffusion network*, which highlights the propagation pattern of news on social media [31].

Table 15.1 Linguistics-based features

Feature set	Purpose	Features
Zhou et al. [24]	Detecting deception on computer-mediated interaction	**Quantity**: word, sentence, modifier, verb, noun phrase **Complexity:** average clause and punctuation count, average word, sentence, noun phrase lengths **Non-immediacy**: usage of passive voice, modal verbs, objectification, generalizing terms, uncertainty, self-references, group references, and other references **Expressiveness**: emotiveness [25] **Diversity**: lexicon and content word diversity, redundancy **Specificity**: perceptual and spatio-temporal content, positive and negative affect **Informality**: spelling error ratio
Brennan and Greenstadt [26]	Authorship attribution	Unique words' count, Gunning Fox readability index, another alternative readability index, character count excluding whitespaces, lexical density, sentence count, mean of syllable count per word, mean sentence length
Writeprints feature set [27]	Authorship attribution for online messages	**Lexical features**: character-level features, such as count of characters, uppercase characters, digits, tab spaces, white spaces, and alphabets. Frequency of letter/special character. Word-level features including word count, short words' count, average word, and sentence lengths, hapax legomena and dis legomena, vocabulary richness measures, distribution of word lengths **Syntactic features**: sentence-level features including punctuation frequency, function words' frequency, etc. **Structural features**: line count, sentence count, paragraph count, sentences/characters/words per paragraph, greetings, separators, quoted content and its position, indentation, signature **Content-specific features**: frequency of selected domain-specific words
Castillo et al. [28]	Information credibility on social media	Contained emoticons (smile, frown), positive words count, negative words count, sentiment score, length in terms of words/characters, first- and third-person pronouns, question and exclamation marks, hashtags (#), URLs, and mentions (@)
Horne et al. [29]	Fake news detection	**Complexity**: Gunning Fog, Flesch-Kincaid, and SMOG Readability Indexes, depth of syntax, noun and verb phrase trees, average word length, lexical diversity **Psychology**: sentiment strength, count of analytic, insightful, and causal words, count of words expressing discrepancy, certainty, tentativeness, differentiation, affiliation, power, reward, risk, emotion and personal concern **Style**: word, noun, possessive pronoun count, past tense and future tense word count, interrogatives count, and so on

15.2.2 Fake News Detection Categories

The fake news detection techniques can be organized into three broad categories, which make use of the features discussed above.

1. News content and source characteristics:
   (a) General linguistics: Linguistic features of fake news or claims differ from those of real news [13, 14, 29, 32–34];
   (b) Deception: Fake news may contain deceptive language in which the author tries to hide writing style, shows detachment, or put an extra effort to enforce that the news is true [21, 24, 35];
   (c) Objectivity: Fake news may lack objectivity and be one-sided toward an ideology [22];
   (d) Images and other multimedia: Fake news may contain unrelated, poor quality, repetitive, or tampered images [16, 20, 36, 37];
   (e) Source credibility: Some sources may have produced false content in the past, and some social media accounts may actually be bots [10, 38].
2. Social context:
   (a) Doubt and Opposition: Fake news may attract comments from other social media users that express doubts, incredulity, or even opposition [39–41];
   (b) Propagation patterns: False news may spread faster, farther, and deeper [42];
   (c) The users propagating fake news may be less credible or highly controversial [30, 43, 44].
3. Fact-checking: Verification of a claim by using credible knowledge sources as ground truth [45, 46].

15.2.2.1 News Content and Source Characteristics

News is posted on social networking websites in the form of microblogs, and an external link to a news website may also be shared in the content. One of the main features for determining social media news credibility includes the features of the news article's content shared in the tweet or the source tweet itself. Additionally, the credibility of the source that published the news article or posted the news on social media is also a factor for determining the credibility of the news. In this section, we discuss fake news detection based on the news content and the credibility of the news source.

General Linguistic Features. Common linguistic features of the news content coupled with textual features specific to a platform such as *mentions* and *hashtags* on Twitter are useful in detecting the credibility of tweets [34, 47]. Qazvinian et al. [32] obtained both high precision and recall with lexical patterns and POS features of the content for rumor identification. In contrast, with platform-specific features such as

Twitter hashtags or URLs, the precision was high, but the recall was low, possibly because a large portion of tweets does not contain hashtags or URLs. Using features of the tweet, such as swear words count, count of words expressing positive/negative sentiment and emoticon count, and the features of the user who posted the tweet such as user's Twitter age and friend/follower count, Gupta et al. ranked tweets related to major events such as UK riots 2011 based on their credibility using both Support Vector Machine (SVM) ranking algorithm and pseudo-relevance feedback mechanism [33, 48]. They noted that unique character count was a good indicator for credibility, possibly due to informative and connected tweets with hashtags, URLs, and mentions having more characters. Boididou et al. used both tweet-based and user-based features and noted that even without using language-specific features, their models performed well for some languages. However, the low success with French showed the need for exploring some language-specific features as well. They achieved an F1 score above 0.93 on both MediaEval 2015 and 2016 datasets [34, 36].

In emergencies such as riots or natural disasters, it becomes crucial to evaluate the credibility of news on social media to prevent panic caused by false information. Based on the automated detection of such an emergency scenario, Xia et al. [49] used the Bayesian Network Classifier [50] with content-related features along with author-related, topic-related, and diffusion-related features of tweets to determine their credibility, and their model was able to classify credible tweets better than non-credible ones. Linguistic features of tweets were also found to be useful for identifying fake images about 2012 Hurricane Sandy circulated on Twitter [47]. The authors achieved 97% prediction accuracy in differentiating tweets with fake image URLs from the genuine ones using J48 Decision Tree classifiers on common linguistic features of tweets, including text length, question mark count, exclamation mark count, word count, uppercase character count, and types of the pronoun as well as hashtag count, URL count, and mention count. For real-time credibility detection, Gupta et al. created a system called *TweetCred* for determining the credibility score of tweets on a point-scale in real time, using the tweet content and some other features [51]. Some of these features include tweet metadata such as geo-coordinates and time since the tweet, author metadata such as friends/followers count and Twitter age of the user, tweet network information such as retweets' count, and information related to links in the tweet including WOT[2] score of URLs linked in the tweet. Based on users' feedback, 63% of users either concurred or disagreed only by the difference of 1–2 points with their generated scores on a 1–7 scale.

The analysis of psycho-linguistic features of news using Linguistic Inquiry and Word Count (LIWC) [52] has shown that while fake news has a higher amount of perceptual, social, and positive words, legitimate news contains words that express insights and other cognitive processes [53]. With the complete LIWC feature set and a linear SVM classifier, the authors were able to obtain an accuracy of 70% on a dataset created via crowdsourcing. O'Brien et al. [14] observed the usage of strong words for capturing people's attention and exaggeration in fake news. Rashkin et

[2] Reputation score in Web of Trust: https://www.mywot.com/.

al. [54] showed that (i) fake news tends to have more usage of the first-person and second-person pronouns, possibly indicating writing with imagination, and also a greater amount of exaggerating words such as superlatives, and (ii) trustworthy news is more evidence-backed with concrete figures and less vagueness. They further try to differentiate hoaxes, satire, and propaganda in the category of untrustworthy news. They observed that different types have different properties; for example, propaganda news contains more superlatives as compared to hoaxes.

Horne et al. [29] observed that satire shares more similarity with fake news as compared to real news in terms of psychology, style, and complexity. They were able to differentiate satire versus real news with a 91% accuracy, whereas satire versus fake news with 67% accuracy. Fake news has also been shown to contain more redundant content and lesser quotes, possibly due to the lack of real content and quotations as evidence to write about [29, 55]. Fake news can also be differentiated based on its headlines. Fake news headlines may have a large amount of content squeezed into a single sentence, copious usage of capital letters, and skipping of stop-words to mention as many entities as possible [29]. Horne et al. [29] achieved an improved accuracy of 78% on using title text content as compared to 71% on body text using a linear kernel SVM on dataset pertaining to political news. This can be used to target people who do not read beyond headlines of news without looking for any concrete evidence, arguments, and logic in the actual content of the article.

In text documents, the content can be represented in the form of a word embedding matrix, constructed using word-vectorization techniques. One of the simplest models is Bag of Words (BOW), in which the count of occurrences of words in a text document is used to construct a vector. Term Frequency-Inverse Document Frequency (TF-IDF) is another method that considers the importance of the token with respect to the document instead of occurrence count. More complex models include Word2Vec (consisting of Skip-gram and Continuous Bag of Words) [56] and FastText [57]. The idea behind Word2Vec is to learn the representation of words such that similar words are located in proximity in the vector-space representation. Words are encoded as vectors using neural networks on the basis of the relationship of the words with their neighboring words in the documents. In FastText, words are split into n-grams before the creation of vectors.

In the past few years, deep learning approaches also have been used for fake news detection. These approaches are mainly used with word vector representations of the news text. Neural networks can be used to extract powerful features from the text without the need for hand-crafted features [58]. On the LIAR dataset, Karimi et al. [58] were able to achieve an improved accuracy using multi-class classification over the baseline methods by incorporating multiple sources using Convolutional Neural Networks (CNNs), including the actual statement for which the credibility has to be determined, the speaker's profile and history of statements, and reports by trusted sources. Singhania et al. [59] proposed the 3HAN model, which is a neural network model with a 3-level hierarchy with attention layers to give different weights to different parts of an input news article. The three levels correspond to the encoding of word sequences, sentences, and headline body, and through this neural network, a news vector is created to classify the articles. The authors achieved an accuracy of

96.77% on a dataset based on sites marked as fake by PolitiFact, and those marked as genuine by Forbes. One of the bases for this model is that certain words and sentences are more important than others for document class determination. For input, the word embeddings of the text were obtained using the GloVe model, i.e., a popular unsupervised machine learning method to create word-vector representations [60]. Rashkin et al. [54] used a Long Short-Term Memory (LSTM) network with word sequence as input on the PolitiFact dataset,[3] and the LSTM outperformed other models such as Naive Bayes for dual-class classification. The further addition of LIWC features enhances the performance of other classifiers. However, a problem associated with deep learning is its *black box* nature, which gives rise to the need for more transparency in terms of visualizing the textual patterns that are more useful for classification [14].

Document summarization techniques are used for the automatic generation of a summary of documents, i.e., a short description consisting of the main points of a document. Shim et al. [61] evaluated the performance of fake news detection models when complemented with extractive document summarization techniques. They used Lexrankr, a summarization system, to extract summaries consisting of three sentences from news articles. Based on the TF-IDF word embeddings, the performance of both full-text and summarized text for fake news detection was evaluated using machine learning models. While summary-based models did not show any notable performance differences compared to full-text-based models overall, however, with Logistic Regression, summarized text performed better. The best performing classifier SVM gave equal validation accuracy of 74% with both full-text and summary models.

Deception Detection Methods. Fake news detection is closely linked with the identification of deceptive language. Deceptive language has considerable differences in linguistic features; for instance, there are observed more positive sentiment, complexity, and usage of brand names in deceptive online travel reviews [13]. Deceptive messages are sent intentionally to lead to untrue conclusions, but if the sender sends it unknowingly without the intention of deceit, it may not be considered as deception [24]. In the early research in automated deception detection, Zhou et al. [24] did an analysis of linguistic-based cues in computer-mediated communication involving both truthful and deceptive messages between volunteers. It was observed that deceptive communication contained a higher number of words, verbs, and sentences, possibly because of the need to convince the receiver that the information was true. Deceptive content also contained lower lexicon and content word diversity and used non-immediate language, for example, fewer self-references [24, 62]. Zhang et al. [63] used similar linguistic features and applied feature selection to get the most differentiating features on deceptive and non-deceptive Chinese texts. In another work, human participants were asked to give truthful and deceptive statements; and based on the LIWC word class distributions, it was inferred that words expressing certainty were higher in deceptive statements possibly due to the need to emphasize that the statement is true, while words expressing insight were higher in

[3] www.politifact.com/.

truthful statements [64]. It was also found that the human subjects tried to detach themselves from lies in deceptive statements by not using words related to self.

Deception and fraud in the text can be determined based on the writing style. When the author of an article tries to hide her writing style, certain distinct linguistic features of the text have been observed to be different. Afroz et al. [21] highlighted that the authenticity of an online post is dependant on the authenticity of the source of the post. The short length of some of the online news may make it challenging to identify rich features linked to writing style as compared to the conventional long documents [65, 66]. Nevertheless, a wide variety of stylistic features have been leveraged to attribute authorship in short online documents. Lexical features of the text include character-level and word-level features, whereas syntactic features can discriminate the writing style of authors based on how the author organizes her sentences, i.e., the sentence-level features of the text [27]. Structural features are even higher-level features that capture the overall organization or layout of a piece of text.

A combination of lexical, syntactic, structural, and content-specific features (such as keywords) may be used to identify the author of online messages. Afroz et al. [21] used feature sets for authorship attribution including the Writeprints feature set [27], and Brennan and Greenstadt's feature set [26], as well as other features pertaining to lie detection [67, 68]. They successfully detected deception in online documents with the best overall F1-score of 96.6% with the Writeprints feature set using SVM. Writeprints feature set was also found to be effective for fake news detection in [55]. Feng et al. [69] used deep syntactical features in the form of Probabilistic Context-Free Grammar (PCFG) for deceptive review detection. The authors obtained 91.2% accuracy on a dataset of hotel reviews that is better than the accuracy achieved through shallow lexical and syntactic features.

In a coherent text, unlike a collection of individual sentences, there exist functional relationships between different parts of the text, which can be modeled in the form of a rhetorical structure. Rubin et al. used *rhetorical structure theory* [70, 71] coupled with modeling in vector space to classify the text as truthful or deceptive [35]. Initially, two clusters for deceptive and non-deceptive news were computed, and after this, the incoming news was assigned labels based on distance calculation. Using this approach, the authors were able to achieve around 63% accuracy. However, using predictive modeling, they obtained 56% accuracy on the test set. This screening approach can help in identifying candidates for further fact-checking.

Satire is another form of deceptive news and may contain cues that make it obvious that the news is untrue. Rubin et al. [72] used the SVM classifier to classify satirical news using features such as (i) *absurdity*, which is higher when the last sentence of the news introduces certain entities completely unrelated with the rest of the news, (ii) *humor*, identified through minimum relatedness between the first and last sentences of the news, (iii) grammar-related features including POS, (iv) greater usage of punctuation in satire due to complex sentences with multiple clauses, and (v) negative affect words.

Objectivity Detection. An aspect of writing style analysis that can be taken into consideration for text classification is to study the polarity of the text to determine if it is one-sided or biased. Linguistic cues have helped uncover bias in Wikipedia articles, with *framing bias* identified through the usage of subjective words and one-sided terminologies, and *epistemological bias* through features such as usage of factive and assertive verbs [73]. Potthast et al. [22] explored the relationship between fake news and *hyperpartisan* news by analyzing the news from left-wing and right-wing sources against the mainstream sources in the Buzzfeed dataset.[4] They used the concept of *Unmasking*, which had been previously proposed for the purpose of verifying authorship of text [74]. Unmasking helps in dealing with the difficulty that arises in authorship verification when two different works by the same author have a small set of features that are different due to the difference in the genre or theme of the works or due to intentional masking of the writing style by the author in one of the works. The idea behind unmasking is that once these small sets of distinguishing features are removed, texts by the same author will be difficult to differentiate in terms of features.

Potthast et al. [22] used both common linguistic features, including stop-words and n-grams, as well as some domain-specific features. They noted that it was possible to differentiate hyperpartisan news from balanced news. They also observed that left-wing and right-wing hyperpartisan news had certain commonalities in their style. Though the F1 score was not good by only using the style features for fake news detection, the authors proposed that their experimental results could be used for preliminary screening for fake news detection. Determining news with a partisan tinge can also be done with the help of techniques to tag news sources or users based on their political alignment or affiliation. This has been done through the analysis of usage of specific political hashtags by users on Twitter because hashtags can indicate the political affiliations of the users [75]. The tendency of publishers with partisan attitudes to create articles with fake news has been considered as one of the features in other works [44].

Semantic analysis of fact candidates, i.e., statements for which veracity has to be ascertained, was done in FactChecker to determine whether a given text is objective or opinionated [76]. Nakashole and Mitchell modeled candidates for fact-checking in terms of subject-verb-object (SVO) triples. The semantic assessment of the SVO triple form of the fact candidates involved (i) finding what type of subject (S) and object (O) will be present for the given verb (or verbal phrase), (ii) determination of the cardinality of the relationship between the subject (S) and object (O) for a given verb (V), and (iii) finding other synonymous verbs that can take the place of a given verb (V). This analysis is used to create potential alternative fact candidates, and the given fact candidate can be ranked against them. With the accuracy defined as the probability that the *believability score* of a truthful claim is greater than a false one, the proposed *FactChecker* achieved an accuracy of 0.90 on fact candidates retrieved from Wikipedia.

[4] Refer https://www.buzzfeednews.com/article/craigsilverman/partisan-fb-pages-analysis and https://github.com/BuzzFeedNews/2016-10-facebook-fact-check.

Visual and Multimodal Techniques. News can contain multimedia, such as images, video, and audio. The features of images in the news can help in further improving the accuracy of text-based fake news detection. Fake news may contain images that are not related to the event being described in the news, or images may even be manipulated by adding or removing entities, splicing, and so on [36]. Deep learning techniques have been successful in learning latent features from both images as well as text. Jin et al. [77] used the Recurrent Neural Network (RNN)-based model using an attention mechanism for capturing the correlation between visuals and text. The proposed model is trained based on multimodal input of news content, attached visuals, and some social context. On the Weibo dataset introduced by the authors, the model obtained an accuracy of 78.8% for identifying rumors and non-rumors. Khattar et al. [78] proposed a neural network-based multimodal fake news detection model using both visual and textual features. Based on the shared representation obtained through auto-encoder, the proposed model outperforms single modality models for fake news detection on Twitter dataset [15], obtaining an accuracy of 74.5%, while the accuracies obtained through textual and visual models independently were 52.6% and 59.6%, respectively.

Fake news may contain tampered images that can lead to poor quality or misleading images that are unrelated and possibly taken from some other news. Qi et al. [20] proposed a neural network-based fusion model to leverage frequency domain (physical characteristics) and pixel domain (semantic characteristics) of images associated with the news for fake news detection. This method of visual feature extraction, when used with previously proposed multimodal models, showed an improved accuracy. Images in real news have a tendency to contain more faces, while fake news may contain unrelated images. Another important feature is image resolution, as real news contains higher resolution images as compared to fake news. Yang et al. [37] proposed the TI-CNN model that uses CNNs with both explicit and latent features from raw images and textual content. The proposed model outperforms other baseline methods on Kaggle dataset [79] with a high precision of 92%.

Jin et al. [16] noted that microblog content having fake news contains repetitive images. In contrast, the content associated with real news contains more diversity in images, as in the case of a real event, there would be many legitimate images available. Real news would also contain more images compared to the same amount of fake news tweets. Based on this, they used visual features along with statistical features for verifying news. One of the features is the *Visual Clarity* score, which indicates how different the distribution of images is between a given news event and the complete collection of news events. The images in real news threads are from various sources, unlike fake news threads, which get images from less diverse sources; therefore, real news posts' score is lower for this feature. They were able to improve accuracy by over 14% for news verification using non-image and image features as compared to the accuracy obtained by using the 11 top features used in [39]. Another issue on social media is that outdated, inaccurate, and manipulated images are often kept circulating on social media. In another work, Jin et al. [80] used a large weakly labeled auxiliary dataset along with the training set to learn image representations using CNN to differentiate credible images from fake ones.

Extracting features from the RGB (Red, Green, Blue) channels of the pixel domain in images using CNNs have shown promising results for detecting fake images that were created using Generative Adversarial Networks [81, 82].

Source Credibility. The credibility of news is correlated with the credibility of its source. For example, the source may have produced factually incorrect content in the past, where the source can be the website that published the news or the social media handle that posted the news [10]. For fake news detection in articles on the Web, textual features proved to be good indicators of factuality. More credible websites have Wikipedia articles on them, and the textual features of the articles published by a website indicate its bias and factuality. In [10], the use of Wikipedia features led to an accuracy of 62.29% for factuality, and excluding Wikipedia-related features led to the highest decrease in performance. It is also noted that the more credible tweets contain URLs to the most popular domains on the Web [28].

It is observed on Wikipedia that the creators of legitimate articles are mostly established users, while articles identified as hoaxes are mostly created by relatively new accounts [83]. The credibility of social media user accounts that created or posted the news is an important factor for the determination of news credibility. The features of a user account, such as profiling the account in terms of its geo-location, when the account was created, and tweet count, can be used to determine if the account is suspicious or a bot [38, 84, 85]. Gurajala et al. [86] analyzed the data comprising Twitter profiles and proposed a method to detect fake profiles using pattern-matching on the display-names and the evaluation of the update times of tweets. Though bots can be used for social benefits, they can also be used for purposes such as spreading false information and also impacting political debates [38]. The tweets posted by bots are prominently through API-based tools, unlike humans who post content on Twitter by themselves using the web or mobile applications [87]. The temporal behavioral patterns of automated bots are different from those of credible users. The bot activity may also be identified using the distribution of time gaps between retweets and the number of times the account retweets a specific URL [88]. Bots normally have similar activity levels throughout the week, unlike normal human users [87]. Also, unlike non-celebrity human users who have friend count close to follower count, bots show a tendency to add a large number of friends but end up having fewer followers, due to which their *account reputation* is lower [87]. Sentiment-based features have shown that human users express stronger positive or negative sentiments as compared to bots, and bots are less likely to change their sentiments on a particular topic [89]. Hu et al. [12] showed that the social network information of users represented as adjacency matrices is useful in detecting spammers on microblogs.

Researchers have also worked on identifying credible sources of information about specific topics on Twitter. The results have shown that it depends on the social network structure of the source account and how strongly the source's content is related to the domain [90]. The credibility of a particular news source or publisher can be ascertained from their viewpoint (leaning toward left or right), the expertise of the source on the topic, and the news format (research, editorial, and so on) [23]. Long et al. obtained 14.5% improvement in the accuracy over the baseline methods on the LIAR dataset [11] by incorporating speaker information including the party they were affiliated with, job, title, and a record of any inaccurate claims made by the speaker in the past [11, 91]. Yang et al. [92] incorporated a new feature called the *client program* used to create the microblog, in addition to the already existing content-based, user-based, and propagation-based features for rumor detection on Sina Weibo. The authors found that if a non-mobile client program was used to create a microblog about an event that happened in a different country, the chances of being a rumor are high. Adding the client program and event location features to the already popular account-based features, such as if the account is verified or not, follower count, and so on, improved the accuracy from 72.6 to 77.4%. In detecting tweets containing fake images, the features of users who posted the tweet has been found to perform poorly [47].

15.2.2.2 Social Context

News shared as a microblog on a social networking platform leads to the engagement of other users with the news. Users may like, comment, or share the post further to their 'followers' or 'friends.' The features of the users who respond to or engage with the posted news, the network of interaction between the users, the type of response given by the users, and the diffusion pattern of the news on social media are used to identify the differences in the social context of fake news and real news. Fake news posts may lead to responses expressing incredulity [39, 41, 93]. It is observed that the fake news propagates differently and diffuses at a higher speed with a wider reach [42]. Social media websites may utilize crowdsourcing to detect fake news by requesting users to identify and flag fake news appropriately. In such a case, the credibility of users who mark the news is also essential and can be determined based on the user's history of flagging news [94].

In this section, we discuss the different aspects of the social context of fake news in detail, starting with the detection of fake news based on its propagation patterns.

News Propagation. News propagation-based features, including the number of retweets, thread lifetime, number of comments on the post, and retweet tree depth, have been used to ascertain the credibility of news on social media [28, 92, 93]. Propagation features from the retweet tree achieved a high true positive rate for fake news, thus highlighting the importance of graph-based features for detecting non-credible news [39]. It was also noted that while propagation trees of newsworthy events/topics had a great depth, the tweets having a high width at a particular level, i.e., a large number of re-posts at a particular level in the propagation tree, tended to be more credible.

Suzuki proposed a way to compute the credibility of messages on social media such as Twitter using the re-posting or retweeting behavior of other users with respect to the message [95]. This was based on the premise that a highly credible message is often re-posted with only slight modifications so that the original message remains intact. In contrast, less credible messages are unlikely to be re-posted, and even if they are re-posted, users will tend to add their own opinion. Kwon et al. [31] noted that false rumors have a high proportion of singleton tweets, implying that such tweets are ignored by other users. They used structural patterns along with temporal and linguistic features to detect false/unverified rumors, and the Random Forest classifier achieved the best F1 score of 89.3%.

The propagation of a tweet can be represented with a tree structure, with the source tweet being the root and the responses of other users to the tweet being the nodes connected to the source with directed edges. A top-down approach considers the direction of information diffusion for the edges, whereas a bottom-up approach considers the direction of response [41]. This structure-based information has been incorporated into rumor detection using kernel functions by computing the similarity between the trees [96]. RNNs have been used with propagation tree structures to improve the learning to detect rumors [41]. Monti et al. [97] combined heterogeneous features, including those related to the content, social network structure, user's profile, and propagation information using geometric deep learning. They modeled the propagation information in the form of a diffusion tree and were able to achieve high fake news detection accuracies, showing that such news could be flagged at the early stages of propagation. A graph-kernel-based SVM model using the propagation trees, content, and user-based features showed an improved rumor classification accuracy on Sina Weibo [98]. Another study compared the spread of fake and true news and showed that the retweeting of fake news is more probable as compared to real news as fake news contains seemingly 'novel' content. The fake news retweet cascade, therefore, reaches greater depth (distance from the source node) and is retweeted by more people at any given depth [42].

The diffusion of rumors from low-degree (degree based on the number of followers) users to higher degree users is one of the important features that has been used for verification of rumors. Kwon et al. [31] studied the classification of rumors on Twitter and found that the fraction of information diffusion from low-degree users to high-degree users had high predictive power. Vosoughi et al. [30] noted that the diffusion from low- to high-degree users is high when the rumor is true. The reason is that high-degree or high-influence users avoid taking the risk of retweeting a rumor from a less influential user without strong reasons for the veracity of information.

Temporal Modeling. Ma et al. [99] worked on detecting rumor events; they do not classify individual microblog posts but capture the microblog posts associated with the events to ascertain if an event is a rumor. The posts are modeled as a time-series, and a Recurrent Neural Network (RNN) model is trained using the TF-IDF values of the terms in the posts batched into intervals. The proposed model achieved the maximum accuracy of 0.91 on the Weibo dataset. Yu et al. [100] used CNNs on the paragraph vectors of microblog posts pertaining to an event that was separated into time windows to detect misinformation versus truth and got an improved accuracy of 0.933 on the Weibo data.

Engaging Users' Features. The analysis of the users who interact with a post on social media helps in determining hoaxes as users tend to prefer and interact with the content that suits their narrative. Tacchini et al. [43] obtained an accuracy greater than

99% on hoax classification using both Logistic Regression and Harmonic Boolean Label Crowdsourcing based on the users who 'liked' the Facebook posts. This work was extended further by supplementing social features with content-based features to account for cases in which news pieces receive very little social engagement [101]. The tweets, retweeted by users who have engaged in sharing or posting rumors in the past, have a higher probability of being a rumor [32]. Shu et al. [44] proposed the TriFN model based on the tri-relationship among three entities—news articles, publishers, and the users who share the articles on social media. One of the bases for this model was the tendency of users with less credibility to share fake news and form clusters with similar users. Vosoughi et al. [30] observed that highly controversial users are more probable to be involved in spreading fake rumors, while true rumors are propagated by relatively more credible users. However, it is expected that fake news will be usually spread by users with low credibility, though in some emergency events, for example, in the Boston Marathon bombings, it was found that verified users with more followers also shared fake news in the initial stages, due to the difficulty in verifying the information [102].

Users' Responses. Fake news may elicit reactions such as skepticism, astonishment, and curiosity among the users [40]. Users may express uncertainty with regards to the information in the news by questioning it [39]. The amount of negation or denial expressed by users in the tweets related to a particular rumor is a good measure for verifying the truthfulness of the rumor [30]. The reaction of users in terms of linguistic features such as the higher amount of negating words used in response to a rumor, along with propagation properties such as the proportion of diffusion from low-degree users to high-degree users in a social network, and temporal properties such as external shock's periodicity have been observed to give high precision for rumor classification [31]. Initial stages of news circulation on social media might provide only limited social context information. However, Liu and Wu [103] focused on detecting fake news in the early stages and were able to achieve above 90% accuracy of classification at the end of five minutes since the spread of news started. They used a CNN-based framework using the features of the content of users' responses and features of the users responding. In addition, they used a position-aware attention mechanism to give a higher weight to specific responses crucial for differentiating the news type.

In the real-time detection of rumor events, Liu et al. [104] noted that the use of belief features, i.e., the features to capture whether users support, deny, or question the news through a rule-based algorithm, is useful for early detection and even better in later-stage detection. Jin et al. [105] used the conflicting opinions offered by the users on social media news to determine its credibility by mining opinions using unsupervised learning and constructing a credibility network with supporting and opposing links between tweets. By modeling credibility propagation as a graph optimization problem, they obtained an accuracy of 84% on their Sina Weibo dataset. Hence, using natural language processing (NLP) in user responses, the controversy created due to the news can be detected.

Understanding the stance of other users based on their posts related to the news as well as the stance of other news sources with respect to the claim is a well-studied problem. Based on users' comments on news or claims, there has been an attempt to identify users' stance on the event. The first stage of the *Fake News Challenge*[5] was detecting the stance, i.e., given a headline (claim) and an article, determine the stance of the article text with respect to the headline by indicating whether the article agrees with the headline or disagrees, discusses the headline, or is unrelated to the headline. One approach uses Bag of Words (BOW) features, such as term frequency and TF-IDF for text representation and a multi-layer perceptron (MLP) as a classifier [106]. Another approach used a Support Vector Machine (SVM) on TF-IDF-related features to determine if the article and headline were related or unrelated, and if related, an LSTM-based architecture can classify the news-headline pair as agrees/disagrees/discusses [107]. A similar approach of determining related/unrelated and then doing the fine-grained classification was used in [108].

Stance detection was initially done in terms of an article headline with respect to a particular claim to determine if the article was supporting, opposing, or just reporting the claim [109]. From a social media perspective, there will be tweets that oppose false rumors (expressing opposing stance) and tweets that confirm the news by supporting it [110]. Such stance verification has also been done in RumourEval task that aimed to identify rumors. It first identifies the stance of users' comments on a post and then determines the truthfulness of the rumor presented by a tweet that is unproven at the point of time the tweet was posted [111–113]. In one of the approaches to these tasks, Chen et al. [113] used a CNN-based framework, in which the tweets were encoded as word-vector matrices. In the first task, tweets were labeled as 'supporting', 'querying', 'commenting', and 'denying', and in the second task they were labeled as 'rumor' and 'non-rumor' [113].

Other Approaches. A newsworthy event has specific details, and these details or sub-topics can be referred to as sub-events. Jin et al. [114] created a hierarchical credibility network with three layers—event, sub-event, and individual messages. The sub-events and the related posts were determined with a clustering approach. They modeled credibility evaluation as a graph optimization problem, in which the credibility is propagated on the graph. The entities in the three layers were connected through weighted links, and the optimization was done based on the concept that those entities connected via high weight links will have similar credibility. Ruchansky et al. [115] proposed a hybrid model by leveraging the features related to the text of the news, response to the news, and users promoting the news without using hand-picked features. The model has three modules, in which the first module consists of the textual and response features, which are captured through RNNs based on the temporal data of users' interaction with an article. The source features are learned through a separate module, and the classification of the article is done through the third module.

[5] http://www.fakenewschallenge.org.

15.2.2.3 Fact-Checking

Fact-checking involves evaluating the truth of claims or statements posted by people, such as celebrities and politicians [116]. CLEF Fact Checking [117–119] involved detection of factually false statements made during presidential election conversations in the USA. One well-known method for fact-checking claims performs a comparison of semantic similarity [120] of statements against an already constructed dataset of fact-checked statements [116]. However, this method cannot be used to verify completely new claims, and hence in such cases, a better approach would be to consider a knowledge base such as Wikipedia to be the source of truth. However, Vlachos and Riedel highlighted that this approach would not work in cases where claims include computations based on the existing data [116].

Ciampaglia et al. [46] performed fact-checking of claims using Wikipedia knowledge graphs to get supporting evidence. The 'Fact Extraction and VERification (FEVER)' challenge involved classifying claims as 'supported', 'refuted', or 'not enough info' based on Wikipedia information [45]. The first step of the majority of entries was the retrieval of relevant documents. It was mainly done through the extraction of named entities or nouns from the claims and trying to fetch matching Wikipedia articles by searching using these keywords. The next step was sentence selection, which involved selecting limited sentences from the retrieved documents, which provided some form of evidence pertaining to the claim. The approaches for this step included computing the similarity of sentences, matching entities, and supervised classification. The final stage involved classification, and different approaches were used for combining the evidence together.

The final score given to the teams in the challenge was based on both classification results and the evidence obtained. The highest scoring team (UNC-NLP) obtained 64.21% FEVER score, and this dataset has also been used in later work. Hanselowski et al. [121] modeled the step of retrieval of documents for fact-checking as an entity linking problem and used modified Enhanced Sequential Inference Model (ESIM) [122] to rank sentences for selection in these retrieved documents and verify the claim. Yoneda et al. [123] also used the Sequence Inference model for learning to determine whether a sentence refutes or supports a statement. Yin and Roth [124] developed a system to identify evidence pertaining to a claim based on wiki pages and determine the truthfulness of the claim on the FEVER dataset [125]. While earlier approaches were mostly pipeline-based approaches in which the evidence was first identified and then the claim was verified, the TWOWINGOS framework models these subtasks in an integrated manner. A model is trained jointly for these two subtasks so that they can complement each other. The idea is that based on a claim, the evidence is identified, and then the appropriate evidence reinforces the truthfulness of the claim [126].

Another technique for fact-checking claims is to look for counter-arguments to the claims made in the news [127]. Fact-checking a news piece with the help of a third party can also be done, but fact-checking every piece of news is expensive. Kim et al. [128] proposed the 'CURB' algorithm to select which news pieces to send for further fact-checking from the set of stories flagged as fake by users. In another work, Hassan et al. [129] determined statements worthy for fact-checking using a model trained on an annotated dataset of the statements made in the previous US presidential debates. Linguistic features of the statements, such as the sentiment, sentence length, TF-IDF embeddings, POS, and entity types, were used in the model.

15.2.3 Classification Models

Fake news detection models are usually binary classification models, i.e., the models predict a binary value indicating whether a piece of news is fake or not. However, binary classification might not always work because some parts of the article may be factually true while some might be fake. To account for such cases where the news can be partially fake, multiple classes can be introduced for the purpose of classification to indicate the extent to which news is fake [58]. The available datasets with multiple classes, indicating the level of truth in the news, include LIAR [11] and Vlachos14 [116]. Similarly, for rumor detection, there can be multiple classes, because in some cases, a rumor may remain unverified even after some time. To account for such cases, fine-grained classes are used, namely false rumors, true rumors, non-rumors, and unverified rumors [41, 96, 130]. The fake news detection problem can also be modeled as a regression problem instead of a classification problem, where the regression model will assign a score indicating the level of truth in the news [76]. However, the majority of the datasets in this domain have been designed for classification, either with binary or multi-class labels, hence leading to a problem of converting these labels into continuous scores [131].

Most approaches to fake news detection rely on training models on a pre-annotated dataset [33, 39, 61, 92]. Deep learning approaches have been used to extract hidden features in vectorized representations of text and images [37]. Convolutional Neural Networks (CNNs) have been used mainly for fake news detection based on news text and images [14, 20, 37, 58]. Recurrent Neural Networks (RNNs) have been used for time-series data [99]. Semi-supervised learning is used for TweetCred [51], which predicts the credibility score of tweets in real time. Ensemble methods, such as Bagging, have been leveraged in addition to a base classifier and led to improved accuracy on feature sets with tweet-based and user-based features [34, 55]. For credibility ranking, the ranking algorithms such as Ranking SVM and AdaRank [132] have been used [33, 51]. Clustering approaches are used for creating clusters of news stories in a vector space based on the similarity of their features and assessing new stories based on their distance, for example, Euclidean distance to the cluster centers [35]. Researchers have proposed several methods to combat fake news once it is detected on a social network [133–135].

Table 15.2 Datasets for fake news detection

Name (language)	Unit	Classes	References
Vlachos14 (En)	Statement	True, mostly-true, half-true, half-false, mostly-false, false	[116]
Twitter15 (En)	Thread	Non-rumors, false rumors, true rumors, and unverified rumors	[96]
CREDBANK15 (En)	Event	Certainly accurate, probably accurate, uncertain, probably inaccurate, certainly inaccurate	[136]
MediaEval16 (En)	Post, images	fake, real	[15]
Kaggle (En): Getting Real about Fake News	Claims	Bias, bs, conspiracy, fake, hate, junksci, satire, state, type	[79]
FNC16 (En) Fake news challenge	(Claim, Article)	Agree, disagree, discuss, unrelated	[137]
PHEME (En)	Twitter rumor story	True, false, unverified	[130]
Twitter17 (En)	Thread	Non-rumors, false rumors, true rumors, and unverified rumors	[96, 99]
Jin17 (Chi)	Tweets with images (Weibo)	Rumor, non-rumor	[77]
Ma17 (En)	Source Tweet (Propagation Tree)	True, false, unverified, non-rumor	[96]
Liar17 (En)	Post	Pants-fire, false, barely true, half-true, mostly-true, and true	[11]
Horne17 (En)	News story (website)	Real, fake, satire	[29]
SemEval17 Task B (En)	Thread	Veracity	[111]
FakeNewsNet (En)	Claims, users, engagement, social links	Fake, true	[3, 138]
Pratiwi17 (Ind)	Article Page	Valid, hoax	[139]
Arabic FNC (Ar)	(Claim, Article)	Agree, disagree, discuss, unrelated	[140]
Fever18 (En)	(Claim, Wikipedia)	Support, refute, not enough information	[45]
Vosoughi18	Tweets	True, false	[42]
CLEF18 (En,Ar)	Claims	Half true, false, true	[141]
Rosas18 (a) (En): FakeNewsAMT	News article	Fake, legitimate	[53]
Rosas18 (b) (En): Celebrity	News article	Fake, legitimate	[53]

15.2.4 Datasets for Fake News Detection

In Table 15.2, we summarize the datasets that have been used in fake news detection studies. Crowdsourcing is a common approach taken for the labeling of datasets. One example of crowdsourcing is collecting social media posts like tweets through Twitter API and asking Amazon Mechanical Turk (AMT) workers to evaluate events' credibility to annotate the dataset [136]. Another approach is to get legitimate news from the news websites and ask AMT workers to create a fake version of them to get enough fake news samples [53].

15.3 Conclusion

Fake News has been shown to be potentially menacing in the online ecosystem, as it can affect many major events outside of it. The proliferation of fake news is a serious problem affecting many societies, and we can expect to see much more work on this topic in the near future. In this chapter, we have provided an overview of the current state-of-the-art techniques and approaches related to various aspects of fake news detection. We have discussed fake news detection techniques based on the news content and source characteristics, the social context, and fact-checking.

Real-world social networks are highly dynamic and growing very fast. Information traverses very quickly on them from one user to another. Researchers are still designing solutions that can be applied in real life for efficiently controlling the inverse impact of fake news on human lives. The amount of content on social media is huge. Hence algorithms should be optimized to be applicable for real-time streaming data. In emergencies like a terrorist attack, fake news can cause irreparable damage, and hence algorithms for fake news detection should be able to do real-time monitoring and flagging as soon as possible.

15.4 Further Reading

We suggest the following readings on this ongoing research topic to get more details from different perspectives. References [142–145] provides a brief introduction of fake news and misinformation. Zannettou et al. [146] is a survey on fake news that talks about how people perceive fake news or false information in the political stage and its propagation and detection. As we discussed, fake news detection is a well-studied area; the following are the surveys on fake news detection [2, 4, 147–149]. Shu et al. [150] discussed fake news detection and mitigation using network analysis techniques. Other works include a survey on automatic rumor detection in microblogs by Cao et al. [151], a survey on the diffusion of rumors on Twitter by Serrano et al. [152], and a survey on rumor detection and resolution by Zubiaga et al. [153]. Shelke and Attar reviewed the source detection approaches for rumors or misinformation; more details are available at [154]. Rehm [155] proposed a hybrid technology infrastructure to empower users to tackle fake news by providing automatic assessments of the content and alternative opinions regarding media consumption. Ndii et al. [156] surveyed the propagation models for rumor spreading; most of the proposed spreading models are based on epidemic models. Wang et al. [157] discussed a wide perspective on social bots and their roles in fake news spreading. Alemanno [158] discussed anti-fake news approaches from the policy-making perspective. Sullivan [159] studied the library and information science (LIS) approaches and their shortcomings for controlling fake news. Other brief reviews on fake news can be seen at [5, 160–163]. Some famous articles on fake news after the USA election are available at [164, 165].

References

1. Mendoza, M., Poblete, B., Castillo, C.: Twitter under crisis: can we trust what we RT? In: Proceedings of the First Workshop on Social Media Analytics, SOMA '10, pp. 71–79. ACM, New York, NY, USA (2010)
2. Parikh, S.B., Atrey, P.K.: Media-rich fake news detection: a survey. In: 2018 IEEE Conference on Multimedia Information Processing and Retrieval (MIPR), pp. 436–441. IEEE (2018)
3. Shu, K., Sliva, A., Wang, S., Tang, J., Liu, H.: Fake news detection on social media: a data mining perspective. SIGKDD Explor. Newsl. **19**(1), 22–36 (2017). September
4. Kumar, S., Shah, N.: False information on web and social media: a survey (2018). arXiv:1804.08559
5. Allcott, H., Gentzkow, M.: Social media and fake news in the 2016 election. J. Econ. Perspect. **31**(2), 211–36 (2017)
6. Hermida, A., Fletcher, F., Korell, D., Logan, D.: Share, like, recommend: decoding the social media news consumer. J. Stud. **13**(5–6), 815–824 (2012)
7. Chen, Y., Conroy, N.K., Rubin, V.L.: News in an online world: the need for an "automatic crap detector". Proc. Assoc. Inf. Sci. Technol. **52**(1), 1–4 (2015)
8. Ferrara, E.: Disinformation and social bot operations in the run up to the 2017 french presidential election (2017). arXiv:1707.00086
9. David, M.J., Lazer, M.A.B., Benkler, Y., Berinsky, A.J., Greenhill, K.M., Menczer, F., Metzger, M.J., Nyhan, B., Pennycook, G., Rothschild, D., et al.: The science of fake news. Science **359**(6380), 1094–1096 (2018)
10. Baly, R., Karadzhov, G., Alexandrov, D., Glass, J.R., Nakov, P.: Predicting factuality of reporting and bias of news media sources. In: Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing, Brussels, Belgium, 31 October–4 November 2018, pp. 3528–3539 (2018)
11. Wang, W.Y.: "liar, liar pants on fire": a new benchmark dataset for fake news detection. In: Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics, ACL 2017, Vancouver, Canada, 30 July–4 August 2017. Short Papers, vol. 2, pp. 422–426 (2017)
12. Hu, X., Tang, J., Zhang, Y., Liu, H.: Social spammer detection in microblogging. In: Proceedings of the Twenty-Third International Joint Conference on Artificial Intelligence, IJCAI '13, pp. 2633–2639. AAAI Press (2013)
13. Yoo, K.-H., Gretzel, U.: Comparison of deceptive and truthful travel reviews. Inf. Commun. Technol. Tour. **2009**, 37–47 (2009)

14. O'Brien, N., Latessa, S., Evangelopoulos, G., Boix, X.: The language of fake news: opening the black-box of deep learning based detectors (2018)
15. Boididou, C., Papadopoulos, S., Dang-Nguyen, D.-T., Boato, G., Riegler, M., Middleton, S.E., Petlund, A., Kompatsiaris, Y.: Verifying multimedia use at mediaeval 2016. In: Working Notes Proceedings of the MediaEval 2016 Workshop, Hilversum, The Netherlands, 20–21 October 2016 (2016)
16. Jin, Z., Cao, J., Zhang, Y., Zhou, J., Tian, Q.: Novel visual and statistical image features for microblogs news verification. IEEE Trans. Multimedia **19**(3), 598–608 (2017)
17. Rubin, V.L., Chen, Y., Conroy, N.K.: Deception detection for news: three types of fakes. Proc. Assoc. Inf. Sci. Technol. **52**(1), 1–4 (2015)
18. Shao, C., Ciampaglia, G.L., Flammini, A., Menczer, F.: Hoaxy: a platform for tracking online misinformation (2016). CoRR arXiv:1603.01511
19. Chen, Y., Conroy, N.J., Rubin, V.L.: Misleading online content: Recognizing clickbait as "false news". In: Proceedings of the 2015 ACM on Workshop on Multimodal Deception Detection, WMDD '15, pp. 15–19. ACM, New York, NY, USA (2015)
20. Qi, P., Cao, J., Yang, T., Guo, J., Li, J.: Exploiting multi-domain visual information for fake news detection. In: 2019 IEEE International Conference on Data Mining (ICDM), pp. 518–527. IEEE (2019)
21. Afroz, S., Brennan, M., Greenstadt, R.: Detecting hoaxes, frauds, and deception in writing style online. In: 2012 IEEE Symposium on Security and Privacy, pp. 461–475. IEEE (2012)
22. Potthast, M., Kiesel, J., Reinartz, K., Bevendorff, J., Stein, B.: A stylometric inquiry into hyperpartisan and fake news (2017). arXiv:1702.05638
23. Mukherjee, S., Weikum, G.: Leveraging joint interactions for credibility analysis in news communities. In: Proceedings of the 24th ACM International on Conference on Information and Knowledge Management, pp. 353–362 (2015)
24. Zhou, L., Twitchell, D.P., Qin, T., Burgoon, J.K., Nunamaker, J.F.: An exploratory study into deception detection in text-based computer-mediated communication. In: Proceedings of the 36th Annual Hawaii International Conference on System Sciences, pp. 10–IEEE (2003)
25. Burgoon, J.K., Burgoon, M., Wilkinson, M.: Writing style as predictor of newspaper readership, satisfaction and image. J. Q. **58**(2), 225–231 (1981)
26. Brennan, M.R., Greenstadt, R.: Practical attacks against authorship recognition techniques. In: Twenty-First IAAI Conference (2009)
27. Zheng, R., Li, J., Chen, H., Huang, Z.: A framework for authorship identification of online messages: Writing-style features and classification techniques. J. Am. Soc. Inform. Sci. Technol. **57**(3), 378–393 (2006)
28. Castillo, C., Mendoza, M., Poblete, B.: Predicting information credibility in time-sensitive social media. Internet Res. **23**(5), 560–588 (2013)
29. Horne, B.D., Adali, S.: This just in: fake news packs a lot in title, uses simpler, repetitive content in text body, more similar to satire than real news. In: Eleventh International AAAI Conference on Web and Social Media (2017)
30. Vosoughi, S., 'Neo' Mohsenvand, M., Roy, D.: Rumor gauge: predicting the veracity of rumors on twitter. ACM Trans. Knowl. Discov. Data (TKDD) **11**(4), 50 (2017)
31. Kwon, S., Cha, M., Jung, K., Chen, W., Wang, Y.: Prominent features of rumor propagation in online social media. In: 2013 IEEE 13th International Conference on Data Mining, pp. 1103–1108. IEEE (2013)
32. Qazvinian, V., Rosengren, E., Radev, D.R., Mei, Q.: Rumor has it: identifying misinformation in microblogs. In: Proceedings of the Conference on Empirical Methods in Natural Language Processing, EMNLP '11, pp. 1589–1599. Association for Computational Linguistics (2011)
33. Gupta, A., Kumaraguru, P.: Credibility ranking of tweets during high impact events. In: Proceedings of the 1st Workshop on Privacy and Security in Online Social Media, pp. 2–8 (2012)
34. Boididou, C., Papadopoulos, S., Zampoglou, M., Apostolidis, L., Papadopoulou, O., Kompatsiaris, Y.: Detection and visualization of misleading content on twitter. Int. J. Multimed. Inf. Retr. **7**(1), 71–86 (2018)

35. Rubin, V.L., Conroy, N.J., Chen, Y.: Towards news verification: Deception detection methods for news discourse. In: Hawaii International Conference on System Sciences, pp. 5–8 (2015)
36. Boididou, C., Andreadou, K., Papadopoulos, S., Dang-Nguyen, D.-T., Boato, G., Riegler, M., Kompatsiaris, Y., et al.: Verifying multimedia use at mediaeval 2015. MediaEval **3**(3), 7 (2015)
37. Yang, Y., Zheng, L., Zhang, J., Cui, Q., Li, Z., Yu, P.S.: Ti-cnn: convolutional neural networks for fake news detection (2018). arXiv:1806.00749
38. Ferrara, E., Varol, O., Davis, C., Menczer, F., Flammini, A.: The rise of social bots. Commun. ACM **59**(7), 96–104 (2016)
39. Castillo, C., Mendoza, M., Poblete, B.: Information credibility on twitter. In: Proceedings of the 20th International Conference on World Wide Web, pp. 675–684. ACM, New York, NY, USA (2011)
40. Wu, L., Li, J., Hu, X., Liu, H.: Gleaning wisdom from the past: Early detection of emerging rumors in social media. In: Proceedings of the 2017 SIAM International Conference on Data Mining, pp. 99–107. SIAM (2017)
41. Ma, J., Gao, W., Wong, K.-F.: Rumor detection on twitter with tree-structured recursive neural networks. In: Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), pp. 1980–1989. Association for Computational Linguistics (2018)
42. Vosoughi, S., Roy, D., Aral, S.: The spread of true and false news online. Science **359**(6380), 1146–1151 (2018)
43. Tacchini, E., Ballarin, G., Vedova, M.L.D., Moret, S., de Alfaro, L.: Some like it hoax: automated fake news detection in social networks (2017). arXiv:1704.07506
44. Shu, K., Wang, S., Liu, H.: Beyond news contents: The role of social context for fake news detection. In: Proceedings of the Twelfth ACM International Conference on Web Search and Data Mining, pp. 312–320 (2019)
45. Thorne, J., Vlachos, A., Cocarascu, O., Christodoulopoulos, C., Mittal, A.: The fact extraction and verification (fever) shared task. In: Proceedings of the First Workshop on Fact Extraction and VERification (FEVER), pp. 1–9. Association for Computational Linguistics (2018)
46. Ciampaglia, G.L., Shiralkar, P., Rocha, L.M., Bollen, J., Menczer, F., Flammini, A.: Computational fact checking from knowledge networks. PloS One **10**(6), e0128193 (2015)
47. Gupta, A., Lamba, H., Kumaraguru, P., Joshi, A.: Faking sandy: characterizing and identifying fake images on twitter during hurricane sandy. In: Proceedings of the 22nd international conference on World Wide Web, pp. 729–736 (2013)
48. Buckley, C., Salton, G., Allan, J.: Automatic retrieval with locality information using smart. In: Proceedings of the First Text REtrieval Conference TREC-1, pp. 59–72 (1993)
49. Xia, X., Yang, X., Wu, C., Li, S., Bao, L.: Information credibility on twitter in emergency situation. In: Pacific-Asia Workshop on Intelligence and Security Informatics, pp. 45–59. Springer (2012)
50. Koller, D., Friedman, N.: Probabilistic Graphical Models: Principles And Techniques. MIT Press (2009)
51. Gupta, A., Kumaraguru, P., Castillo, C., Meier, P.: Tweetcred: real-time credibility assessment of content on twitter. In: International Conference on Social Informatics, pp. 228–243. Springer (2014)
52. Pennebaker, J.W., Boyd, R.L., Jordan, K., Blackburn, K.: The development and psychometric properties of liwc2015. Technical report (2015)
53. Pérez-Rosas, V., Kleinberg, B., Lefevre, A., Mihalcea, R.: Automatic detection of fake news. In: Proceedings of the 27th International Conference on Computational Linguistics, Santa Fe, New Mexico, USA, August 2018, pp. 3391–3401. Association for Computational Linguistics (2018)
54. Rashkin, H., Choi, E., Jang, J.Y., Volkova, S., Choi, Y.: Truth of varying shades: analyzing language in fake news and political fact-checking. In: Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing, pp. 2931–2937 (2017)

55. Reddy, H., Raj, N., Gala, M., Basava, A.: Text-mining-based fake news detection using ensemble methods. Int. J. Autom. Comput. 1–12 (2020)
56. Mikolov, T., Chen, K., Corrado, G., Dean, J.: Efficient estimation of word representations in vector space. In: Bengio Y, LeCun Y (eds) 1st International Conference on Learning Representations, ICLR 2013, Workshop Track Proceedings, Scottsdale, Arizona, USA, 2–4 May 2013 (2013)
57. Bojanowski, P., Grave, E., Joulin, A., Mikolov, T.: Enriching word vectors with subword information. Trans. Assoc. Comput. Linguist. **5**, 135–146 (2017)
58. Karimi, H., Roy, P., Saba-Sadiya, S., Tang, J.: Multi-source multi-class fake news detection. In: Proceedings of the 27th International Conference on Computational Linguistics, pp. 1546–1557 (2018)
59. Singhania, S., Fernandez, N., Rao, S.: 3han: A deep neural network for fake news detection. In: International Conference on Neural Information Processing, pp. 572–581. Springer (2017)
60. Pennington, J., Socher, R., Manning, C.D.: Glove: global vectors for word representation. In: Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP), pp. 1532–1543 (2014)
61. Shim, J.-S., Won, H.-R., Ahn, H.: A study on the effect of the document summarization technique on the fake news detection model. J. Intell. Inf. Syst. **25**(3), 201–220 (2019)
62. Zhou, L., Burgoon, J.K., Nunamaker, J.F., Twitchell, D.: Automating linguistics-based cues for detecting deception in text-based asynchronous computer-mediated communications. Group Decis. Negot. **13**(1), 81–106 (2004)
63. Zhang, H., Fan, Z., Zheng, J., Liu, Q.: An improving deception detection method in computer-mediated communication. J. Netw. **7**(11), 1811 (2012)
64. Mihalcea, R., Strapparava, C.: The lie detector: explorations in the automatic recognition of deceptive language. In: Proceedings of the ACL-IJCNLP 2009 Conference Short Papers, pp. 309–312 (2009)
65. Ledger, G., Merriam, T.: Shakespeare, fletcher, and the two noble kinsmen. Lit. Linguist. Comput. **9**(3), 235–248 (1994)
66. Forsyth, R.S., Holmes, D.I.: Feature-finding for text classification. Lit. Linguist. Comput. **11**(4), 163–174 (1996)
67. Hancock, J.T., Curry, L.E., Goorha, S., Woodworth, M.: On lying and being lied to: a linguistic analysis of deception in computer-mediated communication. Discourse Process. **45**(1), 1–23 (2007)
68. Burgoon, J.K., Blair, J.P., Qin, T., Nunamaker, J.F.: Detecting deception through linguistic analysis. In: International Conference on Intelligence and Security Informatics, pp. 91–101. Springer (2003)
69. Feng, S., Banerjee, R., Choi, Y.: Syntactic stylometry for deception detection. In: Proceedings of the 50th Annual Meeting of the Association for Computational Linguistics. Short Papers, vol. 2, pp. 171–175 (2012)
70. Mann, W.C., Thompson, S.A.: Rhetorical structure theory: toward a functional theory of text organization. Text **8**(3), 243–281 (1988)
71. Rubin, V.L., Lukoianova, T.: Truth and deception at the rhetorical structure level. J. Assoc. Inf. Sci. Technol. **66**(5), 905–917 (2015)
72. Rubin, V.L., Conroy, N., Chen, Y., Cornwell, S.: Fake news or truth? Using satirical cues to detect potentially misleading news. In: Proceedings of the second workshop on computational approaches to deception detection, pp. 7–17 (2016)
73. Recasens, M., Danescu-Niculescu-Mizil, C., Jurafsky, D.: Linguistic models for analyzing and detecting biased language. In: Proceedings of the 51st Annual Meeting of the Association for Computational Linguistics. Long Papers, vol. 1, pp. 1650–1659 (2013)
74. Koppel, M., Schler, J., Bonchek-Dokow, E.: Measuring differentiability: unmasking pseudonymous authors. J. Mach. Learn. Res. **8**, 1261–1276 (2007)
75. Conover, M.D., Gonçalves, B., Ratkiewicz, J., Flammini, A., Menczer, F.: Predicting the political alignment of twitter users. In: 2011 IEEE Third International Conference on Privacy, Security, Risk and Trust and 2011 IEEE Third International Conference on Social Computing, pp. 192–199. IEEE (2011)

76. Nakashole, N., Mitchell, T.: Language-aware truth assessment of fact candidates. In: Proceedings of the 52nd Annual Meeting of the Association for Computational Linguistics. Long Papers, vol. 1, pp. 1009–1019 (2014)
77. Jin, Z., Cao, J., Guo, H., Zhang, Y., Luo, J.: Multimodal fusion with recurrent neural networks for rumor detection on microblogs. In: Proceedings of the 25th ACM International Conference on Multimedia, pp. 795–816 (2017)
78. Khattar, D., Goud, J.S., Gupta, M., Varma, V.: Mvae: multimodal variational autoencoder for fake news detection. In: The World Wide Web Conference, pp. 2915–2921 (2019)
79. Risdal, M.: Kaggle: getting real about fake news (2016). https://www.kaggle.com/mrisdal/fake-news/
80. Jin, Z., Cao, J., Luo, J., Zhang, Y.: Image credibility analysis with effective domain transferred deep networks (2016). arXiv:1611.05328
81. Goodfellow, I., Pouget-Abadie, J., Mirza, M., Xu, B., Warde-Farley, D., Ozair, S., Courville, A., Bengio, Y.: Generative adversarial nets. In: Advances in Neural Information Processing Systems, pp. 2672–2680 (2014)
82. Nataraj, L., Mohammed, T.M., Manjunath, B.S., Chandrasekaran, S., Flenner, A., Bappy, J.H., Roy-Chowdhury, A.K.: Detecting GAN generated fake images using co-occurrence matrices. Electron. Imaging **2019**(5), 532-1 (2019)
83. Kumar, S., West, R., Leskovec, J.: Disinformation on the web: Impact, characteristics, and detection of wikipedia hoaxes. In: Proceedings of the 25th International Conference on World Wide Web, pp. 591–602 (2016)
84. Zhao, J., Cao, N., Wen, Z., Song, Y., Lin, Y.-R.: # fluxflow: visual analysis of anomalous information spreading on social media. IEEE Trans. Vis. Comput. Graph. **20**(12), 1773–1782 (2014)
85. Zhang, X., Ghorbani, A.A.: An overview of online fake news: characterization, detection, and discussion. Inf. Process. & Manag. **57**(2), 102025 (2020)
86. Gurajala, S., White, J.S., Hudson, B., Matthews, J.N.: Fake twitter accounts: profile characteristics obtained using an activity-based pattern detection approach. In: Proceedings of the 2015 International Conference on Social Media & Society, p. 9. ACM (2015)
87. Chu, Z., Gianvecchio, S., Wang, H., Jajodia, S.: Detecting automation of twitter accounts: are you a human, bot, or cyborg? IEEE Trans. Dependable Secure Comput. **9**(6), 811–824 (2012)
88. Ghosh, R., Surachawala, T., Lerman, K.: Entropy-based classification of 'retweeting' activity on twitter (2011). arXiv:1106.0346
89. Dickerson, J.P., Kagan, V., Subrahmanian, V.S.: Using sentiment to detect bots on twitter: are humans more opinionated than bots? In: 2014 IEEE/ACM International Conference on Advances in Social Networks Analysis and Mining (ASONAM 2014), pp. 620–627. IEEE (2014)
90. Canini, K.R., Suh, B., Pirolli, P.L.: Finding credible information sources in social networks based on content and social structure. In *2011 IEEE Third International Conference on Privacy, Security, Risk and Trust and 2011 IEEE Third International Conference on Social Computing*, pp. 1–8. IEEE (2011)
91. Long, Y.: Fake news detection through multi-perspective speaker profiles. Association for Computational Linguistics (2017)
92. Yang, F., Liu, Y., Yu, X., Yang, M.: Automatic detection of rumor on sina weibo. In: Proceedings of the ACM SIGKDD Workshop on Mining Data Semantics. ACM, New York, NY, USA (2012)
93. Buntain, C., Golbeck, J.: Automatically identifying fake news in popular twitter threads. In: 2017 IEEE International Conference on Smart Cloud (SmartCloud), pp. 208–215. IEEE (2017)
94. Tschiatschek, S., Singla, A., Gomez-Rodriguez, M., Merchant, A., Krause, A.: Fake news detection in social networks via crowd signals. In: Companion of the The Web Conference 2018 on The Web Conference 2018, WWW 2018, Lyon , France, 23–27 April 2018, pp. 517–524 (2018)

95. Suzuki, Y.: A credibility assessment for message streams on microblogs. In: 2010 International Conference on P2P, Parallel, Grid, Cloud and Internet Computing, pp. 527–530. IEEE (2010)
96. Ma, J., Gao, W., Wong, K.-F.: Detect rumors in microblog posts using propagation structure via kernel learning. In: ACL (1), pp. 708–717. Association for Computational Linguistics (2017)
97. Monti, F., Frasca, F., Eynard, D., Mannion, D., Bronstein, M.M.: Fake news detection on social media using geometric deep learning (2019). arXiv:1902.06673
98. Wu, K., Yang, S., Zhu, K.Q.: False rumors detection on sina weibo by propagation structures. In: 2015 IEEE 31st International Conference on Data Engineering, pp. 651–662 (2015)
99. Ma, J., Gao, W., Mitra, P., Kwon, S., Jansen, B.J., Wong, K.F., Cha, M.: Detecting rumors from microblogs with recurrent neural networks. In: IJCAI, pp. 3818–3824. IJCAI/AAAI Press (2016)
100. Yu, F., Liu, Q., Wu, S., Wang, L., Tan, T.: A convolutional approach for misinformation identification. In: Proceedings of the Twenty-Sixth International Joint Conference on Artificial Intelligence, IJCAI-17, pp. 3901–3907 (2017)
101. Vedova, M.L.D., Tacchini, E., Moret, S., Ballarin, G., DiPierro, M., de Alfaro, L.: Automatic online fake news detection combining content and social signals. In: 2018 22nd Conference of Open Innovations Association (FRUCT), pp. 272–279. IEEE (2018)
102. Gupta, A., Lamba, H., Kumaraguru, P.: 1.00 per rt # bostonmarathon # prayforboston: analyzing fake content on twitter. In: eCrime Researchers Summit (eCRS), 2013, pp. 1–12. IEEE (2013)
103. Liu, Y., Wu, Y.-F.B.: FNED: a deep network for fake news early detection on social media. ACM Trans. Inf. Syst. (TOIS) **38**(3), 1–33 (2020)
104. Liu, X., Nourbakhsh, A., Li, Q., Fang, R., Shah, S.: Real-time rumor debunking on twitter. In: Proceedings of the 24th ACM International on Conference on Information and Knowledge Management, CIKM '15, pp. 1867–1870. ACM, New York, NY, USA (2015)
105. Jin, Z., Cao, J., Zhang, Y., Luo, J. News verification by exploiting conflicting social viewpoints in microblogs. In: Proceedings of the Thirtieth AAAI Co:nference on Artificial Intelligence, AAAI'16, pp. 2972–2978. AAAI Press (2016)
106. Riedel, B., Augenstein, I., Spithourakis, G.P., Riedel, S.: A simple but tough-to-beat baseline for the fake news challenge stance detection task (2017). arXiv:1707.03264
107. Chopra, S., Jain, S., Sholar, J.M.: Towards automatic identification of fake news: headline-article stance detection with LSTM attention models. In: Stanford CS224d Deep Learning for NLP final project (2017)
108. Bourgonje, P., Schneider, J.M., Rehm, G.: From clickbait to fake news detection: an approach based on detecting the stance of headlines to articles. In: Proceedings of the 2017 EMNLP Workshop: Natural Language Processing meets Journalism, pp. 84–89 (2017)
109. Ferreira, W., Vlachos, A.: Emergent: a novel data-set for stance classification. In: Proceedings of the 2016 conference of the North American chapter of the association for computational linguistics: human language technologies, pp. 1163–1168 (2016)
110. Mendoza, M., Poblete, B., Castillo, C.: Twitter under crisis: can we trust what we RT? In: Proceedings of the First Workshop on Social Media Analytics, pp. 71–79. ACM (2010)
111. Derczynski, L., Bontcheva, K., Liakata, M., Procter, R., Hoi, G.W.S., Zubiaga, A.: Semeval-2017 task 8: rumoureval: determining rumour veracity and support for rumours. In: Proceedings of the 11th International Workshop on Semantic Evaluation (SemEval-2017), pp. 69–76. Association for Computational Linguistics (2017)
112. Enayet, O., El-Beltagy, S.R.: Niletmrg at semeval-2017 task 8: determining rumour and veracity support for rumours on twitter. In: Proceedings of the 11th International Workshop on Semantic Evaluation (SemEval-2017), pp. 470–474. Association for Computational Linguistics (2017)
113. Chen, Y.-C., Liu, Z.-Y., Kao, H.-Y.: IKM at semeval-2017 task 8: convolutional neural networks for stance detection and rumor verification. In: SemEval@ACL, pp. 465–469. Association for Computational Linguistics (2017)

114. Jin, Z., Cao, J., Jiang, Y.-G., Zhang, Y.: News credibility evaluation on microblog with a hierarchical propagation model. In: 2014 IEEE International Conference on Data Mining, pp. 230–239. IEEE (2014)
115. Ruchansky, N., Seo, S., Liu, Y.: CSI: a hybrid deep model for fake news detection. In: Proceedings of the 2017 ACM on Conference on Information and Knowledge Management, pp. 797–806. ACM (2017)
116. Vlachos, A., Riedel, S.: Fact checking: task definition and dataset construction. In: Proceedings of the ACL 2014 Workshop on Language Technologies and Computational Social Science, pp. 18–22. Association for Computational Linguistics (2014)
117. Nakov, P., Barrón-Cedeño, A., Elsayed, T., Suwaileh, R., Màrquez, L., Zaghouani, W., Atanasova, P., Kyuchukov, S., Da San Martino, G.: Overview of the clef-2018 checkthat! lab on automatic identification and verification of political claims. In: Proceedings of the Ninth International Conference of the CLEF Association: Experimental IR Meets Multilinguality, Multimodality, and Interaction. Lecture Notes in Computer Science, Avignon, France, September 2018. Springer (2018)
118. Atanasova, P., Màrquez, L., Barrón-Cedeño, A., Elsayed, T., Suwaileh, R., Zaghouani, W., Kyuchukov, S., Da San Martino, G., Nakov, P.: Overview of the clef-2018 checkthat! lab on automatic identification and verification of political claims, task 1: check-worthiness. In: CLEF 2018 Working Notes. Working Notes of CLEF 2018 - Conference and Labs of the Evaluation Forum, CEUR Workshop Proceedings, Avignon, France, September 2018 (2018). CEUR-WS.org
119. Barrón-Cedeño, A., Elsayed, T., Suwaileh, R., Màrquez, L., Atanasova, P., Zaghouani, W., Kyuchukov, S., Da San Martino, G., Nakov, P.: Overview of the clef-2018 checkthat! lab on automatic identification and verification of political claims, task 2: factuality. In: CLEF 2018 Working Notes. Working Notes of CLEF 2018 - Conference and Labs of the Evaluation Forum, CEUR Workshop Proceedings, Avignon, France, September 2018 (2018). CEUR-WS.org
120. Agirre, E., Cer, D., Diab, M., Gonzalez-Agirre, A., Guo, W.: * sem 2013 shared task: semantic textual similarity. In: Second Joint Conference on Lexical and Computational Semantics (* SEM), Volume 1: Proceedings of the Main Conference and the Shared Task: Semantic Textual Similarity, pp. 32–43 (2013)
121. Hanselowski, A., Zhang, H., Li, Z., Sorokin, D., Schiller, B., Schulz, C., Gurevych, I.: Ukp-athene: multi-sentence textual entailment for claim verification (2018). arXiv:1809.01479
122. Chen, Q., Zhu, X., Ling, Z.-H., Wei, S., Jiang, H., Inkpen, D.: Enhanced LSTM for natural language inference. In: Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics. Long Papers, vol. 1, pp. 1657–1668. Association for Computational Linguistics (2017)
123. Yoneda, T., Mitchell, J., Welbl, J., Stenetorp, P., Riedel, S.: UCL machine reading group: Four factor framework for fact finding (hexaf). In: Proceedings of the First Workshop on Fact Extraction and VERification (FEVER), pp. 97–102. Association for Computational Linguistics (2018)
124. Yin, W., Roth, D.: Twowingos: a two-wing optimization strategy for evidential claim verification (2018). arXiv:1808.03465
125. Thorne, J., Vlachos, A., Christodoulopoulos, C., Mittal, A.: Fever: a large-scale dataset for fact extraction and verification (2018). arXiv:1803.05355
126. Vydiswaran, V.G.V., Zhai, C., Roth, D.: Content-driven trust propagation framework. In: Proceedings of the 17th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, pp. 974–982 (2011)
127. Wu, Y., Agarwal, P.K., Li, C., Yang, J., Yu, C.: Toward computational fact-checking. Proc. VLDB Endow. **7**(7), 589–600 (2014)
128. Kim, J., Tabibian, B., Oh, A., Schölkopf, B., Gomez-Rodriguez, M.: Leveraging the crowd to detect and reduce the spread of fake news and misinformation. In: Proceedings of the Eleventh ACM International Conference on Web Search and Data Mining, pp. 324–332. ACM (2018)
129. Hassan, N., Li, C., Tremayne, M.: Detecting check-worthy factual claims in presidential debates. In: Proceedings of the 24th ACM International on Conference on Information and Knowledge Management, pp. 1835–1838 (2015)

130. Zubiaga, A., Hoi, G.W.S., Liakata, M., Procter, R., Tolmie, P.: Analysing how people orient to and spread rumours in social media by looking at conversational threads (2015). CoRR arXiv:1511.07487
131. Oshikawa, R., Qian, J., Wang, W.Y.: A survey on natural language processing for fake news detection (2018). arXiv:1811.00770
132. Xu, J., Li, H.: Adarank: a boosting algorithm for information retrieval. In: Proceedings of the 30th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval, pp. 391–398 (2007)
133. Saxena, A., Hsu, W., Lee, M.L., Chieu, H.L., Ng, L., Teow, L.N.: Mitigating misinformation in online social network with top-k debunkers and evolving user opinions. In: Companion Proceedings of the Web Conference 2020, pp. 363–370 (2020)
134. Amoruso, M., Anello, D., Auletta, V., Ferraioli, D.: Contrasting the spread of misinformation in online social networks. In: Proceedings of the 16th Conference on Autonomous Agents and MultiAgent Systems, pages 1323–1331. International Foundation for Autonomous Agents and Multiagent Systems (2017)
135. Saxena, A., Saxena, H., Gera, R.: k-truthscore: Fake news mitigation in the presence of strong user bias. In: International Conference on Computational Data and Social Networks, pp. 113–126. Springer (2020)
136. Mitra, T., Gilbert, E.: Credbank: a large-scale social media corpus with associated credibility annotations. In: ICWSM, pp. 258–267 (2015)
137. Pomerleau, D., Rao, D.: The fake news challenge: exploring how artificial intelligence technologies could be leveraged to combat fake news (2016). http://www.fakenewschallenge.org/
138. Shu, K., Wang, S., Liu, H.: Beyond news contents: the role of social context for fake news detection. In: Proceedings of the Twelfth ACM International Conference on Web Search and Data Mining, WSDM '19, pp. 312–320. ACM, New York, NY, USA (2019)
139. Pratiwi, I.Y.R., Asmara, R.A., Rahutomo, F.: Study of hoax news detection using naïve bayes classifier in Indonesian language. In: 2017 11th International Conference on Information & Communication Technology and System (ICTS), pp. 73–78. IEEE (2017)
140. Baly, R., Mohtarami, M., Glass, J., Màrquez, L., Moschitti, A., Nakov, P.: Integrating stance detection and fact checking in a unified corpus. In: Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies. Short Papers, vol. 2, pp. 21–27. Association for Computational Linguistics (2018)
141. Barrón-Cedeño, A., Elsayed, T., Suwaileh, R., Màrquez, L., Atanasova, P., Zaghouani, W., Kyuchukov, S., Da San Martino, G., Nakov, P.: Overview of the clef-2018 checkthat! lab on automatic identification and verification of political claims. task 2: factuality. In: CLEF (Working Notes), vol. 2125 (2018)
142. Sadiku, M.N.O., Eze, T.P., Musa, S.M.: Fake news and misinformation. IJASRE 4 (2018)
143. Gelfert, A.: Fake news: a definition. Informal Logic **38**(1), 84–117 (2018)
144. Waldrop, M.M.: News feature: the genuine problem of fake news. Proc. Natl. Acad. Sci. **114**(48), 12631–12634 (2017)
145. Campan, A., Cuzzocrea, A., Truta, T.M.: Fighting fake news spread in online social networks: actual trends and future research directions. In: 2017 IEEE International Conference on Big Data (Big Data), pp. 4453–4457. IEEE (2017)
146. Zannettou, S., Sirivianos, M., Blackburn, J., Kourtellis, N.: The web of false information: rumors, fake news, hoaxes, clickbait, and various other shenanigans (2018). arXiv:1804.03461
147. Meinert, J., Mirbabaie, M., Dungs, S., Aker, A.: Is it really fake?–towards an understanding of fake news in social media communication. In: International Conference on Social Computing and Social Media, pp. 484–497. Springer (2018)
148. Wu, L., Morstatter, F., Hu, X., Liu, H.: Mining misinformation in social media. In: Big Data in Complex and Social Networks, pp. 123–152 (2016)
149. Özgöbek, Ö., Gulla, J.A.: Towards an understanding of fake news. In: CEUR Workshop Proceedings (2017)

150. Shu, K., Bernard, H.R., Liu, H.: Studying fake news via network analysis: detection and mitigation. In: Emerging Research Challenges and Opportunities in Computational Social Network Analysis and Mining, pp. 43–65. Springer (2019)
151. Cao, J., Guo, J., Li, X., Jin, Z., Guo, H., Li, J.: Automatic rumor detection on microblogs: a survey (2018). arXiv:1807.03505
152. Serrano, E., Iglesias, C.A., Garijo, M.: A survey of twitter rumor spreading simulations. In: Computational Collective Intelligence, pp. 113–122. Springer (2015)
153. Zubiaga, A., Aker, A., Bontcheva, K., Liakata, M., Procter, R.: Detection and resolution of rumours in social media: a survey. ACM Comput. Surv. (CSUR) **51**(2), 32 (2018)
154. Shelke, S., Attar, V.: Source detection of rumor in social network-a review. Online Soc. Netw. Media **9**, 30–42 (2019)
155. Rehm, G.: An infrastructure for empowering internet users to handle fake news and other online media phenomena. In: International Conference of the German Society for Computational Linguistics and Language Technology, pp. 216–231. Springer (2017)
156. Ndii, M.Z., Carnia, E., Supriatna, A.K.: Mathematical models for the spread of rumors: a review. In: Issues and Trends in Interdisciplinary Behavior and Social Science, pp. 65–73. CRC Press (2018)
157. Wang, P., Angarita, R., Renna, I.: Is this the era of misinformation yet? combining social bots and fake news to deceive the masses. In: The 2018 Web Conference Companion (2018)
158. Alemanno, A.: How to counter fake news? A taxonomy of anti-fake news approaches. Eur. J. Risk Regul. **9**(1), 1–5 (2018)
159. Sullivan, M.C.: Why librarians can't fight fake news. J. Librariansh. Inf. Sci. 0961000618764258 (2018)
160. Niedringhaus, K.L.: Information literacy in a fake/false news world: Why does it matter and how does it spread? Int. J. Legal Inf. **46**(2), 97–100 (2018)
161. Tarran, B.: Why facts are not enough in the fight against fake news. Significance **14**(5), 6–7 (2017)
162. Rochlin, N.: Fake news: belief in post-truth. Libr. Hi Tech **35**(3), 386–392 (2017)
163. Berghel, H.: Lies, damn lies, and fake news. Computer **50**(2), 80–85 (2017)
164. DiFranzo, D., Gloria, M.J.K.: Filter bubbles and fake news. ACM Crossroads **23**(3), 32–35 (2017)
165. Andrews, W.: Fighting fake news **93**, 2 (2017)

Chapter 16
Fake News Propagation and Mitigation Techniques: A Survey

Akrati Saxena, Pratishtha Saxena, and Harita Reddy

Abstract Today, major online social networking websites host millions of user accounts. These websites provide a convenient platform for sharing information and opinions in the form of microblogs. However, the ease of sharing also brings ramifications in the form of fake news, misinformation, and rumors, which has become highly prevalent recently. The impact of fake news dissemination was observed in major political events like the US elections and the Jakarta elections, as well as the distortion of celebrities and companies' reputation. Researchers have studied the propagation of fake news over social media websites and have proposed various techniques to combat fake news. In this chapter, we discuss propagation models for misinformation and review the fake news mitigation techniques. We also compose a list of datasets used in fake news-related studies. The chapter is concluded with open research questions.

16.1 Introduction

Since the advent of the World Wide Web (WWW), there has been a shift in news consumption from conventional news sources such as newspapers to online news sources, which enable people from different parts of the world to stay updated with diverse news and trends. The most common online sources currently include (i) standalone websites that include news websites, blogs, and media websites, and (ii) social media, which offers a platform for sharing news or opinions, which can be

A. Saxena (✉)
Department of Mathematics and Computer Science, Eindhoven University of Technology, Eindhoven, The Netherlands
e-mail: a.saxena@tue.nl

P. Saxena
G. L. Bajaj Institute of Technology and Management, Greater Noida, India
e-mail: pratistha.saxena@glbitm.ac.in

H. Reddy
Surat, Gujarat, India

A. Biswas et al. (eds.), *Principles of Social Networking*,
Smart Innovation, Systems and Technologies 246,
https://doi.org/10.1007/978-981-16-3398-0_16

shared further by other users [1]. Social media has transformed how people communicate information and opinions with each other. Online Social Networks (OSNs), such as *Google+*, *Facebook*, and *Twitter*, have made it easy to share not only textual content but also multimedia and URL-links to external websites with a potentially broad audience. The main reasons why the consumption of news on social media has increased include (i) the low-cost Internet access and (ii) the ease of sharing or discussing the news with other people on social media [2]. However, the increased ease of accessing the Internet and fast dissemination of content on social media has also given rise to the problem of the spread of "fake news."

The extent its adverse impact has been observed in the United States presidential elections, 2016; a study of Twitter over a time range of 10 days revealed that the users across the country got relatively more content containing misinformation, conspiracy, and polarity than the authentic content created by professionals [3]. A popular conspiracy theory that 'Pizzagate' was widely spread during the election with around 1 million tweets on Twitter, and it was proven to be fake [2]. It is difficult for humans to ascertain the veracity of information, especially if the information is made to look legitimate. This has been shown in an experiment on hoax Wikipedia articles in which even trained users misconstrued some hoax articles with references to be true [4, 5]. Unreliable news may consist of *hoaxes*, which try to mislead readers with paranoia, *propaganda*, which is created for promoting a certain agenda, and *satire*, which is fake news created for humor [6]. From these, readers are likely to be able to identify the intent of satirical news and not take them seriously.

Fake news is closely linked with rumors on social media. DiFonzo and Bordia [7] defined that the rumor is an unverified information that is circulated in contexts of danger, ambiguity, or potential threat. Though unverified at a certain point of time, rumors may come out to be false or true or may even remain unverified in the future. Some research works, however, consider rumors as those statements which are eventually verified to be false [8]. Social media posts pertaining to a rumored event may evoke different kinds of responses from users; users may agree, disagree, or even ask for further evidence supporting the claims made in the post [8]. In this work, we will discuss research works related to fake news, rumor, and misinformation; and these terms will be used interchangeably during the discussion.

The adverse impact of fake news has attracted researchers' interest in studying their propagation, detection, and mitigation. Understanding how fake news propagates compared to real news involves modeling social media connections and information sharing as a network. There are four main kinds of propagation models to study misinformation propagation, which are discussed later in the chapter. On social networking websites such as Facebook, there is selective exposure to news in terms of ideology because users tend to follow other users who have similar viewpoints or opinions, and it leads to *Echo Chamber Effect* or *filter bubble* [9–11]. Due to echo-chambers, users are exposed repetitively to news catering to a specific ideology favored by their neighbors and are likely to believe it. The presence of echo-chambers, therefore, makes the problem of fighting fake news more challenging.

Researchers have proposed several methods for mitigating fake news; the main categories of approaches are (i) influence blocking [12] and (ii) truth campaigning

[13]. In *influence blocking* approaches, the aim is to determine a minimal number of users who should be immunized to minimize the fake news spread. In the second category, i.e., *truth campaigning*, the basic idea is to ensure that the users are aware of the correct information so that they will tend to believe in it. In truth campaigning techniques, we identify a set of users called 'mitigators' who will spread correct information to counter the impact of fake news propagating in the social network. Researchers also have developed several mitigation tools to inform users regarding the credibility of the content to prevent further dissemination of fake news. For these tools, the fake news detection techniques need to be leveraged to analyze the credibility and flag the posts appropriately [14, 15].

The chapter structure is discussed next. In Sect. 16.2, we discuss propagation models for fake news spreading. In Sect. 16.3, fake news mitigation techniques have been discussed. This is followed by the conclusion and further ideas for research in this domain.

16.2 Propagation Models

On OSNs, information starts spreading from a group of source nodes, also called seed nodes. These nodes share the information on their accounts and influence or infect their neighbors. Once a neighboring node believes in the information, it further shares the information with its neighbors. Thus, the information keeps spreading in the network. In social networks, each directed/undirected link has an influence probability that represents the probability of passing the information to the target node from the source node. If a user believes in the given information, it is called influenced or infected user. Several propagation models have been proposed to simulate how information spreads on a network. These models are also referred to as spreading models. Next, we will discuss the models capturing underlying propagation phenomena and their extensions.

16.2.1 Independent Cascade Model

In Independent Cascade Model (ICM) [16], one source node or a group of nodes start the infection. When a node is infected, in the next iteration, it tries to infect all of its neighbors with the influence probabilities of their connection, and it will not infect any of its neighbors in later iterations. If in an iteration, there is no newly infected node, the spreading process will be finished. The total number of infected nodes denotes the spreading power or influencing power of the source nodes. In the ICM model, a node's influential power is determined as follows. The propagation is started from the given node, and the process is repeated several times. The average number of infected nodes in all iterations is considered the spreading or influential power of the source node. The ICM model has been extended to model if two competitive information propagates in the network, and it is called the Competitive Independent Cascade Model (CICM) [17].

The network structure plays a crucial role in information propagation. Saxena et al. [18] extended the independent cascade model and proposed a penta-level spreading model that is based on two meso-scale properties of the network, (i) community and (ii) core-periphery structure. Each node belongs either to the core or the periphery of the network. The periphery nodes are also organized into the communities [19]. In the network, the edges are divided into five categories based on the type of the source node and the target node. The influence probability of an edge depends on its category. The probabilities are ordered as $P_{cc} > P_{cp} > P_{pp_0} > P_{pp_1} > P_{pc}$ where c represents core node, p represents periphery node, pp_0 shows that both the source and target are from the same community, and pp_1 shows that the source and target are from different communities. The proposed model is verified using the Higgs-Boson Twitter dataset [20] and captures the role of the hierarchical ordering of the society in the information sharing process [21].

16.2.2 Linear Threshold Model

In the real world, it is observed that a user accepts or agrees with a piece of information if the number of its friends believing in that information is higher than a particular threshold value. The threshold might be different for different users. In the Linear Threshold Model (LTM) [22], the information spreading starts from a group of source nodes. In each iteration, the new nodes will be influenced if the number of influenced neighbors is more than their threshold value. The infection process will stop if no new node is influenced in an iteration. This model is also known as Tipping Model when the threshold value for each node is the same [23].

LTM also has been extended to model competitive information propagation. In the Competitive Linear Threshold Model (CLTM), different kinds of information propagate simultaneously. A user believes in any one of the information if its impact computed based on the neighboring node is higher than the threshold value. Yang et al. [24] extended the CLTM for modeling competitive information propagation in directed networks and proposed the Linear Threshold model with One Direction state Transition (LT1DT) model. Pham et al. [25] modified LT model when information from multiple topics cascade in the network, called Multiple Adoption Linear Threshold (MA-LT) model.

16.2.3 Compartment Spreading Models

Compartment models are used for the mathematical modeling of the infection propagation. In compartments models, users are divided into compartments, and all the users belonging to one compartment follow the same rule. The Susceptible-Infected-

Recovered (SIR) model [26] is the simplest compartment model where a given node can be present in either of the three possible states: (i) S (susceptible), (ii) I (infected), and (iii) R (recovered). To initialize the model, all nodes are set to the susceptible (S) mode. The infection or influence propagation will start by infecting a node (or group of nodes), and their status is set to be Infected. An infected/influenced node u influences each of its susceptible neighbors with a given influence probability (λ) and will update their status to Infected if the node is infected. The influence probability is also referred to as infection probability in the SIR model. The status of node u is updated to Recovered with probability μ, once it has tried to influence all of its neighbors. A recovered node will not change its state again during the spreading process. The influence propagation is stopped when no new node is infected in the network.

The other variants of compartment models are Susceptible, Infected (SI) [27] and Susceptible, Infected, Susceptible (SIS) [28] which are used very often to simulate the propagation where a node can be in any of the two states. Zhao et al. [29] further extended the SIR model by including the ignorant behavior of users where users can have any of the three states: Stiflers, Ignorants, and Spreaders. They did the numerical analysis of studying the impact of various parameters on misinformation spreading. They further proposed Susceptible-Infected-Hibernator-Removed (SIHR) model [30], where nodes can also be Hibernators. Xiong et al. [31] proposed Susceptible, Contacted, Infected, and Refractory (SCIR) model and showed that the degree-based density of affected nodes increases monotonously with their degree. Nekovee et al. [32] combined the SIR model with Maki–Thompson (MK) model [33] to study rumor spreading on different kind of networks, such as random networks and different types of scale-free networks. Jin et al. [34] studied rumors and news spreading on Twitter using the Susceptible, Exposed, Infected, Skeptic (SEIZ) model having four user states [35].

Tambuscio et al. [36] extended the SIS model, where infected users are further divided into two compartments, called Fact-Checkers and Believers. The proposed model is based on the fact-checking activity that can be assigned based on varying probabilities. The authors provided a threshold for this probability that can be used to get an idea of the number of fact-checkers that will be sufficient to remove the fake news from the system altogether. They further verified the proposed model on scale-free, random, and real-world networks using different model parameters.

16.2.4 Opinion Formation Model

Opinion Formation (OF)-based models are designed to model the propagation of opinion or belief on OSNs [37]. In OF model, the current opinion of a node depends on its previous opinion and the opinion of its neighbors. Evans and Feng [38] proposed an OF model that considers the network structure while computing users' opinion. In the proposed model, the user's opinion depends on the strength of the user's opinion, the level of agreement of the user with its neighboring nodes, and its weighted degree. There are several other variations of OF models, including [39–43].

OF models also have been extended to model the propagation of two competitive opinions, such as misinformation and its counter-belief. In [13], the authors proposed an opinion formation model for misinformation mitigation where the opinion of a user is decided based on its neighbors' opinion. A user can be in any of the three states, believing in misinformation, believing in counter-message, or neutral. To compute the opinion of a node u, first, we identify its neighbors believing in misinformation and its counter-message, and put them in two sets called negative and positive neighbors, respectively. Next, we compute the influence of both types of neighbors on node u using the proposed formula. If the influence of misinformation is greater than the influence of counter-message (computed based on the opinion of its neighbors) plus a threshold, then node u will believe in misinformation. Similarly, if the influence of positive neighbors is greater than the influence of negative neighbors plus the threshold, the user believes in counter-message. Otherwise, the user remains neutral. The model is verified on the Twitter network for rumor propagation, and its accuracy in opinion modeling is better than the ICM and classic opinion formation model.

We have briefly discussed all the propagation models, which are mainly used to model misinformation propagation and its mitigation. In the next section, we discuss the fake news mitigation techniques.

16.3 Fake News Mitigation

Fake news mitigation has been very challenging for the researchers. The literature on mitigation techniques can be categorized into the following four main categories.

1. **Influence Blocking**: This approach aims to identify a minimal set of users whose immunization will minimize the spread of misinformation in the network. We will discuss several proposed methods to choose an optimal set of users given various parameters, such as the set of source nodes who starts the spread, target nodes that should be saved, the deadline of the rumor (after that time, the rumor will not be effective, for example, the rumor related to elections are effective only before the election), and so on.
2. **Truth Campaigning**: Another approach to combat the inverse impact of fake news is to aware users about the true information. Research studies show that if users are exposed to both true and fake information, users tend to believe in true information and also reduce sharing of fake information further [44, 45]. In this section, we will cover approximation, greedy, and heuristic-based truth campaigning techniques.
3. **Mitigation Tools**: Researchers have designed tools that notify the user about the credibility of the information to minimize the spread of fake information. These tools are intended to mitigate the flow of false information over the network.

4. **Miscellaneous**: In this section, we will cover the related studies which do not fall under any of the above categories. This section mainly includes social studies that have been performed to understand why a user shares misinformation, what kind of environment prompts a user to spread it further, and how we can make people aware so that they do not share further the fake information.

16.3.1 Influence Blocking

Influence maximization is a well-studied problem in network science that focuses on identifying a minimal set of initial adopters for maximizing the influence spread in a given network [22, 46]. However, in the case of fake news propagation, we aim to find out a minimal set of users whose immunization will minimize the fake news spread. This is referred to as *influence minimization* or *influence blocking* problem.

16.3.1.1 Problem Formulation

Let's assume that $G(V, E)$ is the given graph, M is the set of nodes that start spreading misinformation, and k is the number of blocked/immunized nodes. In some research works, k is also referred to as budget or cost as they aim to minimize the inverse impact with the given budget k, and there is a fixed cost of blocking a node in the network. $\pi_{G(V,E)}(M)$ represents the number of affected nodes in the given graph $G(V, E)$ if set M ($M \subset V$) spreads misinformation. In influence minimization, given $G(V, E)$, set M, and the budget constraint k, the aim is to identify a subset T of nodes of size k from $V - M$ such that $\pi_{G(V,E)}(M)$ is minimized.

The first feasible intuitive approach toward the solution is a greedy method. In the greedy method, we select a node that minimizes the influence in the network and iteratively keep adding node that further minimizes the influence until the required number of nodes are chosen. The method is explained in Algorithm 1.

Another popular approach to solve such problems is using heuristic methods. There are several centrality measures [47] in the literature that can be used to identify influential users in a given network. Once the influential users are known, the top-k users having the highest centrality value can be selected to be immunized. Amoruso et al. [12] presented a two-step heuristic approach that works as follows, (i) it first identifies a set of users containing the most likely source users of the spread, then (ii) places a few monitors to block the misinformation spread in the network. A heuristic method is desired as the first step of source identification is NP-hard, and the second step of Monitor placement is $\#P-$complete. Cao et al. [48] also studied information blocking maximization for rumor containment (CIBM) with the given cost budget in the e-commerce environment. The proposed problem is NP-hard with the submodular and monotone characteristics. The authors proposed a community dividing algorithm that chooses a set of nodes to block from both the inactive and

Algorithm 1: InfluenceBlocking-GreedyApproach($G(V, E), M, k$)

Input : $G(V, E)$ is the given graph,
M is the set of nodes who starts spreading misinformation,
k is the number of selected nodes.
Output: T is the set of selected nodes of size k

$T = \phi$;
for *i in range*$(1, k)$ **do**
  for *each node v in* $\{V - M - T\}$ **do**
    $s_v = \pi_{G(V',E)}(M)$, where $V' = V - T - \{v\}$;
  end
  $T = T \cup argmin_{v \in \{V-M-T\}}\{s_v\}$;
end
Return T;

the negative nodes and optimize the containment of the rumors using community structure.

In real-world networks, users are organized into communities. The community information has been used to propose more effective rumor containment methods as it can provide information about the set of nodes going to be affected. Apart from this, in real life, there also might be situations where the aim is to save a given community from the negative effects of rumor. Zheng and Pan [49] proposed the Minimum Vertex Cover Based Greedy method to contain the rumor when it is originated from a given community (C_R). The proposed solution aims to find out a subset of users to be blocked so that the effect of the rumor is minimized in C_R, and also, the total number of affected nodes is not greater than a given limit. Wu et al. [50] also proposed a community-based dynamic blocking strategy to control the rumor spread. The proposed method first computes each node's influence within its community and also within the entire network and then, integrates this information to choose top-k nodes to block the effect of the rumor.

Fan et al. [51] also analyzed the Least Cost Rumor Blocking (LCRB) problem in which the rumor originates from a given community C_R and a minimal subset of the nodes are blocked to minimize the number of infected people in C_R's neighboring communities. The authors create a set of vertices, named bridge end set, containing vertices where every vertex has at least one in-neighbor in the community C_R and can be reached by the rumor starters. They showed that the LCRB-P problem, in which a given fraction of bridge end nodes should be protected, is submodular and presented a greedy approach-based solution with $(1 - 1/e)$-approximation. Pham et al. [52] studied the Targeted Misinformation Blocking (TMB) problem, in which the aim is to identify the optimal set of users whose immunization will minimize the fake news spread by the given threshold value γ. They proved that the problem is $\#P - hard$ problem of the LTM model. The authors further presented a greedy method to provide the solution within $1 + ln(\gamma/\epsilon)$ ratio of the optimal solution.

Wang et al. [53] studied rumor minimization where each user has a tolerance time threshold, and the utility of the network is decreased if the immunization time of a user (immunization time denotes the duration for which the user is considered immunized) exceeds its tolerance threshold. The influence is propagated using the dynamic Ising propagation model, which considers both the popularity at the global level and the attraction of individual rumor topic. The authors proposed a greedy and dynamic blocking solution based on survival theory and the maximum likelihood principle. Yao et al. [54] studied influence minimization from a topic modeling perspective where the influence probability from one user to another depends on the topic. A piece of prominent misinformation news can have multiple topics, and its propagation can be modeled using a Multiple Topics Linear Threshold model [25, 55]. The problem of selecting k-users for minimizing the impact of multi-topic misinformation is NP-hard. The authors proved that the objective function for this is monotone and submodular. They further showed that an approximation method with the factor of $(1 - 1/\sqrt{e})$-approximation outperforms heuristic methods.

Yang et al. [56] solved two variations of influence blocking called Loss Minimization with Disruption (LMD) and Diffusion Minimization with Guaranteed Target (DMGT) using Integer Linear Programming (ILP). In LMD, they focus on finding out the set of initial adopters whose cost is more than the given cost, but the spread of influence (total cost of active nodes) should be minimized. They presented heuristic methods for LMD where k nodes with the minimum PageRank or degree are selected and showed that the proposed methods perform well on real-world networks. In the DMGT problem, the authors focus on finding the minimal set of initial spreaders from a given set of initial nodes such that all target nodes are influenced while the spread over the network is minimized. The authors presented a greedy solution in which, at each iterative step, a node is selected based on the maximal marginal gain. However, in this work, the authors aimed to minimize the influence, but it is different from real-life scenarios of fake news spreading where the seed nodes are already infected and start spreading the false information over the network while we try to block the nodes or edges to minimize its impact in the network.

In edge-blocking problems, a minimal set of edges are selected that will not propagate the information further to minimize the spread of misinformation. These are referred to as edge-blocking or edge-deletion-based solutions as the edges are inactive for propagating the information. Wang et al. [57] studied target influence minimization using edge-blocking, where the focus is to save a given set of nodes, namely, the target users, from the misinformation. The authors showed that under the constrained budget, the problem is NP-hard and proposed a greedy method having $(1 - 1/e)$-approximation. However, for an unconstrained budget, the authors provided an optimal solution. Both solutions have been verified on real-world social networks for their effectiveness and efficiency. Kimura et al. [58] also studied the link blocking method for the average contamination minimization and the worst contamination minimization problem. The average contamination degree of a network is the average of influence degrees, and the worst contamination degree is the maximum of influence degrees of all nodes in the given network.

Yan et al. [59] proved that the Rumor Spread Minimization (RSM) problem that removes edges from a given edge set to minimize the spread of the rumor is not submodular. The authors presented submodular lower bound and submodular upper bound for RSM objective function and designed a heuristic solution for approximating the given objective function. Tong et al. [60] presented the NetMelt method that removes k-edges from the network to minimize the rumor impact. The proposed solution identifies the edges that should be removed using the eigenvalues of the graph's adjacency matrix. Other works based on link removal for rumor containment include [61–64].

He et al. [65] studied the Mixed Generalized Network Security (MGNS) model by using a combination of both node blocking and edge removal. The authors contribution are (i) a polynomial time $(d + 1)$-approximation solution for identifying the optimal set when the rumor spread till d-hops, (ii) derived an $O(logn)$-approximation when $d = \infty$, and (iii) a polynomial time 32-approximation on bipartite graphs when $d = 1$. The paper also contains other results for regular graphs and tree structure.

In dynamic real-world networks, there might be some unforeseen conditions. The discussed methods that identify a set of k-nodes to be blocked based on a network snapshot may not work for the dynamic networks. Shi et al. [66] proposed an adaptive solution for rumor containment in dynamic networks where k nodes are chosen at each iteration until the budget is exhausted.

All the above-discussed methods assume that users receive misinformation from their neighbors; however, the users can also browse the misinformation on their own. Zhang et al. [67] considered this proactive behavior of users while controlling the rumor, called users' Browsing-based rUmor blocK (BUK) problem. The authors modeled the influence propagation using a random walk and showed that BUK is submodular. They proposed a greedy solution that approximates BUK with the $(1 - 1/e)$-approximation.

Table 16.1 mentions the important work in this direction with the considered parameters. In Table 16.1, 'Complexity' represents the complexity of the studied problem, k (a fixed number of blocking nodes) shows if the work considers that the number of nodes to be blocked are fixed or bounded above by a given budget, 'Target Nodes' shows if the work aims to save a fixed set of nodes from getting infected by fake information, 'decontamination ratio' shows if the authors aim that $\theta\%$ of the nodes should not get infected, 'time constraint' shows if the work considers some time constraints like the infection is detected after time t or login time of users, etc., 'Deadline' refers if the work considers that the rumor will die out after some time, 'Diffusion Model' mentions the basic diffusion method used in the work, such as Independent Cascade Model (ICM), Linear Threshold Model (LTM), or compartment models, 'Baseline methods' mentions the methods that authors have used to compare their work with. Note: If in a work, the authors have not addressed any of the constraints, the table cell is left blank.

16.3.2 *Truth Campaigning*

In influence blocking methods, the nodes or edges are blocked from further spreading the fake news to reduce its impact. However, the inverse impact can also be reduced

Table 16.1 Influence blocking methods

References	Complexity	k	Target nodes	Decontamination ratio	Time constraints	Deadline	Diffusion model	Baseline heuristics
Cao et al. [48]	NP-Hard	✓					LTM	Infected Maximum Out-degree , random
Fan et al. [51]	LCRB-P: submodular, LCRB-D: NP		✓	✓			ICM	MaxDegree, Proximity
Wang et al. [53]					✓		ICM	Greedy
Yao et al. [54]		✓					ICM	TopICM-aware Betweenness, TopICM-aware Out-degree
Zheng and Pan [49]				✓			ICM	MaxDegree, Betweenness Centrality and K-core
Amoruso et al. [12]	NP-hard	✓	✓				ICM	[68]
Zhang et al. [68]	#P-complexity						ICM	
Wang et al. [69]		✓					ICM	Out-Degree, Betweenness, PageRank
Yan et al. [70]	Objective function is not submodular	✓					ICM	Max Out-Degree, Betweenness Centrality and PageRank
Tan et al. [71]							SIR	Most susceptible neighbors, LRIE [72]
Gai et al. [73]	NP-Hard	✓					ICM	
Zhang et al. [67]	NP-Hard	✓					Random Walk	Highest Degree
Cui et al. [74]	NP-hard	✓			✓		ICM	DAVA [75]
Wang et al. [57]	Edge blocking: NP-Hard (constrained budget)	✓					LTM	Random, Largest-weight edges

(continued)

Table 16.1 (continued)

References	Complexity	k	Target nodes	Decontamination ratio	Time constraints	Deadline	Diffusion model	Baseline heuristics
Kimura et al. [58]	Edge blocking	✓					ICM	link Betweenness, link out-degree
Yao et al. [76]	Edge blocking	✓					ICM	link Betweenness, link out-degree
Yan et al. [59]	Edge blocking: Not Submodular	✓					ICM	Bond Percolation Method [77], K-edge deletion [60], Out-degree, Pagerank, random
Nandi and Medal [78]	Edge blocking	✓					SIR	Betweenness, Greedy, random
Wang et al. [79]	Edge Blocking NP-Hard (budget Constraint)	✓	✓				LTM	Random, Highest edge-weight
Wu et al. [50]	O (l + Tmn), l #of edges, T iterations, m communities, each community has n nodes	✓			✓		ICM	normal propagation [53], greedy, positive cascades blocking [80], and dynamICM blocking
Zhang et al. [81]	NP-hard	✓					SIR	Random, degree, Pagerank, NETSHIELD [82]
Pham et al. [83]	#P-hard under LTM and NP-hard under ICM Model	✓		✓			LTM and ICM	Greedy, degree, PageRank, DAVA [81]

by making the users aware of true information. This will help users to understand the news from different perspectives and build an unbiased opinion about the given news topic. It will also influence users' decisions regarding sharing, and they will be motivated to further share the true information and not the rumor. This is a more feasible approach in real life, and it will reduce the impact of fake news over the network.

Garrett's research [84] showed that users do not abandon news articles showing information from a different perspective from what they believe. The study also showed that users spend even more time in exploring the opinion-challenging news articles. Van der Linden et al. [85] showed that the public attitude could be inoculated about climate change by providing facts against misinformation. Cook et al. [86] found that inoculating messages that explain either the flawed arguments used in the misinformation or the scientific consensus about the topic are more effective in neutralizing the adverse effects of misinformation. Tanaka et al. [44] further examined this and found that if people are exposed to true information before the rumors than after, it will significantly reduce the rumor spread. Ozturk et al. [45] studied this problem in a more realistic way, in which they focus on the intent of sharing/retweeting the rumors and counter-rumors than the number of people resharing that. They showed that displaying rumors and non-rumors, at the same time, will help reduce the rumor spread and is also more feasible to implement. All these studies support the concept that truth campaigning methods for controlling fake news work better in real life.

Initially, counter-campaigning or defensive mechanisms were studied to reduce the infected nodes in the case of infection or virus spreading. Nicol and Liljenstam [87] studied active defense mechanisms against Internet worms and showed that by starting defense with enough nodes (converted to counter-worms), any desired fraction of the nodes could be protected from getting infected by the worm. Researchers have studied several competitive strategies where two companies market their products and compete to maximize their influence [88, 89]. Kostka et al. [90] showed that selecting the starting node in a two-player rumor maximizing game is NP-complete for both the players. They further showed that finding the approximate solution for first player is also NP-complete, and being the first player is not always advantageous as the second player can infect more nodes than the first player even when they follow optimal strategies. Similar approaches can be used to combat the negative impact of fake news spreading in a network, and we will discuss them further.

16.3.2.1 Problem Formulation

Let's assume, $G(V, E)$ is the given graph, M is the set of nodes that start spreading the misinformation, and k is the number of nodes to be chosen. $\pi_{G(V,E)}(M, T)$ represents the number of affected nodes from misinformation in graph $G(V, E)$ if set M ($M \subset V$) spreads misinformation and set T ($T \subset V - M$) spreads the true information. In most of the truth campaigning methods, given $G(V, E)$, set M, and k, the aim is to select a subset T of nodes of size k from set $V - M$ such that

$\pi_{G(V,E)}(M, T)$ is minimized. The greedy method is explained in Algorithm 2. In the greedy method, we select a node to start a truth campaign for minimizing the influence of misinformation in the network and iteratively keep adding nodes, which further minimize the negative influence until the required number of nodes are selected.

Algorithm 2: TruthCampaigning-GreedyApproach($G(V, E), M, k$)

Input : $G(V, E)$ is the given graph,
M is the set of nodes who starts spreading misinformation,
k is the number of selected nodes.
Output: T is the set of selected nodes of size k for truth campaigning

```
T = φ;
for i in range(1, k) do
    for each node v in {V − M − T} do
        s_v = π_G(V,E)(M, T ∪ v) ;
    end
    T = T ∪ argmin_{v∈{V−M−T}}{s_v};
end
Return T;
```

There are several constraints that have been considered while studying truth campaigning techniques by different works. For example, some works consider that the rumor has a deadline, and after that, it is not effective. Some other works consider that there is a group of users that might be prospective truth campaigners, and k users should be selected from that group. Another interesting variation is where our aim is that a given set of target users should be aware of true information by the end of the deadline. A summary of constraints for different works is given in Table 16.2. Next, we discuss some of these works in detail.

Budak et al. [91] showed that selecting a minimal group of users to disseminate "good" information for minimizing the influence of "bad" information is NP-hard. The authors proved the problem to be submodular and presented a greedy approach-based solution. They further proposed the Multi-Campaign Independent Cascade Model (MCICM) that models the propagation of two cascades (good and bad) and assumes that when both good and bad information try to affect a user at the same time, the user will accept the good information. Once a node accepts any of these information, it will never change its status in the future. The experimental results showed that in most of the cases, selecting nodes based on degree centrality, i.e., an easy-to-compute centrality measure, performs well with respect to the greedy approach.

Tong et al. [92] proved that the misinformation containment problem cannot be approximated within a factor of $\Omega(2^{log^{1-\epsilon}n^4})$ in polynomial time unless $NP \subseteq DTIME(n^{polylogn})$. They proposed multiple cascade priorities called homogeneous cascade priority (both misinformation and positive cascades have the same priority), M-dominant cascade priority (misinformation cascade has priority), and P-dominant cascade priority (positive information cascade has priority) and stud-

Table 16.2 Truth campaigning methods

References	Complexity	k	Prospective mitigators	Target saved nodes	Decontamination ratio	Time constraint	Deadline	Diffusion type	Baseline heuristics
Nguyen et al. [93]					✓	✓		LTM and ICM	
Budak et al. [91]	NP-Hard		✓			✓		ICM	Degree, Early Infectees, Largest Infectees
Tong et al. [92]		✓	✓					ICM	High Weight, Proximity, Random
He et al. [102]	NP-Hard	✓						LTM	Degree, Random, Proximity Heuristic
Yang et al. [103]		✓						ICM	MaxDegree, DegreeDiscount [104], BetweennessCentrality, and LocalProximity
He et al. [105]	APX-Hard				✓			ICM	PIDS Random
Litou et al. [106]	NP-Complete	✓						Dynamic-LTM	degree, greedy
Peng and Pan [17]	NP-Hard	✓						ICM	degree, proximity, random, greedy
Zhu et al. [107]	NP-Hard	✓						ICM	degree, proximity, random, greedy
Lv et al. [108]	NP-Hard	✓						ICM	degree, proximity, random
Simpson et al. [109]		✓						ICM	
Fan et al. [110]	NP-Hard	✓					✓	ICM and LTM	no implementation shown
Vu and Hoang [111]	NP-Hard	✓				✓		LTM	max degree and random
Zhang et al. [112]	NP-Complete	✓						LTM	random, maxdegree, maxgreedy, mingreedy
Zhang et al. [113]		✓					✓	LTM	random, maxdegree, pagerank, greedy

(continued)

Table 16.2 (continued)

References	Complexity	k	Prospective mitigators	Target saved nodes	Decontamination ratio	Time constraint	Deadline	Diffusion type	Baseline heuristics
Yang et al. [24]		✓						LTM1DT	random, maxdegree
Hosni et al. [114]	NP-Hard	✓		✓				ICM	maxdegree, random
Tong et al. [115]	NP-Hard	✓						ICM	greedy, proximity, random
Song et al. [95]	NP-Hard	✓					✓	ICM	PageRank, LSMI [116], Largest Infectees
Wu et al. [117]			✓			✓		SCTIR	Random, MaxDegree, Nearest Point
Fan et al. [118]	NP-hard	✓						Asymmetric-trust (AT)	Random, MaxDegree
Saxena et al. [13]		✓	✓				✓	OF	Random, Highest Out-degree, highest weighted out-degree, minimum distance, TIB [95], TMB [83]
Lin and Dai [119]		✓						ICM	CMIA-O Peng and Pan [17], Greedy, HD, Proximity, and Random

ied them on real-world networks. They also proposed an approximation algorithm using the upper lower bound technique and performed the experiments to show the near-constant approximation ratio. The proposed method can be further improved by proposing a better approximation for the upper and lower bound.

Nguyen et al. [93] studied β_T^I problem where the aim is to identify a minimal set S of influential vertices who will spread the correct news so that the expected decontamination ratio is β after time T, given that the users from set I will spread the fake information. The authors proposed Greedy Viral Stopper (GVS) method, where the nodes are iteratively selected for spreading good information so that the total number of decontaminated nodes will be maximized. They also provided an upper bound for the proposed solution with respect to the optimal solution mathematically. The authors assumed that good information and misinformation both propagate using the same propagation model with the same infection probabilities. They also assumed that once a node receives both the information, the node will believe in good information and will spread it further. The proposed approach is further modified to get a better solution set for community structured networks, where the nodes are greedily selected for decontamination from each community until the β fraction of the nodes from the community is decontaminated, hence achieving the β fraction in the entire network. The proposed community-based solution works better and faster because a user has a higher probability of infecting the users from its own community than in other communities due to the effect of homophily [94].

In real-world scenarios, the impact of fake news dies out after some time. Song et al. [95] considered this as a parameter and proposed a solution to identify appropriate truth campaigner given the rumor deadline. The proposed solution has two steps: (i) find out the set of the nodes that might be infected by the rumor and compute the threat level (the number of nodes that might be affected by the given node) of each node belonging to the set, and (ii) use Weighted Reverse Reachable (WRR) trees to greedily select k truth campaigners that will save most of the nodes from the misinformation in the given deadline. In the above-discussed works, it is assumed that the selected truth campaigners will propagate true information; however, in real life, a user might not be willing to start a truth campaign or post a counter-message on the profile. A more realistic approach will be when we know the inclination of a user towards starting the campaign. Saxena et al. [13] considered this realistic problem, where a prospective set of mitigator users, who might be interested in mitigating misinformation or can be controlled by authorities to spread some specific message, is already provided. The authors aim to identify a set of k-truth campaigners from the given set of mitigators, knowing the rumor starters and the rumor-deadline. The proposed solution, first, computes the mitigation power of each prospective truth campaigners with respect to the seed set of rumor starters and then chooses the top-k nodes having the maximum mitigation power. The proposed work outperforms state-of-the-art methods and well-studied heuristic methods. Saxena et al. [96] also proposed a method, called k-TruthScore, that identifies k truth campaigners to minimize the inverse impact of misinformation in the presence of strong user biases.

Xu et al. [97] presented a new method to combat fake news spreading where they deploy decoy users in Online Social Networks (OSNs) and connect them with some selected active users of the network. To deploy the decoy user, the authors first select a minimal set of nodes and monitor them. Once the nodes are selected, the OSN service provider consults these users. Each of the agreed users will be connected with two decoy nodes. These decoy users regularly observe the propagation of information in the network to detect the worm or misinformation propagation. The authors do not use these decoy nodes to control the worm propagation, though the work is extendable to address such problems. They showed that the problem of implanting the decoy nodes in such a way that all the nodes are covered within r hops from the decoy users is equivalent to the extended dominating set problem, which is NP-complete. However, one can propose greedy or heuristic algorithms to identify a minimal set of users for connecting with decoy users.

Wilder and Vorobeychik [98] used a game-theoretic approach for defending the fake news spreading through different channels during the elections. They studied the attacker-defender strategies using the zero-sum game where the attacker aims to subvert the election by spreading the fake news, and the defender targets for minimizing the attacker's impact. The problem is studied using two different population structures, (i) disjoint populations where the voters are partitioned by the channels, and (ii) non-disjoint populations where the voters can be reached by multiple channels. The authors show that in the case of non-disjoint populations, computing an optimal defender mixed strategy is APX-hard. The experimental results show that the proposed defender strategies provide near-optimal payoffs and show that the election can be defended by the modest use of limited resources and enough information about the voters' preferences. Tanınmış et al. [99] also discuss a game-theoretic approach where the first player targets to minimize the impact of false information and second player targets to maximize it. The authors proposed two methods, (i) matheuristic, and (ii) greedy heuristic, for solving the misinformation minimization from the first player perspective.

Farajtabar et al. [100] presented a point process-based mitigation method using reinforcement learning framework. The presented solution aims to optimize the actions for the maximum total reward under the given budget constraints. The presented solution was applied in real time on the Twitter network to combat a fake news campaign that was started for the research purpose and showed promising results. Yan et al. [101] observed malware propagation in OSNs to analyze the effects of initial infection, social structures, user click probability, and the patterns of activities. They further studied user-oriented and server-oriented defense schemes and their effectiveness against malware spreading that can be applied in real-life applications. In Table 16.2, we compare different works based on the different parameters they have considered.

The discussed methods can mitigate the spread of misinformation by making users aware of correct news, but the feasibility of applying them to OSNs is still an open research question. In OSNs, the challenge is to convince the users for truth campaigning, and it further opens up a new line of research that calls for an in-depth analysis of OSN users' characteristics and behavior.

16.3.3 Mitigation Tools

Psychology-based studies show that users intend to believe in true information when exposed to both true and fake information. There are several mitigation tools to compute and display the news's credibility to help a user decide to share the information further. Park et al. [120] designed a service called NewsCube, which provides users with different points of view on a news event of interest. That way, the users can study and understand the news articles from different perspectives and conclude an unbiased opinion on their own. To evaluate the usefulness of NewsCube, the authors selected 33 participants, including students, researchers, and administrative staff (16 males and 17 females), and performed a controlled experiment. 72% of participants said that they felt like reading multiple articles on NewsCube; however, 70% of users actually read them. Except for three participants, all users said that they found it important to read and compare the multiple articles. Sixteen participants said that NewsCube helped them to build an unbiased viewpoint about the news event, two were negative, and the rest were neutral as they expressed that the session duration of the experiment was not long enough to evaluate the effectiveness of the product. Thus, we can see the usefulness of such tools in the mitigation of fake news spreading.

Ennals et al. [121] designed "Dispute Finder", i.e., a browser extension to notify users if the information they are reading is disputed by other sources and display a list of news articles supporting other viewpoints. Users can also add disputed claims to the database that will further help other users with more information. The authors also interviewed the participants who used the tool during the experiment session, and most of them were positive regarding the tool. Hassan et al. [122] proposed a system called FactWatcher that helps users by providing facts that act as leads in news stories. The system also provides additional services like fact ranking, keyword-based fact search, and fact-to-statement translation. ClaimBuster [123] is a fact-checking system having different components focusing on different steps of fact-checking, such as (i) Claim Monitor that collects the data from different websites and social media, (ii) Claim Matcher that finds similar facts on different fact-checking websites, (iii) Claim Checker collects search results on the topic from the web, and (iv) a fact-Check reporter that prepares a final report to display to the user. Ratkiewicz et al. [124] designed a web service called Truthy that collects the tweets related to U.S. political election and detects astroturfing, smear campaigns, other misinformation, and abusive behaviors using tweet features, such as hashtags, retweets, URLs, and mentions.

Figueira and Oliveira [125] briefly discussed other fact-checking algorithms such as "FiB: Stop living a lie" that is a browser extension, Facebook fake news detection approaches, etc. Guha designed a browser extension called "Related Fact Checks" that displays related facts of the searched news item to the user [126]. de Alfaro et al. [127] designed a tool for Twitter called Truth Value,[1] where they assign a reputation score to each news post. It also provides the voting facility to users so that they can vote news articles as reliable or fake/misleading, and users' votes are weighted

[1] The project is available at https://truthvalue.org.

depending on their reputation. The authors also designed a Twitter bot for responding to users' inquiries and a browser bookmarklet to quickly display the scores of sites while users are accessing them on the browser. Pavleska et al. [128] studied the accuracy and efficiency of European fact-checking organizations; results show that they still need a whole lot of improvements in their workflow and functionality so that they can provide better recommendations to users. There are several other tools to detect fake news including FakeNewsTracker [14], PolitiFact [15], FactCheck.org [129], snopes.com [130], etc.

Online networking websites use crowdsourcing for controlling the spread of fake news by allowing users to flag the story if it has misinformation or fake news. Once the story is flagged by enough people, it will be sent for fact-checking to some trusted third party, and if found containing fake news, the story is marked as disputed. Each fact-checking call has a cost, so it is required to determine an optimal trade-off between the total number of flags and fact-checking calls. If the threshold for the number of flags is high and later on information is found to be false, then by that time, the news already would have impacted a good amount of the crowd. Similarly, if the threshold for the number of flags is small then, the story may or may not be false as users have their own biases about different topics and might have marked it as fake based on their biases and thus, the cost of fact-checking will be wasted if the information was not actually fake. Kim et al. [131] proposed an algorithm called "CURB" to find an optimal trade-off between the number of flags and fact-checking, and it optimally selected which stories should be sent for checking to minimize the misinformation spread. The proposed work can be further improved by considering more realistic factors, such as all users not being equally good at flagging misinformation, considering the user credibility while computing the total flagged weight of the post, the weight of a user by considering how many users might be influenced by the information shared by that user, etc. Facebook also provides a feature where users can flag a news article as disputed if they think the information is not correct, and researchers have studied the impact of flagging on further spreading of the article. A website called Newport Buzz published an article that how thousands of Irish were brought to the United States to be enslaved, and this story was flagged as disputed by Facebook [132]. However, the editor of the website reported that the traffic on the article was increased after the Facebook warning, although it doesn't show that more people believed it to be true as the users might have visited the article out of curiosity. Another example is RealNewsRightNews, whose owner mentioned that a disputed flag on one of the articles has no effect on the incoming traffic. These examples show that we need better education and awareness systems for users so that the traffic to fake news websites can be reduced, and it will surely have an impact on users' behavior of posting fake information.

Vo and Lee [133] showed the existence of Guardian users on Twitter who correct fake news or misinformation by replying or providing URLs having the right information in online discussions. The guardians who directly reply or correct the information are called direct guardians, and the people who retweet the corrected information are called secondary guardians. These guardian users can use different fact-checking tools and can post the correct information on social networking websites. The authors also proposed a fact-checking URL recommendation model that will help guardians by making the fact-checking process easier.

The main aim while designing these tools is that the tools should be easily accessible to users so that they can constantly add new information, and the outcome of the news is updated in real time. These tools help users to get an unbiased perspective, but at the same time, if users start trusting these tools blindly, then the wrong outcome of the tool can also affect their viewpoints. Nguyen et al. [134] studied how the correct and incorrect prediction of fact-checking tools impact the user's decision accuracy. They showed that though users improve their accuracy by interacting with these tools, the tools' incorrect predictions affect the accuracy of human judgment negatively. Thus, it is required that these tools should convey the right probability of the credibility of a news article. Lease discusses the designing of Fact-Checking tools from the Information Retrieval (IR) point of view and talks about the requirements, challenges, and expected outcomes; more details are available at [135].

16.3.4 Social Scientific Studies

In this section, we cover all other studies that have been performed to understand several other aspects of fake news spreading, such as why people spread fake news, how they spread, how we can change their attitude towards it, and so on. Psychologists have performed studies to understand how we can immunize people against fake news spreading. Rozenbeek and van der Linden [136] designed a game called "fake news game," where they asked users to create fake news articles about a strongly politicized issue. Each group of users could create a fake article from one of the four different perspectives called (i) Denier, whose motive is that a topic should look insignificant and small, (ii) Alarmist, whose focus is that the topic should look large and problematic, (iii) Clickbait Monger, whose focus is that the article should get as many clicks as possible, and (iv) Conspiracy Theorist, whose aim is to distrust official mainstream narratives and make the audience to follow their suit. In the experiments, the participants (total 95) were given a structure to create the article from the perspective of the role they chose. The authors observed that after playing this game, the users were less affected by fake news articles. The authors suggested that early media education might help in making people aware of different perspectives and how they arise so that people can fight against the risk of misinformation. Jang and Kim [137] studied fake news spreading from Third-Person Perception (TPP) and showed that users with greater TPP do not support media regulation methods, but they show significant support for media literacy intervention methods. This study can be used as a base method to design media literacy intervention techniques to make the users aware of different TPPs. Pennycook and Rand [138] showed that the people with prior exposure to fake and real news could differentiate fake news with higher accuracy, and it is correlated with the analytical thinking power of the user.

Kanoh [139] studied the impact of eating and drinking habits on fake news spreading and showed that people spread fake news more when they are in a comfortable position, like while eating or drinking. This also supports the fact of the extensive spread of fake news on social networking websites as people come there for entertainment in their free time. Ghenai and Mejova [140] collected Twitter data of ineffective cancer treatment and studied the properties of users spreading them versus the users genuinely interested in cancer. They proposed a classifier that can identify users spreading misinformation about ineffective medical treatments with more than 90% accuracy using user attributes, writing style, and sentiment. This method can be used to detect such users in online social networks, and further actions can be taken to control the spreading.

Fake news spreading can also be controlled by assigning a credibility score to users so that other users can carefully examine the validity of news posts shared by low-credibility users. Balmau et al. [141] check the truthfulness of a news item using a team of fact-checkers and use this information to compute the trustworthiness of the users based on their shared news items; then, they use a Bayesian model to compute the truthfulness of future news items using the trustworthiness of the users. Future work can focus on better techniques to control fake news spreading from low credibility users; this is still an open research question and can be looked further in-depth.

In online social media, people prefer being connected with other like-minded users, and it gives rise to the communities following the same belief about a topic called echo-chambers. Nguyen et al. [142] proposed a technique to disrupt echo-chambers. In the proposed method, authors first divide the users into two groups based on political points of view, where one group consists of Democrats and another group consists of Republicans. If there is any discrepancy about a topic in the groups, the famous posts/content about that topic will be picked from one group and will be recommended to another group of users. Thus, the proposed method will make users aware of different viewpoints on a topic and will disrupt the growth of echo-chambers. In the real world, a network is divided into different community structures based on different topics as two users might agree on one topic while having completely different viewpoints about some other topic. The proposed method can be further extended to apply to real-world networks using a topic-based community structure, but the complexity and feasibility of such an approach is still an open question. Unlike this work, Colleoni et al. [143] used n-grams, and tf-idf [144] for applying a supervised learning method to cluster Twitter users in republican and democrat groups.

Blair [145] performed a study during 8–9 May 2017 on 2994 participants recruited from Amazon Mechanical Turk (MTurk) where 54% were female, median age group was 25–34, 55% had a bachelor's degree or higher, and 32% identified themselves as lean Republican or Republican, and 58% identified themselves as lean Democrat or Democrat. In the experiment, the author observed that the news with the "Rated false" tag had lower perceived accuracy than the news with the "Disputed" tag; however, it had no impact on the perceived accuracy of the unlabeled news.

The work in this area can be further extended using different sources of fact-checking websites and displaying the results on the portal. One can further study how people perceive different fact-checking resources. Further details about the related literature and limitations of automated fact-checking can be seen at [146].

16.3.5 Datasets for Mitigation Studies

The influence blocking techniques and truth campaigning techniques have mainly been verified on existing network datasets. We have summarized some of the datasets that have been used.

1. Facebook: One Facebook snapshot containing 63392 nodes and 816831 edges [147] is used by [148]. Budak et al. [91] performed experiments on four snapshots of Facebook, two for each of (i) Santa Barbara regional network, and (ii) Monterey Bay regional network. One snapshot of Facebook having 4,039 nodes and 88,234 edges is used in [50, 54, 76]. Another Facebook dataset (43953 nodes and 262631 edges) build on using wall posts is used in [13].
2. Twitter: The extracted subset of Twitter has been extensively used to study fake news mitigation. In [149], the authors collected a Twitter dataset of 554k nodes and 4.29M edges, and it is used in [95]. One Twitter snapshot [150] having 81306 nodes and 1768149 edges is used in [13, 57]. A Twitter subgraph extracted using Higgs-Boson [20] related tweets is used in [92]. Other Twitter snapshots are [71, 79, 151].
3. Gowalla [152]: This dataset is extracted from Gowalla, i.e., an online location-based social network. This dataset was collected over the period of February 2009–October 2010 and consisted of 196,591 nodes and 950,327 edges. This is used by [67, 95].
4. Weibo: [149] crawled *weibo.com* and collected a dataset of 1.02M nodes and 166.7M edges. This dataset is used by [95]. One another snapshot having 23086 nodes and 183549 edges is used in [53].
5. Foursquare: [149] collected Foursquare dataset of 4.9M nodes and 53.7M edges and it is used in [95].
6. Wiki-Vote [153]: This network contains the data of Wikipedia voting from its beginning until Jan 2008. It contains 7,115 nodes and 103,689 edges. This dataset is used in [12, 57, 59, 79, 151, 154].
7. Gnutella08 [155, 156]: This dataset is extracted from Gnutella peer-to-peer file-sharing network where nodes are the host users and edges are the connections between them. One snapshot having 6301 nodes and 20777 edges is used in [12], and one another used in [81, 83].
8. Slashdot: This is a friendship dataset of users from the Slashdot website, having 13,182 nodes and 30,914 edges. This is used in [59, 67].
9. Google+: This directed network is extracted from Google+, and a directed edge between two nodes indicates that one node has the other node in its circles. It contains 23,628 nodes and 39,242 edges. This is used in [59, 151].

10. Epinions [157]: This is a trust network extracted from Epinions.com, a site for general consumer reviews. It contains 75879 nodes and 508837 edges. This dataset is used in [12, 55, 57, 71, 74, 79, 154].
11. Enron [158]: This is an email communication network. This dataset contains 36692 nodes and 367662 edges. This is used in [51, 67, 69].
12. NetScience [159, 160]: This is a co-authorship network of Network Science researchers. It contains 1588 nodes and 2742 edges. This dataset is used in [24, 52, 56].
13. Autonomous systems graph [161]: This is who-talks-to-whom network extracted from the traffic flow data of routers. This dataset is collected over a period of 785 days from November 8,1997 to January 2, 2000, and contains 6.4k nodes and 12.5k edges. It is used in [52].
14. Hep-Th [22]: This is a co-authorship network created using the publications in the high-energy physics theory section of the arXiv. This is used in [52, 56, 66, 83, 102, 119, 148, 154].
15. Hep-Ph [161]: [92] used Hep-Ph citation graph having 34,546 nodes and 421,578 edges. Other versions are used in [51, 55, 59].
16. NetPhy: [66, 102] used co-authorship network extracted from papers in physics section [104, 119].
17. Brightkite: It has 197k nodes and 950k edges, and is used in [74]. Other versions of Brightkite are used in [81, 83].

16.4 Conclusion

Fake news spreading, especially on social media, has been a real threat to society, given its recent impact on major events. Fake news and misinformation have become an integral aspect of online social networking, where users influence each other with their opinions, beliefs, and perspectives. In this chapter, we first discussed how fake news spreads on social media. A better understanding of its propagation will help in designing efficient techniques for fake news detection and mitigation. For fake news mitigation, there have been proposed influence blocking and truth campaigning methods, which focus on identifying an optimal set of users that either can be immunized to block the influence or can propagate the true information in the network, respectively. We have discussed the state-of-the-art techniques for fake news mitigation. We further discussed the tools that have been designed to assist users by mentioning whether a given news is fake or not and with what probability.

Currently, the literature contains generalized methods to combat all kinds of false information. However, Haiden and Althuis [162] reviewed the literature on fake news and defined the three categories of fake news, (i) disinformation (false information shared intentionally), (ii) misinformation (false information shared unintentionally), and (iii) expletive (dismiss information that one disagrees with, for closing down a debate). The propagation of different types of false information and specific mitigation techniques for them is still an open question for further studies. It is also required that academic research and policy-making should be concurrent to combat the inverse impact of false information effectively. This is also an open research question, and has not been looked in-depth as how to execute both of these in parallel, given their complexities.

References

1. Parikh, S.B., Atrey, P.K.: Media-rich fake news detection: a survey. In: 2018 IEEE Conference on Multimedia Information Processing and Retrieval (MIPR), pp. 436–441. IEEE (2018)
2. Shu, K., Sliva, A., Wang, S., Tang, J., Liu, H.: Fake news detection on social media: a data mining perspective. SIGKDD Explor. Newsl. **19**(1), 22–36 (2017). September
3. Howard, P.N., Kollanyi, B., Bradshaw, S., Neudert, L.-M.: Social media, news and political information during the us election: was polarizing content concentrated in swing states? (2018). arXiv:1802.03573
4. Kumar, S., West, R., Leskovec, J.: Disinformation on the web: Impact, characteristics, and detection of wikipedia hoaxes. In: Proceedings of the 25th International Conference on World Wide Web, pp. 591–602 (2016)
5. Kumar, S., Shah, N.: False information on web and social media: a survey (2018). arXiv:1804.08559
6. Rashkin, H., Choi, E., Jang, J.Y., Volkova, S., Choi, Y.: Truth of varying shades: analyzing language in fake news and political fact-checking. In: Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing, pp. 2931–2937 (2017)
7. DiFonzo, N., Bordia, P.: Rumor, gossip and urban legends. Diogenes **54**(1), 19–35 (2007)
8. Zubiaga, A., Hoi, G.W.S., Liakata, M., Procter, R., Tolmie, P.: Analysing how people orient to and spread rumours in social media by looking at conversational threads (2015). CoRR arXiv:1511.07487
9. Spohr, D.: Fake news and ideological polarization: filter bubbles and selective exposure on social media. Bus. Inf. Rev. **34**(3), 150–160 (2017)
10. Quattrociocchi, W., Scala, A., Sunstein, C.R.: Echo chambers on facebook. Available at SSRN 2795110 (2016)
11. Bozdag, E., van den Hoven, J.: Breaking the filter bubble: democracy and design. Ethics Inf. Technol. **17**(4), 249–265 (2015)
12. Amoruso, M., Anello, D., Auletta, V., Ferraioli, D.: Contrasting the spread of misinformation in online social networks. In: Proceedings of the 16th Conference on Autonomous Agents and MultiAgent Systems. International Foundation for Autonomous Agents and Multiagent Systems, pp. 1323–1331 (2017)
13. Saxena, A., Hsu, W., Lee, M.L., Chieu, H.L., Ng, L., Teow, L.N.: Mitigating misinformation in online social network with top-k debunkers and evolving user opinions. In: Companion Proceedings of the Web Conference 2020, pp. 363–370 (2020)
14. Shu, K., Mahudeswaran, D., Liu, H.: Fakenewstracker: a tool for fake news collection, detection, and visualization. Comput. Math. Organ. Theory 1–12 (2018)
15. Adair, B.: Principles of politifact and the truth-o-meter **21**, 2011 (2011). PolitiFact. com.
16. Kempe, D., Kleinberg, J., Tardos, É.: Influential nodes in a diffusion model for social networks. In: International Colloquium on Automata, Languages, and Programming, pp. 1127–1138. Springer (2005)
17. Peng, W., Pan, L.: Scalable influence blocking maximization in social networks under competitive independent cascade models. Comput. Netw. **123**, 38–50 (2017)
18. Saxena, A., Iyengar, S.R.S., Gupta, Y.: Understanding spreading patterns on social networks based on network topology. In: Proceedings of the 2015 IEEE/ACM International Conference on Advances in Social Networks Analysis and Mining 2015, pp. 1616–1617 (2015)

19. Saxena, A., Iyengar, S.R.S.: Evolving models for meso-scale structures. In: 2016 8th International Conference on Communication Systems and Networks (COMSNETS), pp. 1–8. IEEE (2016)
20. De Domenico, M., Lima, A., Mougel, P., Musolesi, M.: The anatomy of a scientific rumor. Sci. Rep. **3**, 2980 (2013)
21. Gupta, Y., Saxena, A., Das, D., Iyengar, S.R.S.: Modeling memetics using edge diversity. In: Complex Networks VII, pp. 187–198. Springer (2016)
22. Kempe, D., Kleinberg, J., Tardos, É.: Maximizing the spread of influence through a social network. In: Proceedings of the Ninth ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, pp. 137–146. ACM (2003)
23. Wu, L., Morstatter, F., Hu, X., Liu, H.: Mining misinformation in social media. In: Big Data in Complex and Social Networks, pp. 123–152 (2016)
24. Yang, L., Li, Z., Giua, A.: Containment of rumor spread in complex social networks. Inf. Sci. **506**, 113–130 (2020)
25. Pham, D.V., Duong, H.V., Pham, C.V., Bao, B.Q., Nguyen, A.V.: Multiple topics misinformation blocking in online social networks. In: 2019 11th International Conference on Knowledge and Systems Engineering (KSE), pp. 1–6. IEEE (2019)
26. Anderson, R.M., May, R.M.: Infectious Diseases of Humans: Dynamics and Control. Oxford University Press (1992)
27. Allen, L.J.S.: Some discrete-time si, sir, and sis epidemic models. Math. Biosci. **124**(1), 83–105 (1994)
28. Hethcote, H.W., Van den Driessche, P.: An SIS epidemic model with variable population size and a delay. J. Math. Biol. **34**(2), 177–194 (1995)
29. Zhao, L., Cui, H., Qiu, X., Wang, X., Wang, J.: SIR rumor spreading model in the new media age. Phys. A **392**(4), 995–1003 (2013)
30. Zhao, L., Wang, J., Chen, Y., Wang, Q., Cheng, J., Cui, H.: SIHR rumor spreading model in social networks. Phys. A **391**(7), 2444–2453 (2012)
31. Xiong, F., Liu, Y., Zhang, Z., Zhu, J., Zhang, Y.: An information diffusion model based on retweeting mechanism for online social media. Phys. Lett. A **376**(30–31), 2103–2108 (2012)
32. Nekovee, M., Moreno, Y., Bianconi, G., Marsili, M.: Theory of rumour spreading in complex social networks. Phys. A **374**(1), 457–470 (2007)
33. Maki, D.P., Thompson, M.: Mathematical models and applications: with emphasis on the social life, and management sciences. Technical report (1973)
34. Jin, F., Dougherty, E., Saraf, P., Cao, Y., Ramakrishnan, N.: Epidemiological modeling of news and rumors on twitter. In: Proceedings of the 7th Workshop on Social Network Mining and Analysis, p. 8. ACM (2013)
35. Bettencourt, L.M.A., Cintrón-Arias, A., Kaiser, D.I., Castillo-Chávez, C.: The power of a good idea: Quantitative modeling of the spread of ideas from epidemiological models. Phys. A: Stat. Mech. Appl. **364**, 513–536 (2006)
36. Tambuscio, M., Ruffo, G., Flammini, A., Menczer, F.: Fact-checking effect on viral hoaxes: A model of misinformation spread in social networks. In: Proceedings of the 24th International Conference on World Wide Web, pp. 977–982. ACM (2015)
37. Helbing, D.: Opinion Formation Models, pp. 159–201. Springer, Netherlands (1995)
38. Evans, T., Feng, F.: Opinion formation on dynamic networks: identifying conditions for the emergence of partisan echo chambers. R. Soc. Open Sci. **5**(10), 181122 (2018)
39. Slanina, F., Lavicka, H.: Analytical results for the SZNAJD model of opinion formation. Eur. Phys. J. B-Condens. Matter Complex Syst. **35**(2), 279–288 (2003)
40. Shen, B., Liu, Y.: An opinion formation model with two stages. Int. J. Mod. Phys. C **18**(08), 1231–1242 (2007)
41. Jalili, M.: Social power and opinion formation in complex networks. Phys. A **392**(4), 959–966 (2013)
42. Grabowski, A., Kosiński, R.A.: Ising-based model of opinion formation in a complex network of interpersonal interactions. Phys. A **361**(2), 651–664 (2006)

43. Li, Q., Braunstein, L.A., Wang, H., Shao, J., Stanley, H.E., Havlin, S.: Non-consensus opinion models on complex networks. J. Stat. Phys. **151**(1–2), 92–112 (2013)
44. Tanaka, Y., Sakamoto, Y., Matsuka, T.: Toward a social-technological system that inactivates false rumors through the critical thinking of crowds. In: 2013 46th Hawaii International Conference on System Sciences (HICSS), pp. 649–658. IEEE (2013)
45. Ozturk, P., Li, H., Sakamoto, Y.: Combating rumor spread on social media: The effectiveness of refutation and warning. In: 2015 48th Hawaii International Conference on System Sciences (HICSS), pp. 2406–2414. IEEE (2015)
46. Song, G., Zhou, X., Wang, Y., Xie, K.: Influence maximization on large-scale mobile social network: a divide-and-conquer method. IEEE Trans. Parallel Distrib. Syst. **26**(5), 1379–1392 (2015)
47. Saxena, A., Iyengar, S.: Centrality measures in complex networks: a survey (2020). arXiv:2011.07190
48. Cao, C., Yu, L., Hu, Y.: Containment of rumors underlimitcost budget in social network. In: Fourteenth Wuhan International Conference on E-Business, pp. 341–348 (2015)
49. Zheng, J., Pan, L.: Least cost rumor community blocking optimization in social networks. In: 2018 Third International Conference on Security of Smart Cities, Industrial Control System and Communications (SSIC), pp. 1–5. IEEE (2018)
50. Wu, Q., Zhao, X., Zhou, L., Wang, Y., Yang, Y.: Minimizing the influence of dynamic rumors based on community structure. Int. J. Crowd Sci. (2019)
51. Fan, L., Lu, Z., Wu, W., Thuraisingham, B., Ma, H., Bi, Y.: Least cost rumor blocking in social networks. In: 2013 IEEE 33rd International Conference on Distributed Computing Systems, pp. 540–549. IEEE (2013)
52. Pham, C.V., Phu, Q.V., Hoang, H.X.: Targeted misinformation blocking on online social networks. In: Asian Conference on Intelligent Information and Database Systems, pp. 107–116. Springer (2018)
53. Wang, B., Chen, G., Luoyi, F., Song, L., Wang, X.: DRIMUX: dynamic rumor influence minimization with user experience in social networks. IEEE Trans. Knowl. Data Eng. **29**(10), 2168–2181 (2017)
54. Yao, Q., Shi, R., Zhou, C., Wang, P., Guo, L.: Topic-aware social influence minimization. In: Proceedings of the 24th International Conference on World Wide Web, pp. 139–140 (2015)
55. Pham, D.V., Nguyen, G.L., Nguyen, T.N., Pham, C.V., Nguyen, A.V.: Multi-topic misinformation blocking with budget constraint on online social networks. IEEE Access **8**, 78879–78889 (2020)
56. Yang, L., Li, Z., Giua, A.: Influence minimization in linear threshold networks. Automatica **100**, 10–16 (2019)
57. Wang, X., Deng, K., Li, J., Yu, J.X., Jensen, C.S., Yang, X.: Targeted influence minimization in social networks. In: Pacific-Asia Conference on Knowledge Discovery and Data Mining, pp. 689–700. Springer (2018)
58. Kimura, M., Saito, K., Motoda, H.: Blocking links to minimize contamination spread in a social network. ACM Trans. Knowl. Discov. Data (TKDD) **3**(2), 1–23 (2009)
59. Yan, R., Li, Y., Weili, W., Li, D., Wang, Y.: Rumor blocking through online link deletion on social networks. ACM Trans. Knowl. Discov. Data (TKDD) **13**(2), 1–26 (2019)
60. Tong, H., Prakash, B.A., Eliassi-Rad, T., Faloutsos, M., Faloutsos, C.: Gelling, and melting, large graphs by edge manipulation. In: Proceedings of the 21st ACM International Conference on Information and Knowledge Management, pp. 245–254 (2012)
61. Khalil, E.B., Dilkina, B., Song, L.: Scalable diffusion-aware optimization of network topology. In: Proceedings of the 20th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, pp. 1226–1235 (2014)
62. Medya, S., Silva, A., Singh, A.: Influence minimization under budget and matroid constraints: extended version (2019). arXiv:1901.02156
63. Kuhlman, C.J., Tuli, G., Swarup, S., Marathe, M.V., Ravi, S.S.: Blocking simple and complex contagion by edge removal. In: 2013 IEEE 13th International Conference on Data Mining, pp. 399–408. IEEE (2013)

64. Khalil, E., Dilkina, B., Song, L.: Cuttingedge: influence minimization in networks. In: Proceedings of Workshop on Frontiers of Network Analysis: Methods, Models, and Applications at NIPS (2013)
65. He, J., Liang, H., Yuan, H.: Controlling infection by blocking nodes and links simultaneously. In: International Workshop on Internet and Network Economics, pp. 206–217. Springer (2011)
66. Shi, Q., Wang, C., Ye, D., Chen, J., Feng, Y., Chen, C.: Adaptive influence blocking: minimizing the negative spread by observation-based policies. In: 2019 IEEE 35th International Conference on Data Engineering (ICDE), pp. 1502–1513. IEEE (2019)
67. Zhang, P., Bao, Z., Niu, Y., Zhang, Y., Mo, S., Geng, F., Peng, Z.: Proactive rumor control in online networks. World Wide Web **22**(4), 1799–1818 (2019)
68. Zhang, H., Abdul Alim, Md., Thai, My.T., Nguyen, H.T.: Monitor placement to timely detect misinformation in online social networks. In: 2015 IEEE International Conference on Communications (ICC), pp. 1152–1157. IEEE (2015)
69. Wang, S., Zhao, X., Chen, Y., Li, Z., Zhang, K., Xia, J.: Negative influence minimizing by blocking nodes in social networks. In: Workshops at the Twenty-Seventh AAAI Conference on Artificial Intelligence (2013)
70. Yan, R., Li, D., Wu, W., Du, D.-Z.: Minimizing influence of rumors by blockers on social networks. In: International Conference on Computational Social Networks, pp. 1–12. Springer (2018)
71. Tan, Z., DanKe, W., Gao, T., You, I., Sharma, V.: AIM: activation increment minimization strategy for preventing bad information diffusion in OSNs. Futur. Gener. Comput. Syst. **94**, 293–301 (2019)
72. Scaman, K., Kalogeratos, A., Vayatis, N.: A greedy approach for dynamic control of diffusion processes in networks. In: 2015 IEEE 27th International Conference on Tools with Artificial Intelligence (ICTAI), pp. 652–659. IEEE (2015)
73. Gai, L., Hongwei, D., Lidong, W., Zhu, J., Yuehua, B.: Blocking rumor by cut. J. Comb. Optim. **36**(2), 392–399 (2018)
74. Cui, W., Gong, X., Liu, C., Dan, X., Chen, X., Fang, D., Tang, S., Fan, W., Chen, G.: Node immunization with time-sensitive restrictions. Sensors **16**(12), 2141 (2016)
75. Zhang, Y., Aditya Prakash, B.: Dava: distributing vaccines over networks under prior information. In: Proceedings of the 2014 SIAM International Conference on Data Mining, pp. 46–54. SIAM (2014)
76. Yao, Q., Zhou, C., Xiang, L., Cao, Y., Guo, L.: Minimizing the negative influence by blocking links in social networks. In: International Conference on Trustworthy Computing and Services, pp. 65–73. Springer (2014)
77. Kimura, M., Saito, K., Motoda, H.: Minimizing the spread of contamination by blocking links in a network. AAAI **8**, 1175–1180 (2008)
78. Nandi, A.K., Medal, H.R.: Methods for removing links in a network to minimize the spread of infections. Comput. & Oper. Res. **69**, 10–24 (2016)
79. Wang, X., Deng, K., Li, J., Yu, J.X., Jensen, C.S., Yang, X.: Efficient targeted influence minimization in big social networks. World Wide Web 1–18 (2020)
80. Tong, G., Weili, W., Ding-Zhu, D.: Distributed rumor blocking with multiple positive cascades. IEEE Trans. Comput. Soc. Syst. **5**(2), 468–480 (2018)
81. Zhang, Y., Aditya Prakash, B.: Data-aware vaccine allocation over large networks. ACM Trans. Knowl. Discov. Data (TKDD) **10**(2), 1–32 (2015)
82. Tong, H., Aditya Prakash, B., Tsourakakis, C., Eliassi-Rad, T., Faloutsos, C., Chau, D.H.: On the vulnerability of large graphs. In: 2010 IEEE International Conference on Data Mining, pp. 1091–1096. IEEE (2010)
83. Pham, C.V., Phu, Q.V., Hoang, H.X., Pei, J., Thai, My.T.: Minimum budget for misinformation blocking in online social networks. J. Combin. Optim. **38**(4), 1101–1127 (2019)
84. Garrett, R.K.: Echo chambers online?: Politically motivated selective exposure among internet news users. J. Comput.-Mediat. Commun. **14**(2), 265–285 (2009)
85. Van der Linden, S., Leiserowitz, A., Rosenthal, S., Maibach, E.: Inoculating the public against misinformation about climate change. Glob. Chall. **1**(2), 1600008 (2017)

86. Cook, J., Lewandowsky, S., Ecker, U.K.H.: Neutralizing misinformation through inoculation: exposing misleading argumentation techniques reduces their influence. PloS One **12**(5), e0175799 (2017)
87. Nicol, D.M., Liljenstam, M.: Models and analysis of active worm defense. In: International Workshop on Mathematical Methods, Models, and Architectures for Computer Network Security, pp. 38–53. Springer (2005)
88. Bharathi, S., Kempe, D., Salek, M.: Competitive influence maximization in social networks. In: International Workshop on Web and Internet Economics, pp. 306–311. Springer (2007)
89. Dubey, P., Garg, R., De Meyer, B.: Competing for customers in a social network: the quasi-linear case. In: International Workshop on Internet and Network Economics, pp. 162–173. Springer (2006)
90. Kostka, J., Oswald, Y.A., Wattenhofer, R.: Word of mouth: rumor dissemination in social networks. In: International Colloquium on Structural Information and Communication Complexity, pp. 185–196. Springer (2008)
91. Budak, C., Agrawal, D., El Abbadi, A.: Limiting the spread of misinformation in social networks. In: Proceedings of the 20th international conference on World wide web, pp. 665–674. ACM (2011)
92. Tong, A., Du, D.-Z., Wu, W.: On misinformation containment in online social networks. In: Advances in Neural Information Processing Systems, pp. 339–349 (2018)
93. Nguyen, N.P., Yan, G., Thai, My.T., Eidenbenz, S.: Containment of misinformation spread in online social networks. In: Proceedings of the 4th Annual ACM Web Science Conference, pp. 213–222. ACM (2012)
94. McPherson, M., Smith-Lovin, L., Cook, J.M.: Birds of a feather: homophily in social networks. Annu. Rev. Sociol. **27**(1), 415–444 (2001)
95. Song, C., Hsu, W., Lee, M.-L.: Temporal influence blocking: minimizing the effect of misinformation in social networks. In: 33rd IEEE International Conference on Data Engineering, ICDE 2017, San Diego, CA, USA, 19–22 April 2017, pp. 847–858 (2017)
96. Saxena, A., Saxena, H., Gera, R.: k-truthscore: fake news mitigation in the presence of strong user bias. In: International Conference on Computational Data and Social Networks, pp. 113–126. Springer (2020)
97. Xu, W., Zhang, F., Zhu, S.: Toward worm detection in online social networks. In: Proceedings of the 26th Annual Computer Security Applications Conference, pp. 11–20. ACM (2010)
98. Wilder, B., Vorobeychik, Y.: Defending elections against malicious spread of misinformation (2018). arXiv:1809.05521
99. Tanınmış, K., Aras, N., Altınel, İK., Güney, E.: Minimizing the misinformation spread in social networks. IISE Trans. **52**(8), 850–863 (2020)
100. Farajtabar, M., Yang, J., Ye, X., Xu, H., Trivedi, R., Khalil, E., Li, S., Song, L., Zha, H.: Fake news mitigation via point process based intervention. In: Proceedings of the 34th International Conference on Machine Learning, vol. 70, pp. 1097–1106 (2017). JMLR. org
101. Yan, G., Chen, G., Eidenbenz, S., Li, N.: Malware propagation in online social networks: nature, dynamics, and defense implications. In: Proceedings of the 6th ACM Symposium on Information, Computer and Communications Security, pp. 196–206. ACM (2011)
102. He, X., Song, G., Chen, W., Jiang, Q.: Influence blocking maximization in social networks under the competitive linear threshold model. In: Proceedings of the 2012 SIAM International Conference on Data Mining, pp. 463–474. SIAM (2012)
103. Yang, L., Ying, S., Jixi, S., Bo, J., Jianjun, W.: Preventing the diffusion of negative information based on local influence tree. In: Proceedings of the 29th Annual ACM Symposium on Applied Computing, pp. 650–652. ACM (2014)
104. Chen, W., Wang, Y., Yang, S.: Efficient influence maximization in social networks. In: Proceedings of the 15th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, pp. 199–208. ACM (2009)
105. He, J.S., Xie, Y., Du, T., Ji, S., Li, Z.: Influence spread in social networks with both positive and negative influences. In: International Computing and Combinatorics Conference, pp. 615–629. Springer (2017)

106. Litou, I., Kalogeraki, V., Katakis, I., Gunopulos, D.: Efficient and timely misinformation blocking under varying cost constraints. Online Soc. Netw. Media **2**, 19–31 (2017)
107. Wenlong Zhu, W., Yang, S.X., Man, D., Wang, W., Xiaojiang, D.: Location-aware influence blocking maximization in social networks. IEEE Access **6**, 61462–61477 (2018)
108. Lv, J., Yang, B., Yang, Z., Zhang, W.: A community-based algorithm for influence blocking maximization in social networks. Cluster Comput. 1–16 (2017)
109. Simpson, M., Srinivasan, V., Thomo, A.: Scalable misinformation prevention in social networks (2018). arXiv:1807.01162
110. Fan, L., Weili, W., Zhai, X., Xing, K., Lee, W., Ding-Zhu, D.: Maximizing rumor containment in social networks with constrained time. Soc. Netw. Anal. Min. **4**(1), 214 (2014)
111. Vu, M.M., Hoang, H.X.: Minimizing the spread of misinformation on online social networks with time and budget constraint. In: 2017 9th International Conference on Knowledge and Systems Engineering (KSE), pp. 160–165. IEEE (2017)
112. Zhang, H., Zhang, H., Li, X., Thai, My.T.: Limiting the spread of misinformation while effectively raising awareness in social networks. In: International Conference on Computational Social Networks, pp. 35–47. Springer (2015)
113. Zhang, H., Kuhnle, A.,David Smith, J., Thai, My.T.: Fight under uncertainty: restraining misinformation and pushing out the truth. In: 2018 IEEE/ACM International Conference on Advances in Social Networks Analysis and Mining (ASONAM), pp. 266–273. IEEE (2018)
114. Hosni, A.I.E., Li, K., Ding, C., Ahmed, S.: Least cost rumor influence minimization in multiplex social networks. In: International Conference on Neural Information Processing, pp. 93–105. Springer (2018)
115. Tong, G., Wu, W., Guo, L., Li, D., Liu, C., Liu, B., Du, D.-Z.: An efficient randomized algorithm for rumor blocking in online social networks. IEEE Trans. Netw. Sci. Eng. (2017)
116. Tsai, J., Nguyen, T.H., Tambe, M.: Security games for controlling contagion. In: Twenty-Sixth AAAI Conference on Artificial Intelligence (2012)
117. Wu, Y., Huang, H., Zhao, J., Wang, C., Wang, T.: Using mobile nodes to control rumors in big data based on a new rumor propagation model in vehicular social networks. IEEE Access **6**, 62612–62621 (2018)
118. Fan, L., Weili, W., Xing, K., Lee, W.: Precautionary rumor containment via trustworthy people in social networks. Discret. Math. Algorithms Appl. **8**(01), 1650004 (2016)
119. Lin, K.-S., Dai, B.-R.: Biog: an effective and efficient algorithm for influence blocking maximization in social networks. In: International Conference on Data Mining and Big Data, pp. 328–337. Springer (2019)
120. Park, S., Kang, S., Chung, S., Song, J. Newscube: delivering multiple aspects of news to mitigate media bias. In: Proceedings of the SIGCHI Conference on H:uman Factors in Computing Systems, pp. 443–452. ACM (2009)
121. Ennals, R., Trushkowsky, B., Agosta, J.M.: Highlighting disputed claims on the web. In: Proceedings of the 19th International Conference on World Wide Web, pp. 341–350. ACM (2010)
122. Hassan, N., Sultana, A., You, W., Zhang, G., Li, C., Yang, J., Cong, Yu.: Data in, fact out: automated monitoring of facts by factwatcher. Proc. VLDB Endow. **7**(13), 1557–1560 (2014)
123. Hassan, N., Arslan, F., Li, C., Tremayne, M.: Toward automated fact-checking: detecting check-worthy factual claims by claimbuster. In: Proceedings of the 23rd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, pp. 1803–1812. ACM (2017)
124. Ratkiewicz, J., Conover, M., Meiss, M., Gonçalves, B., Patil, S., Flammini, A., Menczer, F.: Truthy: mapping the spread of astroturf in microblog streams. In: Proceedings of the 20th International Conference Companion on World Wide Web, pp. 249–252. ACM (2011)
125. Figueira, Á., Oliveira, L.: The current state of fake news: challenges and opportunities. Proc. Comput. Sci. **121**, 817–825 (2017)
126. Guha, S.: Related fact checks: a tool for combating fake news (2017). arXiv:1711.00715
127. de Alfaro, L., Di Pierro, M., Tacchini, E., Ballarin, G., Vedova, M.L.D., Moret, S.: Reputation systems for news on twitter: a large-scale study (2018). arXiv:1802.08066

128. Pavleska, T., Školkay, A., Zankova, B., Ribeiro, N., Bechmann, A.: Performance analysis of fact-checking organizations and initiatives in Europe: a critical overview of online platforms fighting fake news. Social Media Converg. 17 (2018)
129. Jackson, B.: Firefighters, fact-checking and American journalism (2012). FactCheck. org
130. LaCapria, K.: Snopes' field guide to fake news sites and hoax purveyors (2017). Snopes. com
131. Kim, J., Tabibian, B., Oh, A., Schölkopf, B., Gomez-Rodriguez, M.: Leveraging the crowd to detect and reduce the spread of fake news and misinformation. In: Proceedings of the Eleventh ACM International Conference on Web Search and Data Mining, pp. 324–332. ACM (2018)
132. Ingram, M.: Facebook's fact checking can make fake news spread even faster (2017). http://fortune.com/2017/05/16/facebook-fact-checking/
133. Vo, N., Lee, K.: The rise of guardians: fact-checking url recommendation to combat fake news. In: The 41st International ACM SIGIR Conference on Research & Development in Information Retrieval, pp. 275–284. ACM (2018)
134. Nguyen, A.T., Kharosekar, A., Krishnan, S., Krishnan, S., Tate, E., Wallace, B.C., Lease, M.: Believe it or not: designing a human-AI partnership for mixed-initiative fact-checking. In: The 31st Annual ACM Symposium on User Interface Software and Technology, pp. 189–199. ACM (2018)
135. Lease, M.: Fact checking and information retrieval (2018)
136. Roozenbeek, J., van der Linden, S.: The fake news game: actively inoculating against the risk of misinformation. J. Risk Res. 1–11 (2018)
137. Jang, S.M., Kim, J.K.: Third person effects of fake news: fake news regulation and media literacy interventions. Comput. Hum. Behav. **80**, 295–302 (2018)
138. Pennycook, G., Rand, D.G.: Who falls for fake news? The roles of bullshit receptivity, over-claiming, familiarity, and analytic thinking. SSRN Electron. J. (2018)
139. Kanoh, H.: Why do people believe in fake news over the internet? An understanding from the perspective of existence of the habit of eating and drinking. Proc. Comput. Sci. **126**, 1704–1709 (2018)
140. Ghenai, A., Mejova, Y.: Fake cures: user-centric modeling of health misinformation in social media. Proc. ACM Hum.-Comput. Interact. **2**(CSCW), 58 (2018)
141. Balmau, O., Guerraoui, R., Kermarrec, A.-M., Maurer, A., Pavlovic, M., Zwaenepoel, E.: Limiting the spread of fake news on social media platforms by evaluating users' trustworthiness (2018). arXiv:1808.09922
142. Nguyen, H., Huyi, C., Warren, P.: Mitigating the spread of fake news by identifying and disrupting echo chambers
143. Colleoni, E., Rozza, A., Arvidsson, A.: Echo chamber or public sphere? predicting political orientation and measuring political homophily in twitter using big data. J. Commun. **64**(2), 317–332 (2014)
144. Manning, C., Raghavan, P., Schütze, H.: Introduction to information retrieval. Nat. Lang. Eng. **16**(1), 100–103 (2010)
145. Clayton, K., Blair, S., Busam, J.A., Forstner, S., Glance, J., Green, G., Kawata, A., Kovvuri, A., Martin, J., Morgan, E., et al.: Real solutions for fake news? Measuring the effectiveness of general warnings and fact-check tags in reducing belief in false stories on social media. Polit. Behav. 1–23 (2019)
146. Graves, L.: Understanding the promise and limits of automated fact-checking. Factsheet **2**, 2018–02 (2018)
147. Viswanath, B., Mislove, A., Cha, M., Gummadi, K.P.: On the evolution of user interaction in facebook. In: Proceedings of the 2nd ACM Workshop on Online Social Networks, pp. 37–42 (2009)
148. Nguyen, N.P., Yan, G., Thai, My.T.: Analysis of misinformation containment in online social networks. Comput. Netw. **57**(10), 2133–2146 (2013)
149. Li, G., Chen, S., Feng, J., Tan, K.-L., Li, W.-S.: Efficient location-aware influence maximization. In: Proceedings of the 2014 ACM SIGMOD International Conference on Management of Data, pp. 87–98 (2014)

150. Leskovec, J., Mcauley, J.J.: Learning to discover social circles in ego networks. In: Advances in Neural Information Processing Systems, pp. 539–547 (2012)
151. Yan, R., Li, D., Wu, W., Du, D.-Z., Wang, Y.: Minimizing influence of rumors by blockers on social networks: algorithms and analysis. IEEE Trans. Netw. Sci. Eng. (2019)
152. Cho, E., Myers, S.A., Leskovec, J.: Friendship and mobility: user movement in location-based social networks. In: Proceedings of the 17th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, pp. 1082–1090 (2011)
153. Leskovec, J., Huttenlocher, D., Kleinberg, J.: Predicting positive and negative links in online social networks. In: Proceedings of the 19th International Conference on World Wide Web, pp. 641–650 (2010)
154. Chen, W., Collins, A., Cummings, R., Ke, T., Liu, Z., Rincon, D., Sun, X., Wang, Y., Wei, W., Yuan, Y.: Influence maximization in social networks when negative opinions may emerge and propagate. In: Proceedings of the 2011 SIAM International Conference on Data Mining, pp. 379–390. SIAM (2011)
155. Leskovec, J., Kleinberg, J., Faloutsos, C.: Graph evolution: densification and shrinking diameters. ACM Trans. Knowl. Discov. Data (TKDD) **1**(1), 2–es (2007)
156. Ripeanu, M., Foster, I.: Mapping the Gnutella network: macroscopic properties of large-scale peer-to-peer systems. In: International Workshop on Peer-to-Peer Systems, pp. 85–93. Springer (2002)
157. Richardson, M., Agrawal, R., Domingos, P.: Trust management for the semantic web. In: International Semantic Web Conference, pp. 351–368. Springer (2003)
158. Klimt, B., Yang, Y.: Introducing the enron corpus. In: CEAS (2004)
159. Newman, M.E.J.: Finding community structure in networks using the eigenvectors of matrices. Phys. Rev. E **74**(3), 036104 (2006)
160. Nguyen, D.T., Zhang, H., Das, S., Thai, My.T., Dinh, T.N.: Least cost influence in multiplex social networks: model representation and analysis. In: 2013 IEEE 13th International Conference on Data Mining, pp. 567–576. IEEE (2013)
161. Leskovec, J., Kleinberg, J., Faloutsos, C.: Graphs over time: densification laws, shrinking diameters and possible explanations. In: Proceedings of the Eleventh ACM SIGKDD International Conference on Knowledge Discovery in Data Mining, pp. 177–187 (2005)
162. Haiden, L., Althuis, J.: The definitional challenges of fake news (2018)

Chapter 17
Data Privacy and Security in Social Networks

Geocey Shejy

Abstract Social networking creates relationships through the internet and it gains indivisible relation with human life nowadays. Social networking sites and applications handle a large volume of data. As more personal information flows in and out of social networks, data privacy and security in the social network become a topic of discussion and arguments. This chapter emphasizes data privacy and also differentiates privacy from security. Initial sections of the chapter explain privacy in its elementary form as a need and right of a human being under the perspective of anthropology and behavioral science. Personal data privacy, its current scenario, threats and its protection by law and policymaking by various governments around the world are discussed further. The chapter considers social networking beyond networking sites and applications, and hence a discussion on privacy threats for sensitive data which spread across fields such as health data, forensic, smart toys, image and video surveillance is also analyzed. Positives and negatives of social network's underlying technologies, like machine learning, artificial intelligence, data sciences, the internet of things and blockchain are discussed in terms of data privacy. The personal data privacy measures imposed by law that need to be incorporated as the part of privacy policies of organizations or that need to be implemented with the support of data security mechanisms are discussed in the last part of the chapter.

17.1 Introduction

Social network helps to build relations using the internet. Social networking sites and applications work beyond leisure and are used for building business relationships also. Improved personal relations and client relations are highlighted as the positives of social networking. But, considering it as a system that deals with loads of personal data, it raises caution on data privacy and security. This chapter considers personal data privacy as a basic human need and desire. It discusses global efforts to implement it in the form of law. Social networking technologies and policies must abide by the

G. Shejy (✉)
VES Institute of Technology, Chembur, Mumbai, Maharashtra, India

A. Biswas et al. (eds.), *Principles of Social Networking*,
Smart Innovation, Systems and Technologies 246,
https://doi.org/10.1007/978-981-16-3398-0_17

privacy laws of each country and territory with respect to stakeholders, the collection of data, processing of data and transferring of it. As social networking penetrated to all walks of life, the understanding of related laws is significant even for a layperson. Considering social networking beyond personal relationships, this chapter extends to other fields which handle personal sensitive data like medical records, educational records, toys etc. The underlying technologies of social networking use big data analytics (BDA), machine learning (ML) and artificial intelligence (AI). The vulnerabilities of these technologies are inherent in social networking and are considered in the chapter. Privacy by design, applying various security safeguards, data protection impact assessment, data audits, grievance redressal and many more precautionary measures and tools are required to avoid data privacy threats. Advanced technologies like those mentioned above must support not only to build a network of relationships but also to build "personal data safe relations".

17.2 Privacy

17.2.1 Privacy as a Human Need

The notion of privacy is quite human. Recent technical advancements raised the term privacy in various platforms of discussion, but in its essence, it remains a human need. Privacy attained its significance in communication technology in the last few decades; it is skyrocketed after the popularity of social networks. But traditionally it is a concept spread across anthropology and behavioral sciences. Our initial discussion is focusing privacy in its original form or how privacy is defined in human life. Defining technical data privacy under the shadow of anthropologically defined privacy helps to draw closer solutions for the human desire for privacy [1].

Hanna Arendt in her book "The Human Condition" illustrates humans as conditioned animals, as everything coming into their contact will immediately convert into a condition for their existence [2]. Humans assimilate life-given conditions and self-made conditions. When our lives are in an era of social networking, we are conditioned to create or change our perspective about privacy and hence defining privacy individually.

Every normal human life always keeps two sides as private and public realm, neither of which inevitable [3]. To get the feeling of excellence in life, man needs a public performance or else there must be the presence of others who see what we see and to hear what we hear; again there comes a choice. But the most intense feeling of a person that is irrelevant for others will be most private and least communicable. Man finds more meaning in small things around him like love, smile, kids, relationships, pain, illness and many more which is kept in a restricted sphere. Man shows more intimacy to these private realms where he never expects the presence of the other and is the inner subjectivity of the individual. Hence we find that each individual has his/her own definition for privacy. Because of this privacy, it remains notoriously hard to capture and conceptualize. The meaning and function of privacy are not fixed by anthropologists but their introspection is highly influenced by the culture or community they belonged to [4].

Stanford Encyclopedia of Philosophy defines privacy from the perspective of various theorists. It is viewed as control over information about oneself. It is defined as a concept related to human dignity and crucial for intimacy. Privacy is necessary for the development of varied and meaningful interpersonal relationships. It is the value that accords us the ability to control the access others have to. It is a set of norms necessary not only to control access but also to enhance personal expression and choice, or some combination of these [5].

17.2.2 Privacy as Human Right

When a person inherits something in the normal course of life there is no question of rights. For example, when freedom is adhered to a person by birth he never thinks that it's his right, but at the same time if he faces a denial of it, then "freedom is a right" thought process starts and will respond according to the severity of the denial. To protect rights laws are framed. But protection in person or property by common law is impossible because of the time-to-time change of the definition of the protection [6]. Political, social and economic changes force new definitions on the "right to life", and thereby a new common law. From the right of a property, it broadened to spiritual nature, intellect and feelings. A lot of intangible properties such as work of literature or art, trademark, trade secrets, goodwill and emotional life all are rights. Many of these properties are personal and hence coming under the realm of privacy.

Thomas M. Cooley, one of the famous jurists of the nineteenth century, in his book "The Law of Torts" mentioning that every distinct invasion of right presumes some damages. Invading the privacy of one's home, by listening to the keyholes, playing spy at windows also makes damages [7]. Privacy and its violation is a complex abstract idea. Even one of the famous definitions of privacy "to be let alone", becomes questionable when police saying they are not physically touching a person, left him alone, but monitoring his words and deeds inside his house with a device attached to his body, according to Judith Jarvis Thomson [8]. Her Article "Right to Privacy" narrates the difficulty of deciding invasion into privacy with several imaginary case studies. Right to privacy violation is hard to decide in many cases.

In essence, the meaning and perspective of personal privacy are dynamic and so the privacy law. Post and pre-industrial revolution, village to urban living, media in its initial development stage like newspaper, Citizen's socio-economic status recording by governments all lead to battle on privacy in various forms [9]. The introduction of computers and information technology, wireless communication, smart electronic gadgets, social networking all intensified the privacy battle. More technology happened to gather more data with and without the consent of individuals and hence privacy battle is on the center stage now. Traditional personal privacy gives way to data privacy and personal sensitive data and is discussed in the following sections.

17.3 Data Privacy Laws

The above discussion considered personal privacy in its elementary form and as a requirement of human life. But communication technology developments redefined privacy. Personal data privacy, data theft, personal sensitive data, anonymization, data privacy law etc. are some of the terminologies roaming in the new field of personal privacy, in comparison to traditional human privacy.

Understanding the gravity of the changed scenario related to personal data and its privacy, many countries came out with their personal data protection bill. European Union implemented the GDPR (General Data Protection Regulation (EU) 2016/679) as the new data protection regulation, put into effect on May 25, 2018 [10, 11]. GDPR is a set of rules to protect individuals regarding their personal data as property and its protection as a fundamental right but equally emphasizes that this right is not an absolute one, and it must maintain a balance with other fundamental rights for the functioning of society. It addresses the transfer and processing of data inside and outside the EU and EEA (European Economic Areas). The United Kingdom updated GDPR to Data Protection Act after leaving EU [12]. UK's Data Protection Act extended laws to areas not covered by GDPR, with some amendments. However, Australia has a mix of federal, state and territory laws for data privacy and protection and most states and territory have their own data protection legislation [13]. There is Information Privacy Act 2014 for Australian Capital Territory, Information Act 2002 for Northern Territory, Privacy and Personal Information Protection Act 1998 for New South Wales, Information Privacy Act 2009 for Queensland, Personal Information Protection Act 2004 for Tasmania and Privacy and Data Protection Act 2014 for Victoria territories. Brazil enacted its Data Protection Law, known as Lei Geral de Proteção de Dados or LGPD. The LGPD is similar to the EU General Data Protection Regulation (GDPR) [14]. Personal Information Protection and Electronic Documents Act is the Canadian Law toward data protection [15].

In a broad inspection of data privacy laws of various countries, the following areas are found common in all: (1) the stakeholders (2) collection and processing of data and (3) transfer of data.

17.3.1 Important Definitions Related to Data Privacy

As the detailed study of privacy laws is not in the scope of this chapter, only important terminologies and definitions are mentioned here.

Personal data means data about or relating to a natural person who is directly or indirectly identifiable, having regard to any characteristic, trait, attribute or any other feature of the identity of such natural person, or any combination of such features with any other information [17].

Sensitive personal data means personal data revealing, related to, or constituting, as may be applicable (i) passwords, (ii) financial data, (iii) health data, (iv) official identifier, (v) sex life, (vi) sexual orientation, (vii) biometric data, (viii) genetic data, (ix) transgender status, (x) intersex status, (xi) caste or tribe [17].

Data controllers and data processors are two entities taking part in personal data collection and processing. *Data controllers* are part of organizations, affiliated with a central body, with a legitimate interest in transmitting personal data within the organization, and this data is used for an internal administrative purpose [16]. *Data processors* mean any person, including the State, a company, any juristic entity or any individual who processes personal data on behalf of a data fiduciary, but does not include an employee of the data fiduciary [17]. A data fiduciary is a person or part of an organization responsible for processing personal data.

Data subject is the identified or identifiable living individual to whom personal data relates [18].

Anonymity can be defined as being without a name or with an unknown name. It is a feeling of lost in the crowd [19]. Anonymity in philosophy and politics helped to create and publish revolutionary thoughts without revealing identity. It provides tight personal space with a feeling of power. But at the same time, it may trigger aggressive, dishonest and inhuman behavior. Anonymity in cyberspace also has its positives and negatives. A cyber attacker uses anonymity to hide identity but for an innocent positive user of the internet, and it enhances his freedom to move around the cyberspace without the worry of prejudice and biases of others. It also protects the user from any negative repercussions flowing from the transaction and shielding them from accountability for unlawful acts and from any unwarranted reprisals [20].

Data anonymization is the use of one or more techniques designed to make it impossible to identify a particular individual from stored data related to them [21].

Pseudonymity is the use of a false name [22]. Pseudonymity gives a name to an otherwise nameless, faceless user [20]. Anonymity is a total hide but pseudonymity gives another name because a name always gives a reputation. This still permits hiding the self-identity with an escape from the prejudice of others, and can also enjoy the benefits of goodwill with limited liability. *Data pseudonymization* is the process that allows switching the original data set with an alias or pseudonym. Pseudonymization is a reversible process—meaning it makes it possible to de-identify the data but allows the reidentification of the data later on if necessary [23].

Traceability is the ability of someone, most often the government, to trace back to the source [20]. This is applicable for both anonymous and pseudonymous activities. Untraceable pseudonymity eliminates data collection threats such as self-censorship of expression, hindrance of free thought, misuse of personal information, interference with autonomy and interference with identity. Even if an attack happening to pseudonymity in cyberspace it may not extend to the person's real life.

Data generalization allows replacing a data value with a less precise one, via binning, reformatting, rounding or truncating, which preserves data utility and protects against linkage attacks [24]. *Linkage attack* is the use of quasi-identifiers (pieces of information, alone cannot reveal the identity) when joined or linking together leads to identification.

Territorial considerations in privacy law: Each privacy law defines the geographical boundary for transmission and use of collected personal information within the territory of the country or state, a third country or specified sector within a third country or an international organization [16]. Law also mentions restrictions and conditions for cross-border transfer of data by data controller [17].

Differential privacy describes a promise made by a data holder, or curator, to a data subject: “You will not be affected, adversely or otherwise, by allowing your data to be used in any study or analysis; no matter what other studies, data sets, or information sources are available” [25].

17.4 Data Privacy Versus Data Security

Data security and data privacy are two domains that needed to address separately. In major cases, privacy is considered similar to security. But data security and privacy are more or less overlapping realms that support each other. Need to view and deal with these as two separate entities, there exist different ways to achieve both. For example, data privacy starts at the point of collection of data (e.g., consent of data subject), whereas data security addresses the way the data are transferred (encryption) and stored (access control). Data privacy is expecting no unwanted personal information leakage, while data is getting processed or it ensures control of data subject over the use of his data by a third party. Security assures only authorized processes or persons to handle the data. There are situations that demand a balancing between security and privacy as in the case of camera surveillance. Normally this helps in security, but the information it collects and processes may lead to privacy violations and cause a conflict between these two entities. Data security can be considered as a set of mechanisms or algorithms that ensures safe and secure collection (password, OTP, multi-factor authentication), transfer (encryption, cryptography, network security algorithms), storage and processing (authentication, access control, integrity check) of data. Data privacy is a set of policies the organizations create to ensure the trustful collection and use of data. It is an ever-existing promise to the data subject that none of his personal information will be revealed even accidentally at any stage of data handling.

Security and privacy are interrelated but they are performing distinct functions to perform protection of information. The idea of security and its implementation across governments and non-government organizations started well before the thought of privacy started. Privacy focus only on personally identifiable information; it deals with the collection and process of personal data through proper legal channels [26].

Implementing data privacy as enforced by law is not an easy task for organizations that uses customer data as their major working fuel. Even though data security mechanisms offer data privacy to some extent, they are unable to give complete privacy as expected. This section is to examine the requirements of data privacy and the amount to which got satisfied by existing data security mechanisms.

The total process of personal data handling is divided into four important subprocesses to make the understanding easy.

- Data creation and collection
- Data processing
- Data storage
- Data transfer

The following section explains the key data privacy requirements in the above process, keeping various personal data privacy laws as reference [16–18]. Organizations need to make policies or find ways to impose privacy legally.

17.4.1 Personal Data Privacy in Data Generation and Data Collection

In a normal situation, data are collected directly from the data subject. But there are situations where data are collected by a third party from an already existing data controller. The data controller and data subject must engage in mutually understood data exchange and trust. For this purpose, organizations (government or non-government) must have visible personal data privacy policies, in compliance with the existing data protection laws of that country or territory.

The following area must be covered by an organization policy to ensure no privacy violation at the data creation or collection point.

Consent of data subject: Taking consent of data subject must be a clear affirmative act. It is expected that the data subject must give an agreement in the form of a written statement by electronic means or an oral statement. This agreement must be very specific and informed to the data subject leaving no way for misunderstandings.

Consent of data subject while collecting data directly from him: In a normal scenario the data controller will collect personal data directly from the data subject with his consent.

Consent of data subject while transferring data to some other parties, not involved in the existing contract: Some situations will arise such that a legally authorized agency (like government bodies) may ask data from an already existing data controller, then the controller is responsible to inform the data subject about the data transfer if it involves any personal data.

Notification to data subject about the data collection: This comes as a part of the consent of the data subject after proper notification from the data controller.

Collection limitation: The collection of data must be with a clear boundary defined with respect to the purpose of collection, categories of personal data being collected; if not collected directly from the data subject must intimate the other source of collection.

Data subject withdraws his consent in already agreed contact: If data subject withdraws consent that he already agreed upon, then the data subject must bear the legal consequences of the withdrawal.

17.4.2 Personal Data Privacy in Data Processing

To ensure complete and accurate data processing, it must be conducted with the support of the law. Lawful processing makes sure that the data undergoing is not misleading and the data provided is an updated one. Various data privacy acts expect proper processing of data in the following ways.

Intimation: Processing of personal data or sensitive personal data must be properly informed to the data subject. The data subject must be aware of the purpose of data collection, the operations happening on the collected data and its consequences.

Clear and Specific: Processing must be clear and specific. The consent of the data subject is required with a clear understanding of the purposes, operations of sensitive personal data relevant to processing.

Right to correction: The provided data can be corrected, if it is not accurate, misleading, incomplete or out of date.

Special situations: Special situations of processing of personal data include medical emergency, law and order of a court or tribunal, the necessary purpose of employment and sensitive data related to children.

Right to be forgotten: The data subject who has already given consent to a data controller can restrict or prevent personal data disclosure. This right can be used if the data provided has served the purpose and longer necessary.

Notify data breach: The controller has the responsibility to communicate to the data subject if any personal data breach has happened. The personal data breach is likely to result in a high risk to the rights and freedoms of the natural person in order to allow him to take the necessary precautions.

17.4.3 Personal Data Privacy in Data Storage

Storage limitation: The lifetime of personal data in data storage is the time necessary to perform the purpose of data collection. Personal data may be retained for a longer period of time only if that is explicitly mandated.

Periodic review: Periodic reviews need to be conducted to decide the retention of personal data in the data controller's storage.

17.4.4 Personal Data Privacy in Transfer of Data

Right to data portability: Personal data received related to the data subject is expected in a structured, commonly used and machine-readable format.

Cross-border transfer of data: The restrictions and conditions for cross-border data transfer are based on the law. Data Privacy Acts are formed in an expectation of establishing international cooperation for the protection of data.

Data security at various levels of handling of personal data

Password: To make sure the authorized person is entering data.

One-time password: To ensure authorized device and person working on data.

Accept terms and conditions before using the services. While doing so, the data subject is agreeing with all privacy and security policies of the organization.

Many algorithms and encryption techniques are used for the security of data processing, storage and transfer. Protection mechanisms can be used to keep the integrity of personal data, prevent misuse, unauthorized access to, modification, disclosure or destruction of personal data. A periodic review of security mechanisms and appropriate measures will support privacy.

17.5 Data Privacy Threats in Various Fields

Social networking sites, social networking applications and social networking services—all are taking, processing and delivering a lot of personal data. This section discusses the data privacy issues that arise in some areas when data handled across various fields such as government records, education, business, industry, health sector, video and image surveillance etc. The data controller and processor must assure the differential privacy of the data subject. A huge amount of data is the fuel of many social networking technologies, like artificial intelligence, machine learning, deep learning and big data analytics. While using these technologies organizations are expected to keep the promise of differential privacy.

This section examines the privacy threat across various fields, where a lot of data are flowing in and out of the system. Privacy threats for sensitive data in the areas of health data, forensic, smart toys, image and video surveillance, industry and business etc. are analyzed here.

17.5.1 Privacy Issues in Health Sector

The health sector is one of the key areas getting the benefit of technology advancement right from disease diagnosis to treatment. It generates a lot of clinical data which is getting stored, edited and exchanged or communicated. Processing and exchanging of these data may lead to personal data disclosure and thereby unauthorized, illegal access. But at the same time, health data sharing is necessary to improve the quality of healthcare. Hence, a hospital management system (HIS) must design to comply with personal data protection laws and standards and must ensure patient privacy [27].

Technology-driven progress in the healthcare industry is indicated as Healthcare 1.0, Healthcare 2.0, Healthcare 3.0 and Healthcare 4.0. Healthcare 1.0 was including technologies that support medical practitioners, while Healthcare 2.0 came up with electronic healthcare records (EHRs). From Healthcare 3.0 onwards it became more patient-centric. Various disruptive technologies like cloud computing (CC), fog computing (FC), Internet of Things (IoT) are effectively applied by Healthcare 4.0. EHR, for its meaningful use, needs to register a person's information from birth and continuing to collect and update the health data of the person and it is obvious that EHR is a large source of personal information [28]. Technologies empowered the healthcare sector, and social networking is used to build closer patient relationships. But, if this data not properly managed, it will erode patient confidentiality and trust in the healthcare system [29].

The indispensable involvement of technology in healthcare and health information creates a new challenge for healthcare providers. Other than the basic intention of providing healthcare, medical professionals must be aware of cybersecurity and possible threats. The technology adopted in healthcare must have proper planning and implementation. This also must ensure consistent updating to keep up with the advancement in cybercriminal technologies [30].

The survey "In An exhaustive survey on security and privacy issues in Healthcare 4.0" by Jigna J. Hathaliya et al. is listing out the major attacks in the healthcare sector that happened from 1989 to 2019 [28].

Possible attacks on privacy in the health sector

Ransomware: Ransomware is one cyber threat found in the healthcare sector. It is characterized by a hijack in the healthcare network and seizing the authorized use of EHR, then ransoming the file for a fee. Ransomware exposes and exploits the vulnerabilities of disruptive technology infrastructure [31]. Locky ransomware is one example of this kind that encrypts medical records on a computer. It keeps the records in a hostage state and thereby demanding a ransom payment. Hospitals had to pay a large amount in bitcoin to locky attackers in 2017. In WinPlock, another ransomware attack in 2016, positively some hospitals in Germany and Canada were able to survive this threat. But all attacks were not easy to resolve. Crypto wall ransomware encrypted electronic medical records and backup copies, and the hospital had to pay an undisclosed ransom fee to get the data back.

Third-Party Vendor: Outsourcing the management of EHRs or electronic medical records (EMRs) to third-party service providers is now increasing. This can serve as a vulnerable point for attacker's entry into healthcare data. The trust of the third-party vendor is quite significant in this case. If the security breach happening for a third-party company with a large market share, it ultimately compromises multiple millions of patient records of a number of various hospitals at once [31]. A third-party vendor must be equipped and must offer multiple levels of security to its clients. It must have a secured network with firewalls and mechanisms for the encryption of data. Password protection to ensure appropriate or authorized access of information is a minimum expectation. Highly secure physical storage locations with restricted entry, reliable backup to avoid the risk of data loss from a natural disaster or irreparable system failures and use of tools to track persons accessing and editing EHR are some data security promises that a third-party vendor can offer to its clients. Encrypting of information is crucial when using EHRs with mobile equipment [32].

Insider Threat: Insider threat is happening in the following levels [33]:

(1) Insiders who unknowingly make mistakes while handling the records may lead to accidental disclosure of personal information.
(2) Insiders who are authorized but misuse their record access privileges.
(3) Insiders who access information with an aim of profit.
(4) The unauthorized physical intruder.
(5) Employees and outsiders with some vested interest in revenge may attack to access unauthorized information, damage systems and disrupt operations.

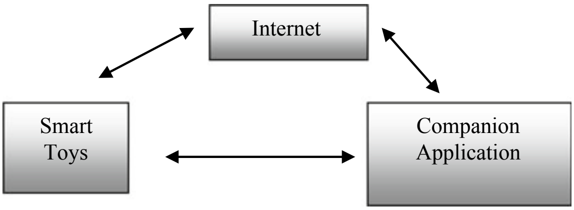
Healthcare Policies to Protect Data: The Health Insurance and Portability Accountability Act by US Government (HIPAA, 1996), Federal Information Security Management Act of 2002 (FISMA) [28], The Patient Protection and Affordable Care Act (ACA), Health Information Technology for Economic and Clinical Health Act (HITECH) are some regulations aimed at safeguarding patients' privacy. HIPPA insists organizations follow privacy rules and privacy rights standards while handling health information [34]. HIPPA proposes some physical and technical safeguards to achieve it. Physically securing workstations and devices, media controls and facility access controls are some physical safeguards proposed by HIPPA. Unique user identification number, emergency access procedure, automatic logoff, encryption and decryption are proposed as technical safeguards [30]. Health Level Seven International (HL7) is another standard accredited by ANSI. It is a comprehensive framework of standards for the exchange, integration, sharing and retrieval of electronic health information. HL7 is a standard supported by more than 1600 members from more than 50 countries. It includes 500+ corporate members representing healthcare providers, government stakeholders, payers, pharmaceutical companies, vendors/suppliers and consulting firms [35]. The components of HL7 standards and their compatibility with GDPR are discussed by Mense et al. in their paper "HL7 standards and components to support the implementation of the European general data protection regulation" [36].

17.5.2 Privacy Threats and Toys

Toys are accepted as an integral part of a child's growth. A toy is an item or product intended for learning or play, which in effect gives numerous benefits to childhood development [37]. Education-focused toys help to learn with play. HOPSCOTCH is one of these kinds which help children to learn English vocabulary while jumping on a sensor pad [38]. When a traditional toy is incorporated with AI, IoT, sensors and cloud-based technologies it turns into a smart toy, and children show a greater fascination toward it. Smart toys are accompanied by a supporting application known as a companion app [39].

A smart toy is a traditional toy incorporated with electronic components, like a microphone, camera, speaker, GPS and sensors to identify the location, time, weather and many more real-time data [39]. The toy uses Wi-Fi or Bluetooth to connect

Fig. 17.1 Smart toys have their associated companion app. Both toys and app generate and communicate a lot of data



with a network. The collected data is stored in the cloud. These toys are capable of identifying the kid's interests over time and can self adapt to provide a more personalized experience for the child (Fig. 17.2).

Possible privacy attacks through toys

Connected devices are always acting as points for personal data theft and attacks, and in the case of toys, the target is more vulnerable [39]. Studies show smart toys are a distinctive surface of attack because the target is a unique age group and as kids show more trust and attachment to the toys, any unfavorable incident will amplify the negative effect [40]. Smart toys collecting personal and contextual information of the kid broadcast MAC address and it is potentially hazardous. Unprotected databases, weak passwords, insecure Wi-Fi networks and strangers communicating to a child keeping smart toys as a medium are raising an alarm on personal data protection of the child [39]. Smart toys are gathering information either in a passive way or active way [40]. A toy like Hello Barbie grabs personal information actively whereas many toys are passively doing it. Sphero BB-8 and Wiggy Piggy Bank are smart toys, where an unauthorized outsider could control the toy. The innocent nature of kids results in deep trust and can act as more easy open access points for personally identifiable sensitive information that may lead a kid to danger. Toys fetched through an unprotected unencrypted channel, an attacker can input disturbing audio or video content and make the kid psychologically scarred [40].

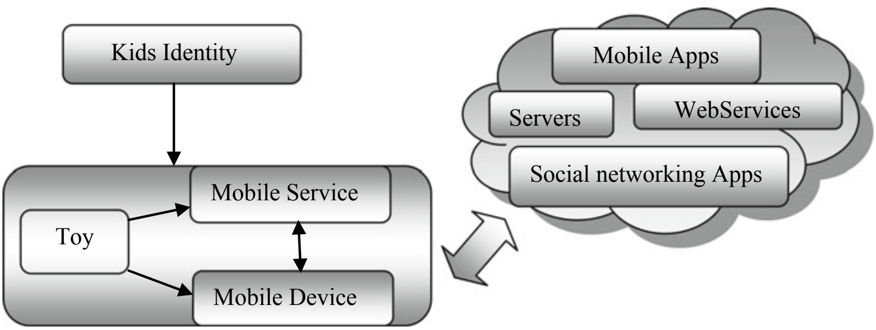


Fig. 17.2 The smart toy and companion app may challenge privacy and cause risk for children if the generated data is reaching a malicious actor [39]

Smart toys have inherent vulnerabilities of technology incorporated within it, and when it is supported by a companion app, the issue doubles. Rapid7, i-Que, My Friend Cayla, HelloBarbie, Motherboard, CloudPet are some smart toys found vulnerable and some got banned by the government bodies or the vulnerability got fixed [40]. Smart Toy Monkey can establish Bluetooth connection to wireless network implicitly and such toys, which may accompany kid everywhere is ending up as the location tracker of the child. This toy can broadcast the MAC address of the device also.

17.5.3 Privacy Issues in Digital Forensic

Digital forensic is a detailed investigation of cybercrime through the examination of digital storage and other digital environments. It comprises a set of techniques, methods and tools for collecting, analyzing and reporting digital data in a legally admissible way [41]. Digital forensic helps to reconstruct criminal events or it helps to anticipate a criminal event with the help of evidence [42]. The first step of the digital forensic investigative process is to identify whether a crime happened or not. If a crime is identified, the investigator starts collecting all data found in data storage of criminal, victim or those who are related to the crime and also will collect other digital evidence from the crime scene [43]. The collected data will be recorded and stored. A lot of personal private data dig out from heterogeneous digital storage devices will be revealed, analyzed and stored while such investigations are going on. EnCase Forensic 20.2 is a tool used in digital forensic and it can connect to the cloud and will collect user credentials forensically from cloud repositories [44]. Forensic ToolKit (FTK) is another one that operates on mobile interoperability and e-discovery technologies [45].

Gathering data that is part of an investigation is justifiable, but a full collection of data may lead to the revealing of personal information that is not part of the investigation subject [46]. When a digital crime is investigated, not a single data owner comes under the investigative process, and the related individual or even an organization data may get revealed [43, 46]. For example, if the investigator needs to check the corporate e-mail server, it is possible to look into other e-mails other than the one under investigation [46]. Data privacy in digital forensic must be ensured by clear policies and a systematic approach, so the investigator can handle personal data appropriately and effectively. Unfortunately, existing data protection acts or digital forensic investigation frameworks do not directly cover the protection of sensitive personal data collected in digital forensic [46]. But there are some research suggestions that trying to incorporate privacy protection along with a digital forensic investigation framework.

While considering a person's privacy as prime, forensic investigations must be able to filter and collect only relevant data [43]. This will make the investigation process cost-effective also. How to make this possible is an ongoing research.

Different levels of privacy for data under investigation are proposed by Waleed Halboob et al. [43]. With respect to the individual who handles the data, its privacy perspective varies. In forensic view, data can be relevant or non-relevant, whereas from the data owner's view, it is private and non-private data. A data owner can decide the level of privacy for his data. But to ensure the protection of the data, the investigator must provide policies and add on safety measures so that the trust element between both the parties can be maintained. This research views data from the forensic and data owner perspectives. Three possibilities are existing now: (1) Non-relevant, non-private data is not significant for both investigator and data owner and non-relevant private data is the one where the investigator is least interested. Both the categories can be omitted from the collection. (2) Relevant, non-private data can be collected without privacy concern. (3) Relevant, private data must be handled with great care by the investigator without harming the owner's privacy concerns.

17.5.4 Privacy Issues in Image and Video Surveillance

Video surveillance is considered a strong mechanism to strengthen security, especially in public places. Statistics [UITP] finds a positive response from the users of public transport because of the enhanced security they feel with installed surveillance. The video surveillance footage helps to identify and analyze the scene of incidents, as in the case of the Brussels bombing attack on March 22, 2016 [47].

Video surveillance at public places is appreciable because of its positive results, but unwanted surveillance in public areas is not favorable in terms of privacy. A structured and rational weighing process between the privacy values and security requirements with definite criteria from authorities is needed before deciding surveillance [48].

The following procedures are recommended by Westin et al. to decide surveillance while balancing privacy.

Measure the seriousness of the need to conduct surveillance. The reason to conduct surveillance must have genuine social importance and should not be a license to invade privacy. Seriousness for surveillance can be re-examined for the enforcement of the law against deviant or more political conduct.

Find if any alternative other than surveillance. The surveillance is supporting crime investigation in public places, and also helps in personality testing and can monitor petty industrial thefts. All these scenarios must opt for lesser privacy-invading mechanisms rather than selecting surveillance as if it is the single existing method.

Decide the degree of reliability required for the surveillance device. Depending on the type of data that needs to be extracted by the device the degree of trust of the device varies.

Determining whether true consent for surveillance has been given or not. Any type of scientific experiment, educational study, the test of a new product and feedback of an entertainment performance must ensure the legal consent of the parties coming under surveillance.

Measure the capacity for limitation and control of surveillance if it is allowed. This is who, when and how the phase of judgment. Who is carrying out surveillance, for example, police or a private investigator? Scope, duration and operation of the surveillance must set. A general agency such as law enforcement officials must set the standards, practices and rules to perform surveillance.

17.6 Data Privacy and Advanced Technologies

Social networking is the totality of electronic gadgets and transmissions, whereas the real world is the physical world of human interactions. The personal identity in the real world is important as it is a gesture of goodwill; it develops mutual confidence and honesty; initiates the next round of communication; and helps to associate transactions and information with other persons. But the identity of a person varies in the real world, like pen name and real name of a novelist [22]. Social networking helps to create a world of virtual relationships; many of its users are in an ambiguous state to decide how much of their identity must be revealed.

Advanced technologies provide a lot of support to day-to-day life and life in the virtual world, but it raises many privacy concerns also. The following section verifies some trending technologies in terms of privacy.

17.6.1 Privacy and Internet of Things (IoT)

Internet of Things is the technology of connecting physical objects called things, which have sensing and communicating capabilities. Sensors and actuators form the hardware part of an IoT device which can work hand in hand with nodes like micro-controllers, system-on-chips, mobile phones, miniaturized single-board computers, cloud platforms along with the embedded software. Internet of Things extends computing capabilities and network connectivity to objects or items that are not normally considered computers. With minimum human intervention, these non-computers can generate data with the help of sensors and can exchange and consume data. There is no universal definition for IoT we can find [49].

Figure 17.3 shows a typical data flow and associated privacy preservation activities in IoT applications.

IoT is proved as a successful technology because of its universally accepted useful implementations. The technology was able to penetrate into the daily life of common people with its promising possibilities. Connected communicating objects exchange generated information automatically and it pauses a caution on the security and privacy of the data [51]. IoT must keep trust, transparency and control in order to ensure data privacy. To achieve these, IoT must support the owners of the IoT device to control the collected information. In the case of IoT, the owner of the device is the data owner also. Thus he/she must have complete control over the data collection through sensors and its transmission. The data owner must be able to track the access of the data from IoT devices that might be located either inside the home, car or even

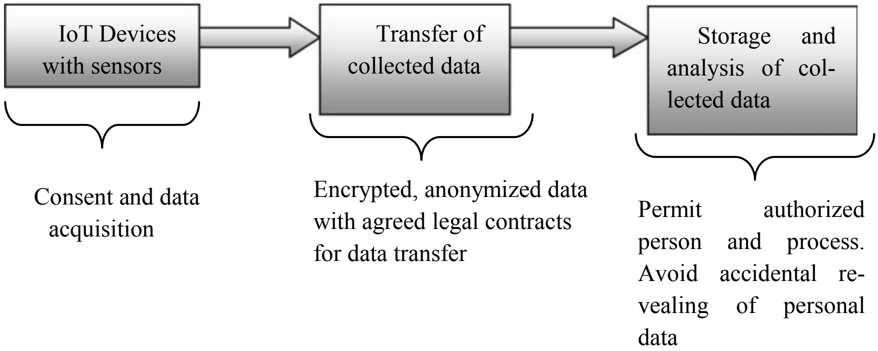


Fig. 17.3 IoT devices generate and transmit a lot of data, and care must be taken to ensure privacy at various phases of data handling

on a personal wearable. Control over the digital footprint, which gives information about the geographical movement of the device, is a privacy feature that IoT can offer. Hardly, no risk for the right to privacy is expected when people enjoying consumer IoT and its benefits. Here the consumer must have clear, informed judgment about the risks and benefits of using consumer IoT. In order to achieve it, strong user controls, proper notifications, transparency and fit-for-purpose governance of instruments are proposed. Sensor technologies are reduced in cost and size. This helps to develop devices that are smaller in size but with a more accurate collection of personal data like facial emotions and minor body movements also. Technologically this sounds positive but raising a lot of challenges in terms of privacy.

Manually activated devices need human physical activity, like a toggling of switch or change of regulator to start functioning. Microphone-enabled speech-activated devices need a wake-up phrase But devices always "ON", like surveillance cameras, will capture data and transmit it continuously. The following enlists the privacy issues with always-on IoT devices.

(1) "Always-On" devices are riskier than devices that need a human interaction to be activated [52]. Home security cameras, baby monitoring devices and wearable video camera for visually impaired all are always on devices and raises more security concerns. These devices are capable of sensing, collecting and transmitting even facial expressions, emotions and other sensitive information which is not favorable.

(2) Identity of the individuals will be revealed in private and public spaces. The presence of high-definition cameras with face recognition and analysis technology makes people be identified without their consent. This is a privacy concern for a normal human being.

(3) Private space of family may get violated by interconnected devices. Data in private space is gathered and used by third parties.

4) As many devices go IoT-based in a home, for example, each needs a separate controller as per the manufacturer, and the consumer cannot work on a single

controller. Reduced consumer choice with increased vendor lock-in is the result of this.

(5) IoT-based devices have the capability to fade behind the background and the user may forget the presence of the device which leads to unwanted data leaks.

17.6.2 Privacy and Big Data Analytics (BDA)

Artificial intelligence, machine learning and data sciences are interrelated technologies, mostly overlapping each other. Some data privacy issues are common among all these technologies. Data is the fuel of digital technology. The daily life of a person, business decisions of an organization, all influenced by intelligence and data analysis performed by electronic machines around us. Big data, analysis and conclusions offer a lot of support and luxury in all dimensions of human life. Social networking sites and applications depend a lot on BDA for relationship creation.

Big data analytics (BDA) is the study, analysis and forecasting based on a huge set of available data; in major cases, the data set contains personal information. Data protection and data-driven innovations under the umbrella of data sciences take diverging paths. Data protection demands a clear and defined purpose of use of personal information, while BDA explores data to find a purpose. The data privacy challenges that BDA raises can be classified into two:

Type 1: Challenges because of the characteristics of BDA.

Type 2: Challenges from the lack of sufficient supporting technology or tools.

Challenges because of the characteristics of BDA

Profiling and predictions based on large data sets of personal data may lead to unexpected exposure to an individual's private behavior [53].

Data generalization and synthesis of data from available data set may lead to the identification of personal sensitive information from non-personal data and hence will challenge data anonymization and data pseudonymization requirements insisted by the data protection laws [53].

Data controllers and data processors are more at risk due to the *linkability* property of data. Linkability is the possibility of revealing any personal information accidentally during an analysis of the non-personal data [54].

There is a difficulty to track and report the processing of big data with the consent of the data owner. Using personal data with the consent of the data subject is a mandatory requirement of data protection law [54].

A good analytical result needs a good training set with diverse data from various sources. This training set information may use data that fall in the category of sensitive personal information and it is objectionable according to the regulatory measures [53].

Challenges from the lack of sufficient supporting technology or tools

The lack of technical measures for easy enforcement of current regulatory measures for data protection is a challenge. Technologies must develop so that data controllers and data subjects can easily define the collection and use of personal data for large-scale commercial needs [53].

There is a need for the best available data analytics to cope with encrypted or anonymized data. Existing encrypted data processing techniques guarantee stronger privacy, but cannot scale it for larger data sets. Handling different and co-existing data types are also a concern [53].

New tools are important to assess the risk of retrieving personally identifiable information by the data controller, when processing combinations of anonymized, pseudonymized, even public data sets. Risk assessment and mitigation activities have to be carried out increasingly in an online and automatic fashion in order to react to changing risk levels during such operations [53].

The dynamic nature of new technologies like cloud and fog computing changes the level of risk of data revealing, and this increases the cost and performance overhead to optimize data protection [53].

Privacy-by-design strategy is a proposed solution for the conflicts existing between BDA and privacy protection [55]. Hoepman et al. define two categories of design strategies: (1) data-oriented strategies and (2) process-oriented strategies.

17.6.3 Privacy and Blockchain Technology

Blockchain is a promising technology with huge positive support from organizations and governments around the world. Blockchain boosts the bitcoin-based financial industry. Many IT vendors, internet and consulting firms are investing a lot for research and business in this field [56]. At the same time, distributed ledger gives secure transactions in a shared database with peers, even with anonymous parties. Prognosticating the influence of blockchain in the future, governments around the world released white papers and roadmaps to support the environment for the growth and deployment of this technology.

Blockchain supports privacy, as found in use cases like smart contract development, supply chain management, asset registers, real estate, healthcare and the retail industry. Proof-of-work, proof-of-stake and proof-of authority are various algorithms acting as consensus mechanisms to verify the legitimacy of the transaction. Transaction immutability is the strong benefit of blockchain, where the blockchain cryptocurrency preserves the basic requirements of information security such as consistency, tamper resistance and resistance to DDoS attacks [56].

The structure of the blockchain transaction model is different from the traditional transaction model. Figure 17.4 depicts the varying scenario of these transaction models.

Existing data protection laws are focusing on the traditional transaction model shown in Fig. 17.4a and peer-to-peer transaction privacy is not under the light of law designing. Blockchain technology has inherent support to privacy with encryption and data integrity verification but distributed peer-to-peer structure is not coming under the scope of data protection laws like GDPR and CCPA [57]. Organizations must ensure privacy by conducting a privacy impact assessment (PIA) or data protection impact assessment (DPIA) before the implementation of blockchain technology.

The following are tensions between blockchain technology and data privacy requirements [57]:

Applicability of data protection and privacy laws in varying perspectives of *anonymity* and *pseudonymity*.

As there is no central agency involved in conducting a transaction and because of the peer-to-peer distributed architecture of blockchain, it is unclear that who will decide the purpose of data processing. How to identify data processors and data controllers in blockchain implemented environment is a major challenge.

Evaluating jurisdiction and applying regulations to decentralized blockchain implementations is not a straightforward exercise compared to traditional centralized systems. Distributed nature of blockchain holds transactions across peers in varying territories and privacy law needs to address territorial implications for distributed blockchain networks. The law should enforce conditions and restrictions for cross-border transactions with personal data.

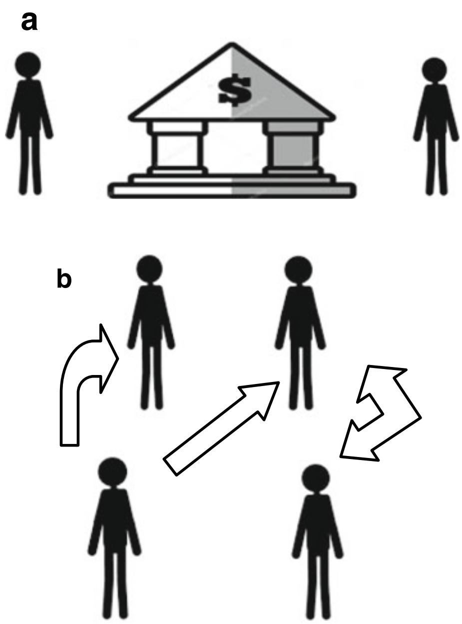


Fig. 17.4 **a** Traditional transaction includes central monitoring or controller. **b**. Blockchain model includes distributed peer-to-peer network transactions without any middle agent

The immutability and data preservation properties of blockchain applications and individual's rights must go conjoint. Data protection law tries to ensure control over the personal data by the data owner. Because of immutability property, blockchain will not delete the data in the record, and the transaction updates are always preserved. There exists a conflict between "the right to be forgotten" nature of the law with immutable and data-preserving nature of blockchain.

Applying criteria for legitimate reasons for processing personal data to blockchain use cases is a challenge.

17.6.4 Privacy in Artificial Intelligence (AI), Machine Learning (ML) and Deep Learning

Privacy issues in artificial intelligence can be seen as an extension of privacy violations created by big data [58]. AI uses a huge amount of data to learn, develop adaptive models and make actionable decisions. The outcome of the AI neural network is influenced by assumptions and biases. An unexpected result from an AI network is creating risk for the related decision-making process.

Privacy issues in AI: In manually controlled data processing, the natural human sense applied may help to filter out unwanted or unexpected turns happening during the process. But when big data is in challenging size for human control, AI is in demand. It drastically reduces the degree of human control over the data and thereby the decision-making is purely dependent on the accuracy of the neural network training.

When a surveillance camera in a public place is combined with face recognition, which is an AI technology, it pauses a data privacy threat. Addressing machine as a human personality with the help of speech recognition technology gives the machine a more human feel, for example, Alexa or Google. This will lead to a more trusted relation of man with machines and eventually unintentional personal information sharing.

Adaptive learning algorithms learn from the provided information, and the data owner cannot predict the conclusions that AI reaches with his information. The consent of the data subject and notification about data usage to the data subject are difficult to follow when the machine changes data in an unpredictable way.

The ability of AI to identify patterns which is never noticeable for the human brain questions the anonymity requirement of data protection. Conclusions drawn by the machine from knowingly shared information will lead to accidental disclosure of personal data and it may be sensitive intentionally undisclosed information.

Privacy Issues in Machine Learning and Deep Learning

Figure 17.5 shows machine learning (ML), artificial intelligence (AI) and deep learning as overlapping domains [59]. Various machine learning algorithms for supervised, unsupervised and reinforcement learning are learned to predict according to

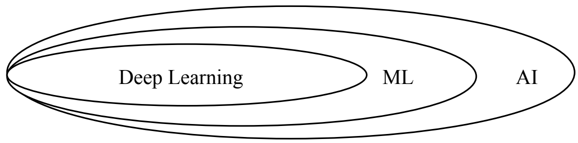


Fig. 17.5 Overlapping domains of AI, ML and deep learning [59]

the input. Since machine learning is the sub-element of AI, the privacy issues existing with AI are also found in machine learning.

17.7 Transparency and Accountability Measures to Ensure Data Privacy

In order to ensure personal data privacy and security, various acts impose the following measures [16, 17].

Privacy by Design: As personal data privacy and the right to privacy are dynamic and pose risks, a data controller must implement appropriate technical and organizational measures to ensure it. Pseudonymisation and data minimization techniques can be implemented for this purpose. These implementations must do in an effective manner to integrate the necessary safeguards into personal data processing, and thereby meeting the requirements mentioned by privacy acts.

Transparency: Transparency is expected in the access and processing of personal data. The information and communications that happen between the data subject and data controller in this regard must have easy understanding and clarity. This agreement must be written in clear and plain language. The important operations on personal data during its processing must be notified to the data subject by the data controller.

Security Safeguards: Data controller must apply various security safeguards and also must conduct periodic reviews to ensure its efficiency.

Notifying and dealing Persona Data Breach: The data controller must notify the authority if any personal data breach in data is processed by him. Such breach is likely to cause harm to any data subject. The authority will assess the severity of the harm that happened to the data subject, and can decide action and can ask to do some remedial actions from the side of the data controller. Authority can publicize the data breach.

Data Protection Impact Assessment: Data controller must conduct a data protection impact assessment if there is a change in technology or strategy by his organization.

Record-Keeping: The data controller must maintain records of important operations performed on data during collection, processing and transfer of the data. A record of periodic review of security safeguards and data protection impact assessments is to be maintained by the data controller.

Data Audits: Independent data auditor must conduct annual auditing of the personal data handled by the data controller and thereby verify its compliance with the law. The audit covers (i) clarity and effectiveness of notices given to data subject, (ii) effectiveness of measures adopted, (iii) transparency in relation to processing activities, (iv) security safeguards adopted, (v) instances of personal data breach and (vi) response of the data controller, including the promptness of notification to the authority.

Data Protection Officer: A dedicated data protection officer must be appointed by a data controller. Data protection officers will provide information and advice to the data controller on matters relating to fulfilling its obligations related to the law. The data protection officer will monitor personal data processing activities by the data controller to ensure that such processing does not violate the privacy act. It is his responsibility to provide advice about the situations where data protection impact assessments must be carried out and must carry out the review of such assessment. He also advises the data controller on internal mechanisms that may be developed in order to satisfy the data subject and providing assistance to and cooperating with the Authority on matters of compliance of the data controller. The data protection officer acts as the point of contact for the data subject for the purpose of raising grievances to the data controller and also this officer will maintain an inventory of all records maintained by the data controller in terms of privacy.

Processing by entities other than the data controller: Only one data processor can be appointed by the data controller. They must have a valid contract between them. The data processor, and any employee of the data controller or the data processor, must process personal data in accordance with the instructions of the data controller.

Classification of data fiduciaries as significant data fiduciaries: The authority can specify certain data controllers or classes of data controllers as significant. The significance is decided by (a) the volume of personal data processed, (b) sensitivity of personal data processed, (c) turnover of the data controller, (d) risk of harm resulting from any processing or any kind of processing undertaken by the controller and (e) the use of new technologies for processing. Such data controllers must be registered with the authority.

Grievance Redressal: Every data controller must have proper procedures and effective mechanisms to address grievances of the data subject efficiently in a speedy manner. A data protection officer or any other dedicated officer will address the grievances.

17.8 Conclusion

Socializing is a normal and quite an essential need of humans. The way to fulfill this need varied time-to-time from human origin. Technology-supported relation-building intensified with various forms of social networking. The glorified efficiency of social networking is in question when it hampers personal data privacy. Globally, many laws are framed to protect privacy. A large volume of stakeholders with varying requirements is a hurdle to implement it. Advanced technologies can support social networking to progress without hurting data privacy, and for this, a lot of research is required.

References

1. Ortlieb, M.: The Anthropologist's View on Privacy, Google Switzerland, Co published by the IEEE Computer and Reliability Societies (2014)
2. Arendt, H.: The Human Condition, Introduction by Margaret Canovan. 2nd ed. (1998)
3. Han Van der Geest, S.: Privacy from an anthropological perspective. In: Van der Sloot, B., De Groot, A. (eds.) The Handbook of Privacy Studies: An Interdisciplinary Introduction, pp. 413–444. Amsterdam University Press, Amsterdam (2018). JSTOR, www.jstor.org/stable/j.ctvcmxpmp.24. Accessed 9 July 2020
4. Warren, S.D., Brandeis, L.D.: The right to privacy. Harvard Law Rev. **4**(5), pp. 193–220 (1890). JSTOR, www.jstor.org/stable/1321160. Accessed 21 August 2020
5. https://plato.stanford.edu/entries/privacy/#PriConOveInf. Accessed 29 June 2020
6. Glancy, D.J.: The invention of the right to privacy. Arizona Law Rev. **21**(1) (1979)
7. Cooley, T.M.: A Treatise on the Law of Torts or the Wrongs Which Arise Independent of Contract. Callaghan and Company, Chicago (1879)
8. Thomson, J.J.: The right to privacy. Philos. Public Aff. **4**(4), 295–314 (Summer, 1975)
9. Scott, G.: The Death of Privacy: The Battle for Personal Privacy in the Courts, the Media, and Society. ASJA Press (1995)
10. Regulations (EU) 2016/679 of the European Parliament and of the Council of 27 April 2016, https://eur-lex.europa.eu/legal-content/EN/TXT/PDF/?uri=CELEX:32016R0679
11. Regulation (EU) 2016/679 of the European Parliament and of the Council of 27 April 2016 on the protection of natural persons with regard to the processing of personal data and on the free movement of such data, and repealing Directive 95/46/EC (General Data Protection Regulation) https://gdpr.eu/tag/gdpr/. Accessed on 16 July 2020
12. https://ico.org.uk/media/2614158/ico-introduction-to-the-data-protection-bill.pdf
13. https://www.dlapiperdataprotection.com/system/modules/za.co.heliosdesign.dla.lotw.data_protection/functions/handbook.pdf?country-1=AU. Accessed 16 July 2020
14. https://www.dlapiperdataprotection.com/system/modules/za.co.heliosdesign.dla.lotw.data_protection/functions/handbook.pdf?country-1=BR. Accessed 21 August 2020
15. Personal Information Protection and Electronic Documents Act https://laws-lois.justice.gc.ca/ENG/ACTS/P-8.6/index.html. Accessed 21 August 2020
16. Regulation (EU) 2016/679 of the European Parliament and of the Council of 27 April 2016 (General Data Protection Regulation), p. 9
17. The Data Protection Bill: https://meity.gov.in/writereaddata/files/Personal_Data_Protection_Bill,2018.pdf (2018). Accessed 21 August 2020
18. An overview of the Data Protection Act 2018, Version 2 January 2019, Information commissioner's office. https://ico.org.uk/media/2614158/ico-introduction-to-the-data-protection-bill.pdf. Accessed 21 August 2020
19. van der Geest, S.: Privacy from an Anthropological Perspective, Hand book of privacy
20. Glancy, D.J: The invention of the right to privacy. Arizona Law Rev. **21**(1) (1979)
21. https://whatis.techtarget.com/definition/data-anonymization
22. Kabay, M.E.: Anonymity and pseudonymity in cyberspace: deindividuation, incivility and lawlessness versus freedom and privacy. In: Annual Conference of the European Institute for Computer Anti-virus Research (EICAR), Munich, Germany (1998)
23. https://dataprivacymanager.net/pseudonymization-according-to-the-gdpr. Accessed on 22 August 2020

24. https://www.privitar.com/data-privacy/de-dentification/ generalization. Accessed 25 August 2020
25. Dwork, C., Roth, A.: The Algorithmic Foundations of Differential Privacy. Foundations and Trends® in Theoretical Computer Science, vol. 9, nos. 3–4, pp. 211–407 (2014). doi:https://doi.org/10.1561/0400000042.page-6
26. Poosarla, S.: Privacy & Data Protection, Infosys Ltd., India, Significance of Data Privacy in Indian IT Industry-A Brief Perspective from Infosys
27. Rahmouni, H.B., Essefi, I., Ladeb, M.F.: Enhanced privacy governance in Health information systems through business process modelling and HL7. Procedia Comput. Sci. **164**, 706–713 (2019), ISSN 1877-0509, doi:https://doi.org/10.1016/j.procs.2019.12.239
28. Hathaliya, J.J., Tanwar, S.: An exhaustive survey on security and privacy issues in Healthcare 4.0, doi:https://doi.org/10.1016/j.comcom.2020.02.018
29. Thimbleby, H.: Technology and the future of healthcare. J Public Health Res. **2**, e28. College of Science, Swansea University, UK (2013)
30. Kruse, C.S. et al.: Cybersecurity in healthcare: a systematic review of modern threats and trends. Technol. Health Care **25**, 1–10 (2017) IOS Press. doi: https://doi.org/10.3233/THC-161263
31. https://www.ibm.com/downloads/cas/PLWZ76MM. Accessed 21 August 2020
32. https://www.cms.gov/Medicare-Medicaid-Coordination/Fraud-Prevention/Medicaid-Integrity-Education/Downloads/docmatters-ehr-providerfactsheet.pdf
33. For the Record: Protecting Electronic Health Information http://www.nap.edu/catalog/5595.html, Committee on Maintaining Privacy and Security in Health Care Applications of the National Information Infrastructure, Commission on Physical Sciences, Mathematics, and Applications, National Research Council, ISBN: 0-309-52425-3, p. 288, 6 x 9, (1997, p. 59)
34. https://www.hhs.gov/sites/default/files/privacysummary.pdf. Accessed 21 August 2021
35. https://www.hl7.org/. Accessed 21 August 2020
36. Mense, A., Blobel, B.: HL7 standards and components to support implementation of the European general data protection regulation. Eur J Biomed Inf **13**(1), 27–33 (2017)
37. Fantinato, M., Hung,, P.C.K., Jiang, Y., Roa, J., Villarreal, P., Melaisib, M., Amancioa, F.: A preliminary study of Hello Barbie in Brazil and Argentina.
38. Lucht, M., Domagk, S., Mohring, M.: Exer-learning games: transferring hopscotch from the schoolyard to the classroom. In Bramer, M. (ed.): IFIP AI 2010, IFIP AICT 331, pp. 25–34 (2010)
39. de Carvalho, L.G., Fantinato, M., Elera, M.M.: Security requirements identification and prioritization for smart toys. Electr. Commer. Res. Appl. **41**, 100972 (2020). doi:https://doi.org/10.1016/j.elerap.2020.100972
40. Shasha, S., Mahmoud, M., Mannan, M., Youssef, A.: Playing with danger: a taxonomy and evaluation of threats to smart toys. IEEE Internet Things J. **6**(2) (2019)
41. Kävrestad, J.: Fundamentals of Digital Forensics Theory, Methods, and Real-Life Applications, eBook ISBN 978-3-319-96319-8, doi:https://doi.org/10.1007/978-3-319-96319-8
42. Afifah, K., Perdana, R.S.: Development of search on encrypted data tools for privacy preserving in digital forensic. In: 2016 International Conference on Data and Software Engineering (ICoDSE), Denpasar, pp. 1–6 (2016) doi:https://doi.org/10.1109/ICODSE.2016.7936152
43. Halboob, W., Mahmod, R., Udzir, N.I., Abdullah, M.T.: Privacy levels for computer forensics: toward a more efficient privacy-preserving investigation. In: International Workshop on Cyber Security and Digital Investigation (CSDI 2015), Procedia Comp. Sci. 56, pp. 370–375 (2015)
44. http://www.guidancesoftware.com/encase-forensic. Accessed 22 August 2020
45. https://accessdata.com/products-services/forensic-toolkit-ftk. Accessed 22 August 2020
46. Law, F.Y.W., et al.: Protecting digital data privacy in computer forensic examination. In: 2011 Sixth IEEE International Workshop on Systematic Approaches to Digital Forensic Engineering, Oakland, CA, pp. 1–6 (2011). doi:https://doi.org/10.1109/SADFE.2011.15
47. International Trends In Video Surveillancepublic Transport Gets Smarter, Statistics Brief, UITP, https://Www.Uitp.Org/Sites/Default/Files/Cck-Focus-Papers-Files/1809-Statistics%20Brief%20-%20Videosurveillance-Final.Pdf. September 2018.

48. Westin, A.F.: Science, privacy, and freedom: issues and proposals for the 1970's. Part II: balancing the conflicting demands of privacy, disclosure, and surveillance. Columbia Law Rev. **66**(7), 1205–1253 (1966). JSTOR, www.jstor.org/stable/1120983. Accessed 14 August 2020
49. https://www.internetsociety.org/wp-content/uploads/2017/08/ISOC-IoT-Overview-20151221-en.pdf
50. Perera, C., Barhamgi, M., Bandara, A.K., et al.: Designing privacy-aware internet of things applications. Inf Sci **512**, 238–257 (2020)
51. Internet Society Policy Brief: IoT Privacy for Policymakers. https://www.internetsociety.org/policybriefs/iot-privacy-for-policymakers. Accessed 22 August 2020
52. Always On: Privacy Implications of Microphone-Enabled Devices. https://fpf.org/wp-content/uploads/2016/04/FPF_Always_On_WP.pdf. Accessed 22 August 2020
53. Timan, T., Mann, Z.Á. (eds.) Data protection in the era of artificial intelligence. Trends, existing solutions and recommendations for privacy-preserving technologies, BDVA (2019)
54. Bloem, J., van Doorn, M., Duivestein, S., van Manen, T., van Ommeren, E.: Privacy, technology and the law Big Data for everyone through good design, VINT research report 3 of 4 (2013)
55. Hoepman, J.-H.: Privacy Design Strategies (The Little Blue Book) (2020)
56. Zhang, R., Xue, R., Liu, L.: Security and privacy on blockchain. ACM Comput. Surv. 1, 1, Article 1, 35 (2019). doi:https://doi.org/10.1145/3316481
57. Shah, P., Forester, D., Polk, D., Wardwell, L.L.P., Berberich, M., Raspé, C., Mueller, H.: Blockchain Technology: Data Privacy, Issues and Potential Mitigation Strategies, Resource ID: W-021-8235, © 2019 Thomson Reuters, https://www.davispolk.com/files/blockchain_technology _data_privacy_issues_and_potential_mitigation_strategies_w-021–8235.pdf
58. Artificial intelligence and privacy Issues paper June 2018
59. The Privacy Expert's Guide to Artificial Intelligence and Machine Learning October 2018. https://iapp.org/media/pdf/resource_center/FPF_ Artificial_Intelligence_Digital.pdf. Accessed 23 August 2020

Chapter 18
Deep Learning Techniques for Social Media Analytics

Muralidhar Kurni, M. Mrunalini, and K. Saritha

Abstract Machine learning has seized both academia and industry's attention as deep learning (DL) is the frontrunner in data science. In order to construct computational models, DL uses multiple layers to epitomize data theories. A few of the key DL techniques, such as model transfer (MT), convolutional neural networks (CNN), and generative adversarial networks (GAN), have completely altered our understanding of information processing. Indeed, DL's processing power while handling images, text, and speech is truly remarkable. Because of the rapid growth and extensive availability of digitized social media (SM), evaluating these data by employing conventional technologies and tools is complex and unmanageable. These challenges are expected to be well managed through solutions offered by DL methods. Hence, we consider the executed DL methods built-in regard to social media analytics (SMA). However, rather than engaging in technical details, we study domains that pose serious challenges to SM where DL is involved and propose solutions to those challenges. We also present a few case studies.

18.1 Social Media and Social Media Analytics

18.1.1 What is Social Media?

The term "Social Media" refers to an application or website developed to enable people to exchange information swiftly, easily, and efficiently in real time. Most people think of social media as an application specifically developed to use in tablets

M. Kurni (✉)
Department of CSE, Anantha Lakshmi Institute of Technology and Sciences, Ananthapuram, India
e-mail: muralidhar.kurni@ieee.org.com

M. Mrunalini
Department of Computer Applications, MSRIT, Bangalore, India

K. Saritha
Sri Venkateswara Degree and PG College, Ananthapuram, India

A. Biswas et al. (eds.), *Principles of Social Networking*,
Smart Innovation, Systems and Technologies 246,
https://doi.org/10.1007/978-981-16-3398-0_18

and smartphones, but in reality, social media developed as a communication tool to be used in computers. The wrong impression is because many people connect their devices through apps.

The ability to share views, events, and images in real time has revolutionized our lives and the way we do business. Retailers who generally employ social media as an indispensable tool in their marketing plan began to get beneficial and consistent results. Rather than treating it as a bonus add-on, investing time, planning, and strategy to develop a strong and effective social media campaign offers huge dividends in a marketing plan [1].

18.1.2 Categories of Social Media

We can categorize social media into 10 types depending on their usage and applicability [2]:

1. **Social Networks**—Links among people.
2. **Discussion Forums**—Apportion suggestions and news.
3. **Bookmarking and Content Administered Networks**—Unearth, store, and apportion the latest content.
4. **Media Sharing Networks**—Apportions images, videos, etc., and various other media file types.
5. **Social Shopping Networks**—Online shopping.
6. **Blogging and Publishing Networks**—Broadcast contents on online platforms.
7. **Consumer Review Networks**—Identify and assess businesses.
8. **Interest-Based Networks**—Apportion amusements and interests.
9. **Anonymous Social Networks**—Connect incognito.
10. **Sharing Economy Networks**—Merchandise services and stocks.

18.1.3 Examples of Social Media

A few of the prominent platforms of social media are explained below [3].

- *Wikipedia* was developed by the collective efforts of a user community called "Wikipedians" and is an unpaid, open-information encyclopedia available online. Any registered user can present an article to publish it. Also, users who wish to edit articles need not be a registered user. Wikipedia was established in January 2001.
- *Facebook* is a prominent, open-source, unpaid social networking platform, which permits legitimate consumers to generate profiles, drop messages, upload images, and videos. Facebook enables users to connect with friends, relatives, associates, etc.

- *LinkedIn* is a social networking website specially built for the business community. This site enables legitimate users to document and authorize a network of people they recognize and trust professionally.
- *Twitter* offers a micro-blogging facility to legitimate consumers to share abbreviated posts named "tweets." Twitter is an open-source, unpaid service where members can send tweets and follow others' tweets through various operating environments and devices.
- *Google+* (read as Google plus) was a social networking environment designed to clone offline interaction methods among people. It compares favorably with other social networking environments. However, Google+ stopped its services to fresh users and finally closed in 2019.
- *Pinterest* is a socially organized web meant to share and categorize photos present online. This web handles only brief explanations and focuses on the visual. If you click an image, you will be re-directed to its source. For instance, if you click on a picture of leather bags, you will be taken to the seller's website, and an image of pizza will take you to the related pizza shop website.
- *Reddit* is a forum and social news website where the site members present and foster stories. This web has several subsidiary communities called "subreddits," where every subreddit contains a particular topic: music, politics, or technology. Members of Reddit web are also called "Redditors" who present content, which is polled by various other web members. The main objective is to present stories considered the best to the website's main thread page's top-most position.

18.1.4 The Importance of Social Media Analytics

There is extensive information available on social media. Such information is called social media analytics. When you share an update, it creates interplay with inputs from your followers and your reply to their comments, and together they combine to produce exceptionally significant content. Such distinct information proves very useful in numerous ways in marketing activities on social media [4].

What is Social Media Analytics? "Social Media Analytics" (SMA) is defined as a method of gathering information from blogs and various social media websites, which are further assessed to aid in developing business strategies. While the regular observation of tweets and analysis of "likes" and "re-tweets" helps in gauging consumer sentiments, SMA helps gain a deeper understanding of consumer preferences.

However, it is to be noted that social media websites are not confined to popular platforms like Twitter and Facebook alone, along with their "likes" and "retweets," and focus on blogs, forums, and review news websites.

That said, when you begin working with social media analytics, it would be difficult, at least initially, to gauge the actual value or worth of a brand because popular phrases lose their actual meaning as used by netizens. Moreover, multitudes of bots, spams, trolls, etc., add their own muddle and confusion.

However, a systematic and focused effort using modern analytical tools provides rich consumer information that is not available anywhere else. Thus, social media analytics may be defined as a mass of information revealed and collected through various methods and means from several sources; each one studied individually [5].

Social Media Analytics—Why should you do it?

- *Social Media Analytics helps you understand the behavior and preferences of your followers.* Interpreting the online activities of netizens accurately could benefit you in numerous ways. Say, for instance, assessing a previous post's reaction could help you identify the most effective time to share the next one. Posting at the right time is significant in social media marketing. Indeed, your posts must appear while most of your intended followers are online to garners maximum attention and consequently boost traffic and sales. This is one example where methodical analysis of social data pays good dividends. Certain social networking platforms allow their users to do these tasks because of the in-built analytics in them. For instance, it is available in the "posts" section on your profile page on Facebook. This will display the date and time when a large number of page followers are available online. Posting at these times ensures a higher reach.
- *Social Media Analytics helps you to visualize the popular social networking sites.* One important matter to understand in social media is that a mere number of users do not mean higher utility and value. For example, Facebook has 2 billion-plus users, Instagram more than 800 million, but smaller social networking platforms such as Flickr or Pinterest could support you better in implementing your business plans because functionally, they are more effective. Hence, it is important first to identify the ideal social networking site suited for promoting your product or service to implement a successful campaign. Next, you could decide how long to manage the selected sites for maximum benefit, focusing more on sites that perform better and less on those that produce lesser results. A tool called "Cyfe" helps in this process. "Cyfe" is a tool used on a social media dashboard that establishes links to various social networks. The collected data from these platforms can be put together in a single dashboard to evaluate their performance.
- *Social information could support you to build finer contents.* Analyzing social networks could help in gaining a better understanding of the content that produces better results. For instance, Twitter and Facebook, popular social media platforms, can be analyzed to ascertain which links, videos, or images perform better. In the case of Instagram and Pinterest, the visual-centric networks make it easy to determine what kind of images produces finer results. Thus, social media analytics can be used to establish which content produces the best results. Few social networking websites such as Facebook contain in-built analytics to help understand the kind of media content that produces finer results, and the same can be seen under the "Posts" link on the user's FB page. Further, analytics can also be used to examine the performance of your blogs. Through these, a user can gather quite a lot of important information: the content shared much, how many times a user visited various social networks, how long he remained there, and the number of people who recorded sales and joined subscriptions. Tools such as "Buzzsumo"

are used to examine how well your content performs on the social media web. Google Analytics can be used to identify the proportion of traffic a social network consumes, and it is also used to show various parameters such as bounce rate, how long users consumed on your web, and how many pages they had seen.

- *Assists you to get knowledge about your competitors.* Your competitors in the business build content and implement various social media plans that produce their own distinctive style and image. Assessing these data could help in understanding which content worked fine and which one did not. Such an analysis allows you to avoid the mistakes they made and build techniques that could yield fruitful results. "Similar Web" can help you in identifying the social networks that worked better for your competitors. Adding the URL to any of the social media webs will help you visualize the amount of traffic your web will draw from the social media and the social networks that steer it.
- *Social data can support you to build a better plan.* It would be difficult to build a perfect plan in your first attempt itself. You are likely to commit mistakes and probably missed out on techniques that could have performed better for you. A concerted effort in studying social media analytics lets you understand these errors, and avoid them and create a stronger plan. Incorporating finer tools available in social media analytics will especially help in identifying these errors. Careful "listening" to social media with the selected tools helps gauge your plan's influence on viewers. Social media listening is defined as a method to analyze online chats to know people's views about a particular product, company, individual, and the like and utilize it to support business better. Social media listening helps you to visualize in real-time people's thoughts about your online business. Hence, it supports you to alter your plan to establish better connectivity with the current and prospective consumers. "Sentione" is a tool used to implement social media listening.
- *You can assess the performance of the social media process through social media analytics.* It is possible to monitor the social media operation after launching it continuously. It is also possible to inspect whether it is traversing in the intended direction or not. If the launched operation is not traversing in the right direction, you can revise your operation and fix it, or if the outcome poses heavy damages, you can eliminate it in the initial stage itself. A tool called "Sentione" is used to undertake a thorough analysis of your operations to understand whether the outcomes produce a positive, negative, or neutral effect. Following this analysis, you can then proceed to do the needful.

From the above discussion, it is evident that social media analytics provides multiple advantages. It can support you to enhance your availability both in limited and longer periods. Your social media analytics data further ensure that you get the best results for the time you spend on it. To get optimal outcomes, it is imperative to utilize the best available tools to inspect your data analytics and draw significant conclusions.

18.2 Deep Learning

18.2.1 What is Deep Learning?

Though deep learning (DL) evolved in 1940 [6], DL algorithms actually became active in 2006 when Hinton presented the layer-wise greedy learning method. These DL algorithms aimed to overcome the defects present in the neural network (NN) technique in identifying the best point by catching the optimum local point, which is aggravated if the size of the training information is not sufficient. Hinton's main aim in presenting this method is to utilize the unsupervised learning technique ahead of the layer-by-layer process [7].

A deep learning algorithm inspired by the human brain's stratified form draws out intricate hidden characteristics with a higher extraction grade. The layered construction of DL algorithms functions productively if huge quantities of unstructured data were available. DL algorithms focus on utilizing diversified transformation layers and ascertain if each layer output characterization had taken place [8]. Big Data Analytics contains the completely acquired basic understanding attained from DL. It has become an advantageous tool for big data analytics because of the significant characteristic that abstracts fundamental characteristics in massive quantities of data [8]. Figure 18.1 shows the semantic approach to deep learning.

The emergence of DL as a sub-ordinate division of machine learning was incorporated into account of events, such as the growth of chip processing, resulting in massive data quantities. DL reduces computers' hardware prices and has also seen significant growth in the developed machine learning algorithms.

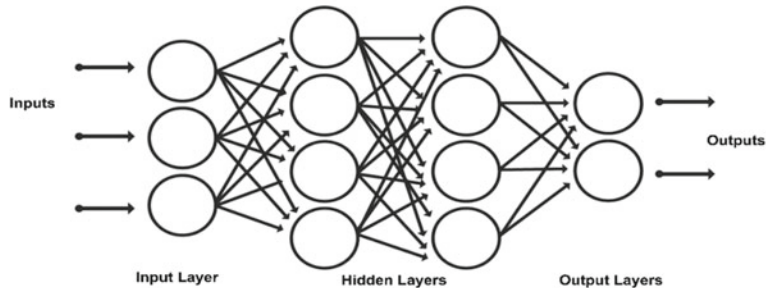


Fig. 18.1 Deep network architecture with multiple layers

18.2.2 Types of Deep Learning Algorithms

The different types of DL algorithms are explained below:

Convolutional Neural Networks (CNN). This concept was developed with inspiration from neural network models. CNN is a category of DL algorithm that contains a "sub-sampling layer" and a "convolutional layer" framework. Multi-instance data content has been set up with many instances where every data point contains a set of instances [9]. CNN is familiar with its three characteristics, such as "weight-sharing," "sub-sampling," and "local field." CNN contains three layers, namely, an input layer, a convolutional layer, and an output layer. Convolutional layers are divided into a "sub-sampling layer" and a "convolutional layer." The input layer is called the "hidden layer." In a hidden layer, every convolutional layer falls after the sub-sampling layer. The entire operation involved in CNN training is implemented through two different phases. In the case of a "feed-forward phase," the outcome of the earlier stage will be passed on to the next stage, whereas in the case of a "backpropagation phase," it corrects errors, and those changes are carried out by the method of distributing training errors and through a stratified process [10]. The convolution process is carried out in the initial (first) layer where, in every instance, several filtering stages occur. Moreover, nonlinear transformation happens because the outcome of the earlier phase is reconstructed into a nonlinear phase. Furthermore, the reconstructed nonlinear phase is contemplated in the max-pooling layer, which refers to various instances. Thus, by contemplating the largest response from every instance present in the filtering stage, this step is carried out. Such a depiction builds a robust advantage and a larger response that can be implemented by identifying the conditions of instances in every class. This gives rise to building a classification model [9].

CNN contains a feature identifier and an automated learning method that abstracts characteristics from the data containing two layers: convolutional and pooling layers. Multi-layer perceptron is another component of CNN that incorporates features learned into the classification phase [11].

Deep Neural Network (DNN). With the advancements in calculating algorithms and techniques, a strong framework in supervised data has been instituted and is called "deep neural network (DNN)" [11]. DNN arose from SANN (shallow artificial neural networks), which is concerned with AI (artificial intelligence) [12]. Since the deep learning algorithm's stratified frameworks could establish nonlinear data in the set of layers, DNN implements a layered framework with intricate functions to handle the larger number of layers and the intricacies [11].

DNN is considered a highly popular tool for classification purposes since it handles intricate classifications exceptionally well [13]. DNN lacks training capabilities, which is considered one of the greatest problems confronting developers. Further, it attempts to decrease an objective function that contains a larger number of specifications in a multi-dimensional probing phase in the optimization issues. Hence, identifying an appropriate optimization algorithm for DNN to train them needed a higher degree of awareness. DNN has been built using the structure mutilated de-noising automatic encoder (SDAE) [14]. It also contains various cascaded automatic encoder layers and a soft-max classifier, where a cascaded automatic encoder utilizes coarse data to produce new features, and soft-max supports to classify features precisely. The above-stated characteristics are interdependent of one another, and it supports DNN to perform classification efficiently and is a primary function of DNN. However, another optimization technique, the gradient descent (GD) algorithm, has been implemented indefinite issues that have no intricate objective function specifically while DNN does training. Most importantly, in this operation, the number of objective specifications must be close to the optimal solutions [15]. DNN has been implemented as a prediction model satisfying the deep framework features as in [12].

Recurrent neural network (RNN). Analogous to neurons, recurrent neural networks contained nodes in a network and were advanced during the 1980s. Every node that looks similar to neurons was interlinked with one another. Nodes are further classified into classes of input neuron, hidden neuron, and output neuron. In this ternary operation, the data accept, reconstruct, and produce outcomes. Every neuron contains the characteristic of actually valued activation that varies over time, and each synapse is reasonable to actual value weights [16]. Neural networks do the classification process remarkably well in the learning process and approximations [17] and model an active system that contains nonlinear methods by employing the current data [18, 19]. As stated earlier, the RNN algorithm was developed inspired by the human brain and data obtained from ANN (artificial neural networks), but they vary moderately from one another. Research studies of RNN have led to the development of several processes: pattern recognition, image processing, robotics, and control associative memories, etc. [20]. Because of the feed-forward and feedback associations of RNN, it draws a diversified perspective from the previous data and utilizes it to adjust with instant modifications.

Similarly, the ability of RNN to make use of the data that varies over time in a repeated manner abridged the architecture of neural networks. The coherence and productive characteristics of RNN while handling real-world challenges will be much more effective [21]. RNN is capable of processing materialistic data in the case of the ranking approach. It is also capable of employing data abstraction in multi-layer to visualize its dynamic characteristics [22]. Additionally, RNN can establish signal connectivity at various levels that deliver outstanding processing power along with massive memory space [23].

Auto-Encoder (AE). The AE (Auto-encoder) is an artificial neural network (ANN) that can be utilized for unsupervised learning. It is called a Diabolo network or an auto-associator. Its main focus is on acquiring knowledge about encoding—commonly a depiction—to use in a data set. The depiction can further be employed for reducing dimensionality. Currently, auto-encoders have been broadly deployed in the learning of generative models. Essentially, the self-effacing structure of AE contains a homogeneous number of nodes in the input and output layers and acts as a feed-forward and non-recurrent neural network. Thus, the autoencoder is an unsupervised learning framework containing two parts: a decoder and an encoder [24].

Restricted Boltzman Machines (RBM). Another neural network is the Restricted Boltzman Machine (RBM) that contains two layers: a layer that has connected visible units and a single layer with hidden unconnected units. Furthermore, the link between these visible and hidden units is undirected, uniform, and symmetrical. There is a hidden bias within visible units inside the network. To represent these visible and hidden units, binary values, namely, 1 for visible and 0 for hidden units, are employed. Feature learning, classification, collaborative filtering, topic modeling, and dimensionality reduction are some of RBM applications [25].

Deep Belief Network (DBN). Another technique is a deep neural network for machine learning is the deep belief network (DBN). Several dormant variables and hidden layers surround its creative graphical model. Links exist between every layer, but there are no links between the units, which means within each layer, connections exist only between the layers but not between units. Based on probability, DBN rebuilds inputs while unsupervised learning is established. DBN algorithm further depends on supervised training to carry out classification after performing the unsupervised learning. Generally, DBNs are recognized as a group of purely unsupervised networks. For example, auto-encoders or RBM in which every hidden layer of the subsidiary network will behave as the visible layer for the upcoming ones [26].

18.3 Deep Learning Approaches for Social Media Analytics

Various social networks (e.g., Twitter and Facebook), through several users' recognition functions, allow them to apportion huge quantities of data, namely, images, videos, views, and speculations. Since deep learning had produced encouraging performances over NLP and visual data analysis, various deep learning methods such as clustering, classification, recommendation, and prediction were embraced. Here, we explain different ways of utilizing deep learning as a decisive method for solving several challenges in social media. The deep learning taxonomy in social media analytics is shown in Fig. 18.2.

18.3.1 Business Investigations

Since the inception of social media, social media networks, review forums, blogs, rankings, and proposals have been flourishing rapidly. An automatic process of differentiating between them is quite important for business organizations to sell their goods and to identify the latest anticipation in the market. However, automatically classifying the users' sentiments on a massive scale is difficult for social media [27, 28]. For example, evaluations derived from two dissimilar areas could carry non-identical vocabulary that gives rise to non-identical data delivery for various areas. Learning transitional depictions and adapting to a domain is inevitable, and it could rival a contingent role.

Fig. 18.2 The taxonomy of the deep learning methods in social media analytics

Classification Using DL. Social media environments help companies manage their relationships with their customers [29] and help them decide about eating out and for short trips [30]. Indeed, social media environments, namely, Twitter, Facebook, etc., become a comprehensive input source and immensely beneficial for government organizations, research organizations, and text mining entities. It induces entities to invest more in SM to promote their business. [29, 31].

Glorot et al. [32] proposed adapting the deep learning method to an area of sentiment classification. The transitional strategies present among the source and the destination information can be understood with deep learning methods. Because of the area's transition, such a movement, let's deep learning gather significant and useful information such as quality of customer service, product costs, customer reviews about products, and many more. Characteristics and attributes gathered at the early stage help understand them better at the next stage, thus enabling stage-wise learning. Additionally, Amazon utilizes deep learning methods and provides enhanced support to learn and depict the areas to promote business. In [32], authors, using Amazon's dataset, summarized various working areas and books, DVDs, electronics, kitchen, etc. The multi-layer perceptron and the stacked De-noising auto-encoder were compared in the feature extraction stage. While comparing, two different SDA-1 and a single layer and SDA-3 and 3 layers were employed.

However, the output of the MLP showed that nonlinear support for the extraction of information is not sufficient to gather all the information required from the data. Use an unsupervised method that can integrate data from various fields is more realistic. Naturally, one layer is not enough to achieve optimum performance on this wide-ranging problem. Three-layer stacking offers the best representation of data. The representation learned from SDAsh3 accurately depicts the domain shift required for different fields.

Ding et al. [33] implemented a CNN model to identify SM platform users and link them to a product requirement. Therefore, a commodity used by a particular customer would certainly be endorsed. Consumption means identifying the necessary products that are more likely to be bought by consumers. The proposed CNN-based model would categorize text terms better than SVM and word embedding or word-bagging.

Recommendation using DL. Despite SM's booming market, people today tend to buy clothes online. Lin et al. [34] suggested that a deep hierarchical CNN system is required to provide a better and more effective choice in online purchases. The authors used a wide range of image data from online shopping on Yahoo. The costume images show a high variation in roles and appearances with noisy backgrounds. The clothes-specific tree is designed for men and women, while subcategories include top, dress, suit, outfit, etc. The approach theory fits only those images of clothes that the consumer is likely to see in the dataset. Deep CNN is automatically learning discerning characteristics, capable of detecting heterogeneous images of cloths. The DL-based hierarchical search also provided an immediate retrieval response compared with the traditional, manually designed CNN.

Deep CNN models are prevalent for the representation of learning features. Kiapour et al. [35] proposed a DL model to align users' queries with the specific shopping venue. For this reason, large images from the fabric shop are used. The problem of matching the consumer's clothing demand with the probable shopping place is intuitively checked by measuring the graph similarities between demand features and photos from online shops. For consumers with better shopping places, shop reviews are graded according to the measured similarity. Therefore, Chen et al. [36] also proposed a DL-based approach to representing people based on their clothing choice.

18.3.2 User Behavior Analysis

Human behavior can be generally divided into individual actions and behavior in a group. Both categories have their reasons and consequences for themselves. Nevertheless, as consumers of society, human beings act differently in various social circumstances. Human activity is the product of such changes in the climate, environmental conditions, or human factors. To get to know society's well-being and the cultural changes, it is imperative to know people's social actions. Therefore, it is important to determine the effect of social factors on the actions of users. As previously established, the SM is a prominent source of social networking and mainly user-generated content. Therefore, the DL provides innovative methods to examine user activity and learning associations between past and present SM characteristics. Here we go through few categorized SM tasks to evaluate the actions of users using DL.

Classification using DL. SM is the most prominent interaction channel among people, which they use to create, share, and exchange ideas and knowledge. SM data are typically noisy, complex, of poor quality, large numbers, and heterogeneous in nature. Users of different backgrounds use SM channels to document daily activities. This makes the SM data subjective. This also offers a large set of attributes, such as the used tools, individuals' presence in a particular context, knowledge distribution, relation analysis, etc. For example, image annotation and classification are not trivial due to their diversity. However, SM encourages joint representation of the data created by multimodal users. For example, a flower image can be associated with several textual tags that indicate complex latent image classification learning. A joint group may help include content-related information.

Yuan et al. [37] proposed the Relational Generative Deep Believe Nets (RGDBN) model that investigates connections between information objects generated through latent interactions. RGDBN uses low-level representations at first and then higher-level representations of deeper architecture to understand the relations between the images and associated text marks. The dynamic and heterogeneous data space would better represent a deep model integrating latent characteristics' cumulative impact. Wang et al. [38] suggested a structural deep network embedding (SDNE) model captures complex networks' extremely no near structure effectively. It is a semi-controlled deep model with several nonlinear feature layers. SDNE's many deep layers allow the model to seize the heterogeneous nonlinear network structure. The dynamic representation of heterogeneous networks is achieved through network integration.

Using social networks, people exchange many different types of data. However, it is doubtful that users would share their personal details, such as age, birth year, ethnicity, etc. The prediction of user behavior includes classifying users based on their age ranges that can provide useful insights into user behavior across different age groups. Guimarãs et al. [39] evaluated 7000 social network sentences. Deep convolutional neural network (DCNN) is used to identify social network posts like a hashtag, retweet, tweet character, followers, comments, etc. Based on extensive experiments of various machine learning algorithms, for example, Random Forest, Decision Trees, SVM, etc., DCNN outperformed its counterparts regarding data classification on a large scale. Guimarãs et al. [39] have introduced an enhanced Sentiment Metric (eSM) to identify users with restricted personal information by age.

Clustering using DL. Community detection in SM data is a practical solution for identifying data objects' inherent grouping (defined in social media concepts). Common attributes may have common contributions for grouping knowledge objects. The meanings of attributes have pairs of importance and affect the grouping process. For example, individuals' academic qualification degrees may be used by community users from the same institutions or organizations.

In social networks, Zin et al. [40] incorporated classification into the Deep Learning Cluster Rank (DeepLCRank) ranking and introduced a new deep model. This approach further explains the description of social networking clusters. For each cluster element, a rank is assigned based on the knowledgeable characteristics of network information items. The different knowledge objects gathered from social networks, in turn, form several complex representations, which can be managed effectively by Deep Learning Cluster Rank.

Prediction using DL. Many studies use DL to predict human behavior in social networks. Aramo-Immonen et al. [41] analyzed community behavior through data collected from Twitter, a leading information dissemination tool. DL can efficiently process multi-dimensional data. A proposal by Zhang et al. [42] to use Tensor Auto-Encoder (TAE) as a deep computational model to learn heterogeneous patterns from YouTube tests. The tensor is used to define the linear vector relationship, given a vector basis. Arrays are one way to reflect memory tensors. Array dimensions make up tensor degree (rank). The TAE extends the regular DL model to represent input data in all layers using high-order tensor space. This model uses tensors to combine heterogeneous data features studied into the hidden layer, thus mapping the DL tensor model's multifaceted input data relationships. To implement the TAE model better, the author used a high-level backpropagation algorithm to improve prediction accuracy. However, compared to using homogeneous TAE data, training parameters took more time for heterogeneous data. SM data provide valuable information for notable predictions. Evidently, learning the heterogeneous source is still non-trivial in SM.

Jia et al. [43] considered integrating social networking environments and implementing an advanced, deep model that integrates social networking environments using deep learning techniques that fuse useful data from diverse social networks. This approach explicitly incorporates data derived from the profound learning model to learn various data sources' nature. The authors used multiple inner layers to understand the proposed model's detailed depictions from various social networking contexts. Their separate social media profiles initially linked users. Then the users were characterized by the derived multifarious features of semanticized, statistical, and physiological characteristics. Nevertheless, social media consumer practices were erratic, leading to data loss. The loss of data is deduced before the extracted features are given by NMF (Non-negative Matrix Factorization) in which NMF includes algorithms in an evaluation with multiple variants, in which M is a matrix factorized by two matric numbers, namely X, H discarding the negative attribute in the three matrices. The low-level features are mapped with deep layers to high-level features, which combine high-level features for the task. The user's trust and consistency within different social forums are calculated to obtain a detailed understanding of the user's interests, actions, and characteristics. Quora, About.me, and LinkedIn are the data sources used.

Social networks derive from online user relationships. They are graded as positive or negative, contributing to SSNs. Liu et al. [44, 45] proposed DBN to predict SSN connections. The considered tasks are friendship, co-authorship, confidence, mistrust, and other relations. Obviously, these days, social networks have increased communication rates in different crises. Lazreg et al. [46] proposed a successful SM analysis of posts like text and pictures to forecast user-communicated crises. The various SM posts are typically brief, informal, and heterogeneous (a mix of acronyms, languages, and misspellings). Without sacrificing generality, identifying the post's context is often necessary to determine its underlying significance. Also, posts on some ordinary accidents are used as data for additional training. When considering disasters, DL further understands these complex principles.

Liu and Zhu [47] have used micro-blog data that predicts users' actions by proposing an unaccounted drawing of the Vector Linguistic Representation (LRFV). This approach will explain users' semantic knowledge in a detailed and more analytical way.

Ranking using DL. People prefer to address any problems by posting inquiries on social media. Such forums are referred to as the CQA forums. This allows users to obtain adequate information. However, this does not guarantee that the users will be able to gain quick access to the necessary information due to a large number of answers for the same CQA questions. This also requires the answers given by experts to be ranked. Chen et al. [48] suggested a multi-instance DL frameworks approach to predict custom satisfaction for users. The authors presented a novel Multiple Instance Deep Learning (MIDL) model to predict customer satisfaction. There is a variety of responses to a single question in CQA. If a response is usually considered to be acceptable, a satisfied tag is given, so the answer is a satisfying response to a question. It needs, of course, that numerous responses are first learned.

Each response to a specific question in the CQA forum is considered to occur in a bag in which each question gets only one suitable answer in the Stack Exchange dataset. Considering users' conventional behavior, a general user space can be described and implemented to represent each individual. When the functional extraction occurs, all will be injected into the deep recurrent NN (neural networks) to decide if they are negative or positive.

Recommendation using DL. SM acts as a promising source of multi-domain user content data that are constantly recommended for appropriate users. If things from various domains are studied together and recommended, this recommendation's impact may be enhanced.

Elkahky et al. [49] suggested a deep neural network of multi-views (MV-DNN), mapping articles and users to a semantic space shared and suggesting articles with the capitalized resemblance. Many data tools were used, including Microsoft software logs for Bing search logs and the Windows Store Download History Log and Xbox film view-logs. DNN is used to combine user space and elements from various domains with lower-dimensional applications. Members of social networks also belong to various realms and thus often search for objects of interest. MV-DNN will suggest articles focused on specific characteristics like the film type, category of application, country, region to which the item belongs.

Collaborative filters (CF) are also a common way of informing users about related content. CF-based approaches typically use the ratings of users to suggest products. However, the accuracy of the recommendation can be substantially diminished by inadequate ratings.

Wang et al. [50] proposed the collaborative deep learning (CDL) model to research the awareness of deep content information representations and collaborative filtering to assess users. They used several fields like Citeu-Like, Netflix, and IMDB to suggest products for consumers in this context. Therefore, consumer morale needs to be taken into consideration when making relevant recommendations. Deng et al. [51] proposed the matrix factorization (DLMF) model focused on deep learning to synthesize users' preferences and trusting ties. DLMF has done well with rare data and cold start users in terms of recommendation accuracy. The authors use Epinions data to learn in the first step the autonomic encoder of the users and the latent functional vectors are learned in the second step. This approach helps to identify trustworthy groups.

18.3.3 Sentiment Analysis

The sentiment analysis is often alluded to by using Facebook, Twitter, etc., to forecast users' attitudes producing large textual content. The study of opinion mining. It should be noted that the aim of evaluating consumers' feelings is to convince them. Therefore, it is important to group their responses and actions in neutral, positive, or negative or some other grouping about a certain subject or commodity. Customer survey results, consumer reviews, user perceptions, and areas such as education, industry, e-commerce, and healthcare are widely used. In this section, we discuss the techniques of prediction, classification, and rating of feelings about the study of feelings.

Classification using DL. DL has been a productive way to deal with issues related to the classification of emotions in the present scenario. A neural network innately and automatically acquires a useful picture without human intervention except during the labeling phase. Nevertheless, DL's effectiveness depends largely on large-scale training data being available. Web 2.0, like e-commerce, allowed people to use large numbers of social networks and post their views and views on their acquisition of information on social media or the seller's website. Such partial knowledge is a valuable tool for vendors to improve their products, increase service quality, and provide sufficient resolution for potential customers.

Poria et al. [52] proposed a systematic approach to extract short text characteristics. The activation values of the deep CNN are based on inner layers. The authors use CNN for the extraction of textual data. The usage of Google Translator, however, is translated from Spanish to English. The CNN consists of seven layers equipped with conventional context propagation typically relaxed to boost the model's precision. The data set consists of 498 short video fragments of a word spoken by the person. The objects are manually labeled as neutral, positive, or negative for the polarity of feelings. However, a total of 447 items are recycled through the disposal of neutral objects. A vector-based on several kernel-learning (MKL) algorithms is used to construct a classifier, a well-recognized heterogeneous data processor in the vision, audio, and textual modes. The authors actually merged task and decision-making outcomes. Therefore, features of a feature-level fusion are entered into a supervised classifier called SVM, following which the extracted features are fed into the individual classifier in decision-making fusion, and decisions are then combined. The importance and reliance on handcrafted features of the CNN app extractor are automatic.

In particular, in a controlled manner, CNN adapts well to the particular data set's uniqueness. The aspect mining role includes identifying aims of opinion found in the prejudiced texts, especially if the holder of the judgment endorses or argues about the basic features of a product or service. The new aspect-driven opinion mining approach used by Poria et al. [53] is being suggested using a deep 7-layered CNN. There are limitations on conventional methods for extracting text features such as CRFs. Such limitations include many features required for improved performance. The language pattern (LP), which relies on the expression's grammatical precision, must be generated manually. Because of the app's automatic extraction design, the CNN system could effectively overcome these extraction limitations. The aspect-term characteristics are based on the words of his neighbor. A 5-word window on all words in one sentence, particularly on plus or minus words, is used for aspect extraction. The local window features are called medium word characteristics. Next, CNN is served with the function vector. For word embedding, the CBOW model is used. Google and Amazon embedding is especially used in electronics (e.g., cellular telephones, computer computers) or food chains (e.g., fast stores, restaurants), where positive LP rules are applied [53]. However, there are fewer aspect-driven words for C BOW than for the electronically regulated domain.

A Nov54.el framework was also proposed by Guan et al. [54]. The system is known as Deep Embedding (WDE), supervised Weakly to distinguish customer feedback. The goal was to concentrate on every phrase semantically. In this context, a high-level representation is first learned that shows the overall distribution of sentences by rating information. First, on the top end of the embedding layer, a classification layer for supervised finishing with the marking of phrases is added. The application of sentiment classification analysis ratings [54] is a first step in the culture of sentiment analyses. A lot of unlisted data are educated with the use of RBM/Autoencoders. Because ratings are rushing marks, a classifier may readily be misled. Therefore, a five-star rating scale is adopted. Amazon's customer reviews are used as data collection for three areas: digital cameras, laptops, and cells. Three-star ratings are dismissed in this situation. The WDE approach proposed is compared with other methods of baseline. Nevertheless, the WDE surpassed the current methods concerning precision. In short, WDE has actually been more successful in training the DNN by exploiting rating information available on social websites.

Araque et al. [55] proposed to present the value of multi-source information as a Classifier Ensemble Model (CEM). It is worth noting that, compared with its basic components, this composition has more detail. The model aimed at improving overall classification efficiency by combining surface (traditional ML classification) and deep (DL-based) features that were not accomplished using classifiers. Seven different public databases from micro-blogging areas and film reviews were used.

Occasionally, users post nonsensical content using SM sites like Twitter. It may be labeled as hate or offensive speech, targeting individuals, including celebrities, politicians, or goods. Identifying the hatred behavior of individuals from a particular community or some other community is important. Furthermore, it is imperative to advise individuals about acceptable content. Various deep learning frameworks have been deployed by Badjatiya et al. [56], such as FasText, various CNN approaches, LSTM (Long/Short Term Memory Networks) to categorize tweets and identify whether the speech is causing hatred through tags such as xenophobic, bigoted or something else. In practice, to detect the purpose of hatred, convolutional neural networks were employed, whereas FasText aims to constitute a document in the structure of word vectors to adjust the word depictions through initially trained word implants received from the GloVe implants [57]. LSMs were deployed to trace well-established contents present in the tweets. The ability of DL to detect hateful content in tweets is its major benefit [56].

Pitsilis et al. [58] implemented a deep framework in which short texts from tweets are marked as hate speech. This method, however, does not necessarily involve pre-trained word embedding. Users prefer to post hateful content through offensive messages. Users often tend to type brief phrases or slang words to express racist intentions. Therefore, tweets containing a minimum of 30 words are used in the deeply proposed model preparation. The reason for this is, word vector frequency is higher than pertained word embedding.

A multi-layered CNN (MLCNN) proposal was made by Alali et al. [59] to segment tweets into five levels: strongly optimistic, optimistic, negative, neutral, and moderately negative. After an empirical review of the proposed model, the authors found the three-layered CNN combinations the best possible solution than other combinations.

Prediction using DL. Prediction of SM data opinions is a typical practice. English was commonly used for the activities of opinion mining. Nevertheless, Li et al. [60] endorsed Chinese opinion labels. The researchers collected 2270 film reviews. The comments were then evaluated using various parameters, such as using rude language with special symbols, using several paragraphs, typos, short or long sentences, or multi-language use. Finally, a Chinese sentiment dataset was developed, named Chinese Sentiment Treebank. Emotions are classified into five classes: highly optimistic, supportive, neutral, negative, and strongly negative. The recursive DL model was proposed to predict labels identifying these classified feelings. Compared with three regular ones, Naïve Bayes (NB), Maximum Entropy (ME), and SVM, the RNDM forecasts all basic guidance because it created recommended sentiment labels for contrasting structural conjoint sentences, such because "X but Y."

DL techniques have evolved rapidly and have achieved tremendous success in deep and complex data training models in a recent development to resolve a wide variety of NLP and text mining tasks. People's text already includes morphological aspects such as grammatical rules, syntactic information, part-of-speech Marking, and semanticizing specifics such as the relationship between words and persons, synonyms, and antonyms.

However, DL increases the use of information for fine word embedding [61]. In a detailed analysis, Bian et al. [62] have shown that information DL is incorporated into text embedding. Nonetheless, Stojanovski et al. [57] proposed to use pre-trained GloVe embedded words [63] for Twitter sentiment analysis.

To ascertain the grammatical meaning from the texts, Bian et al. [62] employed sound of speech, source, and connections. Grammatical and syntax-based understandings were employed as supplementary inputs to improve the depiction of the words. A ground line technique called CBOW (Continuous Bag-Of-Words) method is deployed here. However, WordNet, Longman, Freebase, and Morfessor were made to use datasets to assess the quality of word implants learned with integrated understanding and no understanding. This work also explored three tasks, namely sentence ending, word comparison, and perception of similarity. The DL framework created embeddings by aggregating the morphological elements for each root/affix and syllable in a comparison. Essentially, they are well equipped to provide both semantic and syntactical learning of the DL-based layered nature models. However, the authors concluded that syntactic awareness offers important input information but is not appropriate for regularized goals. Semantic awareness can, however, extend the completion of the sentence and the word similarity function. Furthermore, the use of semantical information as supplementary feedback is beneficial for analogical reasoning.

Since Twitter places limits on the right of users to express themselves and feelings within 140 tweets, the task has become complicated. Stojanovski et al. [57] also removed both URLs and HTML entries from the tweet for cleaning redundant tweet information. The pre-trained word embedding is used to build lookup tables, each linked to its corresponding feature representation. The authors fused two NN models about the DL, one of which is the CNN used to retrieve tweets, and the other is the Gates Neural Network GRNN, which uses sequential data where the input depends on the previous output. There are some impressive properties and reasons for using a Gated Recurrent Unit (GRU): it uses less parameter, allows less data distribution, and allows fast learning. The GRU architecture provides gating systems for tracking information flow within the systems below. CNN and GRNN, therefore, dominated the production of existing NN models.

Ranking using DL. CQA forums handle multiple user queries and strive almost hourly to receive the best answers from the experts. Because furthermore, experts' answers to the same question are multiplied, CQA provides a connection between pairs of questions and answers. Likewise, the connections between query documents exist even as short text pairs in many information retrieval tasks. This means that the questions and query-document pairs have to be graded. Also, related relations between question records occur as brief text sets in assignments involving information retrieval. Moreover, function development is an important part of these pairs' rankings. Later on, CNNs were very successful in mapping resourceful learning and input sentences in low-dimensional vector spaces. A semantic and syntactic characteristic of the input phrase is essential. It is commonly used in contemporary text processing tasks.

Severyn and Moschitti [64] suggested a CNN-based technique called ConvNets to rate short text sets quickly. This approach helps to obtain strong records and queries from intermediaries. It was then used to determine their language match. The network consists of a diversified convolution layer followed by an unusual and simple maximum combination to minimize dimensions. The raw words are used as network data. This data must be represented in real-life vectors. Such real-world function vectors are then processed using the successive network layers. However, the convolutional layer's objective is to extract significant patterns, particularly discriminatory word sequences formed in common training input sentence cases. The authors used two famous TREC measures, namely, the TREC micro blog's collection and reaction phrase. CNN sponsored more intermediate interpretations, increasing the understanding of high-quality sentencing models. The architecture requires an intermediate representation of the questions/responses that generate a much richer representation together. ConvNets also does not need manual software development, virtual preprocessing, and peripheral tools that may be costly or not usable. The same type of architecture also applies to other areas.

Tan et al. [65] proposed a biLSTM-based approach for selecting the correct answer on an SM platform from a resource pool of answers for a particular question. In this analysis, TREC QA question–answer selection dataset is used. The authors used deep representations in open domain question-response structures to fit correct answers based on their semantic structure. This dilemma is overcome by knowing the greater cosine resemblance to the question, and the answer is selected from the lower list with the highest cosine resemblance. The biLSTM, a DL-based model, played a key role where a problem had several relationships with the terms or words used in answers and other relationships and ideas. The notable characteristic of the proposed model is that it does not rely on the engineering of linguistic features and methods used to apply any domain.

18.3.4 Anomaly Detection

The identification of anomalies in the field of the identification of data irregularities. Throughout the analysis of real-life data sets, such instances must be found in data separated by other data instances [66]. These unusual instances of data are termed as outliers, anomalies, or abnormalities. Hawkins [67] described the outlier as "An observation which deviates so much from other observations as to arouse suspicion that a different mechanism generated it." Anomalies are typically generated by glitches in the underlying data, although anomalies are sometimes also generated by previously unknown fundamental processes. Many attempts have been made using DL to detect anomalies. Here we address some important anomaly detection techniques in SM based on DL.

Classification using DL. Supervised identification of irregularities may have a critical role in enhancing safety systems. It can also allow security and law enforcement agencies to track disruptive and malicious cyberspace acts and conversations proactively. Ebrahimi et al. [68] introduced a CNN classifier to detect such behavior in large-volume logs of chats efficiently. The authors used PAN-201222 for research. As a binary classifier, CNNs are used to perform the function of classification of conversational text. The convolutional layer typically operates in multiple input regions in CNN. The pooling layer is therefore used to check for higher abstraction levels in every convolutional layer. Max pooling is used as a text classification function because it improves other average pooling methods [69]. The use of two convolutional layers is less successful than using one layer in the text classification task.

Nonetheless, a lack of convolutional layers may also help with the image classification tasks as the suggested model contained over-fit data with more convolutional layers.

Ribeiro et al. [70] suggested a deep CAE detect anomalies in the video data. The researchers used videos open to the public, including SM channels. Since all training instances come under the non-anomalous category, the proposed CAE model does not require labeled data. Video frames from video clips are subsampled with sliding windows, which removes both motion and appearance. The anomaly value is the frame reconstruction error. The Under Curve Area (AUC) is used for the efficiency estimation of the proposed model. This has been found to improve the efficiency of the proposed classifier, CAE, by collecting high-level information on unprocessed data. This is especially useful to complement high-level knowledge, as the types of anomalies identified beforehand can be detected.

Clustering using DL. It is essential to detect anomalous events, especially in videos. Since the video consists of many frames, it is considered an overlapping issue to detect irregular scenes from video data. Function learning in video surveillance is non-trivial. Nevertheless, Xu et al. [71] suggested a DNN-based Appearance and Motion DeepNet (ADMN) to automatically and effectively learn feature representations. A double fusion system incorporates elements from early and late fusion approaches. In the early fusion technique, the stacked de-noising self-encoder is provided to learn gestures and behaviors discreetly. SVM is then used for the estimation of anomaly results. Eventually, the anomaly scores are combined to detect irregular events. The proposed ADMN model is found to be based on previous knowledge for the study of feature representation. This is also more effective than the current manufactured learning of video apps. The downside to this method involves increased processing costs and numerous variations in the images.

Nonetheless, phenomena are caused by multiple occurrences or by various causes that render instances known as anomalies. One of the drawbacks of [71] is the lack of understanding of video patterns' multiple occurrences. There have been co-evolutionary differences in Hayat and Daud [72] in heterogeneous bibliographical data networks. A set of related attributes (attribute object) is correlated with the anomaly co-occurring (target object). Each attribute object's effect on a target object is measured, which led to the identifying anomalies cause in underlying data.

Feng et al. [73] found anomalies in crowded scenes. They suggested a deep Gaussian Mixture (GMM) model with different layer combinations. The motion and appearance characteristics are extracted and clustered by principle component analytics (PCA). Clusters that contain fewer members in comparison to normal classes are considered anomalies. Deep-GMM is very useful in comparison with handmade function learning. It is found; however, short and long-term time-motion characteristics also have scope to explore.

18.4 Social Media Analytics Tools in the Market

The following are the leading openly available and paid tools in the social media analytics available for merchants along with the other analytics toolkits owned by social networks [74].

18.4.1 Notable Free and Paid Social Media Analytics Tools

These tools are either paid with a free trial or free with limited features.

- **Audiense—**Audiense Connect presents one of Twitter's most detailed analyses. Information about people following you, peers, place, language, preferences, and influence of competitors can be gathered. Audience uses deep learning on social data to help you understand the target audience by providing affinities and demographics information. You can actively reach out to the audience.
- **Buffer Analyse—**Buffer Analyse social media research tool is developed for online brands who want to better assess their online business strategy and keep track of the performance without getting frustrated.
- **Cyfe—**Cyfe, a design dashboard platform that fills hundreds of marketing tools with integrated metrics. With its Social Media component, you can synchronize all major networks and collect summary reports for your monitored and traceable profiles. It features 50 widgets for tracing scope, prints, taps, inspection, and updates, on Facebook alone.
- **Followerwonk—**Followerwonk, a Twitter tool, shows your supporters and behavior comprehensive breakdowns. You can view folder details from the Analytics tab. You are also provided statistics such as when you post a tweet, how many of your followers are online, and how their followers fall into categories such as social authority, operation, total tweets, and count of supporters.
- **Google Analytics—**Although Google Analytics is designed to analyze website traffic in great detail, it is beneficial to analyze the effect of social media as a Marketing Medium and a Traffic Source. You can test the number of visits to your site from every major social network by going through the Acquisition > Social > Summary. With the aid of goals and conversions, you can relate this to your end.
- **Hootsuite—**Hootsuite is a perfect mix of marketing, listening, publishing, and analytical methods in social media. It allows you to decide what content works for you, expand your posts' scope, boost your ads' efficacy, and work in collaboration with your entire social media team. It has extensive analysis features that allow you to understand your success on all social media better.

- **Keyhole—**Keyhole provides numerous trackers that help you control your activities, promotions, and interaction with influencers, brand names, and social talks in the industry. It helps you monitor the hashtags, keywords, and accounts that provide an abundance of information for your brand or competitors, such as outreach, top post, voice share, and more.
- **Likealyzer—**Likealyzer is a tool by a media intelligence company, Meltwater, to evaluate Facebook profiles. It scans your Facebook page and returns metrics such as the number of posts per day, tweets, schedules, and tweets length. Likealyzer tests the efficiency of the Website based on over 70 data points and provides improvement recommendations based on the findings.
- **Quintly—**Quintly is an analytics platform at the enterprise level that tracks the interaction across all major social networks. Thousands of flexible metrics can be mixed, and matched and interactive collaborative dashboards can be deployed. You may compare several pages that belong to you or your competitors with or against pre-set benchmarks.
- **Rival IQ—**Rival IQ can monitor and compare various companies through social networks and SEO. It also analyses Facebook and Instagram advertising and provides historical data for 24 months. You will understand and learn from their strategies the overall digital methods that work for your competitors.
- **Socialbakers—**Socialbakers is a series of analytical tools that integrates all critical resources, such as multiple profile monitoring, main performance metrics, competitive intelligence, and automated reporting. In addition to analytical tools, Socialbakers provides resources to publish and optimize the content, identify influencers, digital customer mapping, and social media exposure.
- **SparkToro—**SparkToro is an intelligence platform developed by Moz Rand Fishkin's former wizard. This will finally be a search engine to find blogs, magazines, podcasts, and social media. It is a series of advanced tools currently evaluating your Twitter handle and followers and offering an unbiased evaluation of your brand presence and expertise on Twitter.
- **Sprout Social—**Sprout Social is robust management and tracking tool for social media, offering a range of solutions at the corporate level. The social listening apps on a cross-channel include a quantitative and qualitative study of subjects, hashtags, and keywords. From its visual and intuitive studies, one can understand how the content performs on various social networks.
- **Tailwind—**Tailwind is a marketing solution from Pinterest and Instagram that allows you to schedule specific content, track talks, and evaluate results. For Pinterest, the research provides you with an insight into the domain's number of posts, the possible interactions, and the number of followers. This displays simple metrics such as followers, updates, comments, and likes for Instagram.

- **Union Metrics—**Union Metrics, a TweetReach intelligence tool kit, offers a free snapshot of the user's Twitter analytic data. Useful charts showing the scope, visibility, operation, and contributor based on a username, keyword, or hashtag. This is pretty useful for reviewing mentions, monitoring branded hashtags, and analyzing keywords from the industry.

18.4.2 Social Media Analytics Dashboards from Social Media Networks

Each major social media network has its own integrated analysis and dashboards that provide you with a clear insight into your network operation. Here is how one can extract metrics data.

- **Facebook Insights—**Facebook Insights is open to all page administrators to gather details regarding your posts, fans, and scope. You can also set up a list of Pages to Watch from the Insights tab, which gives information on other Facebook pages' performance.
- **Instagram Insights—**If you have an Instagram business profile, the native in-app analytics of Instagram will allow you to gain Instagram Insights. It provides a wide variety of information on your profile, stories, updates, and advertisements. It also offers specific information on the number of followers, their most popular days, and preferred time.
- **Twitter Analytics—**Twitter offers a 28-day summary of your profile, which shows how it performs in terms of profile updates, the number of new followers, and tweet views. It also provides an overview of each tweet's feedback and information on their contribution, e.g., retweets, mentions, likes, and clicks. The acquired data can be exported as a report for further analysis.
- **Pinterest Analytics—**Pinterest itself is one of the main examples of Pinterest analytics. The dashboard reveals growth in perceptions and followers, public information, and platform involvement. With each of these, you can click through more detailed reports to see which posts to boards have done the best.
- **LinkedIn Analytics—**You can track your progress on a page of your business or organization and display different details on your followers and guests. If you have linked LinkedIn Career Pages to your profile, you can also access the Talent Brand statistics.
- **YouTube Analytics—**YouTube Analytics is an automated system to monitor your channel's video performance. It reports an extensive range of data, including traffic forecasts, time tracking, views, sales, ad efficiency, and audience and subscribers' retention.

18.5 Case Study

In this case study, we investigate how Facebook benefits from using deep learning.

18.5.1 How Facebook Uses Deep Learning

Close to 2 billion users updated 2,93,000 times per minute, Facebook managed to extract profit from a tiny section of unstructured data, i.e., information that cannot be quantified and placed in computer-analytical lines columns. However, now, profound thinking leads to this [75].

Profound learning methods require computers to learn to identify data alone. A profound image recognition method, for example, can learn to recognize pictures that include cats without knowing how a cat looks like. Through examining the repository of images, the image data can be extracted for better understanding.

- What other information can be found in this picture of a cat?
- Which text or metadata might indicate a cat contained in an image?

It helps to offer an unstructured data structure, making deep learning a beneficial resource for companies like Facebook. An example of how deeply learned social networks will help users understand more, deep neural networks—the cornerstones of deep learning—determine which advertisements to display to users. It has always been the core aspect of Facebook's work, so it wants to retain an advantage over other high-tech competitors, like Google, who are vying for the dominance of the same advertising market as it finds out how much they can about consumers and groups them together in the most informative way possible to serve them ads.

Technicalities. Most deep learning applications on Facebook have been developed on the Torch platform—a deep learning technology and neural network development framework.

Ideas and insights. In the potential growth of Facebook, deep learning plays a key role. It is also possible that several other companies will profit—after all, Facebook gives access to its user details to other companies. Therefore, even small businesses without their own data infrastructure may benefit from the most advanced profound learning technologies.

18.5.2 How Google Uses Deep Learning

The primary application of Google's deep learning is to deliver better video recommendations on YouTube by observing audiences' behaviors and expectations while streaming videos and improving how they will continue to function [76]. Google also knew from the data that audiences could watch the next videos to keep them entertained and keep the advertising dollars rolling. Google Brain is working on this technology.

Technicalities. Artificial neural networks are built-in deep learning that simulates how a human brain handles and processes information. The term "deep" refers to using multiple layers of neural networks stacked over one another. This method of data analysis is called a "deep neural network." A deep neural network's sophistication means that it provides more accurate processed information than the other AI systems preceding this network.

Ideas and insights. Google is a leader in deep learning, and it has definitely contributed to a much broader audience with the technology. There are many high expectations for deep learning, and in the coming years, almost every business and company will make a major leap forward. Google's epic groundwork will play a significant role in the future.

18.5.3 How Twitter Uses Deep Learning

Twitter is a social media site where microblogs (shares 280-character updates) are exchanged amongst 328 million monthly active users. It is a cross between immediate and blogging messages—or social messages, but it was also vital to coverage, promotion of events, marketing, and industry. Twitter has become the world's ninth-largest social network, and Cortex, the on-site development department from Twitter, has switched to artificial intelligence (AI) technology to enhance the app's user experience.

One way for Twitter to use artificial intelligence is to evaluate the tweet suggestions on users' schedules to find the most appropriate tweets for each individual [77]. The algorithms search thousands of tweets per second to identify and categorize them according to each user feed.

Technicalities. Twitter's Ranking Algorithm involves many data, manages it through deep neural networks, and over time has learned which information is important to each user. All tweets are labeled with a rating model that defines the probability of a consumer having a value for feed. The classification model considers the content of the tweet itself, including the number of retweets or likes it has been obtained accompanied by an image or video; the author of the tweet to see if you have had previous experiences with the author and the strength of your connection with the author; the type and lot of tweets which you liked in the previous and how this tweet resumes. The higher the relevance rating, the higher the tweet on your list and the more likely the tweet appears in the section "In case you missed it."

Cortex has used deep learning to train his algorithm to understand what happens with a live stream. A big neural network has been trained to recognize video content from many sources. People looked at the videos and labeled them to describe what they saw using certain keywords. Therefore, a dog's video was labeled with the keyword dog and cats, canines, mammals, and more. Such data were used to train the algorithm so that material could be recognized in the video.

Ideas and insights. Artificial intelligence-enabled technologies can improve or enhance the goods and services like Twitter's team shows. Human communication and artificial intelligence are instrumental in stressing each one's best attributes.

18.6 Conclusion

Recently, SMA received widespread recognition. Deep learning is a recent and hot topic in machine learning defined as a cascade of nonlinear processing layers to learn the representation of multiple data levels. Machine learning experts have been seeking to identify patterns and representations of raw data for decades. This approach is also called representation learning. In comparison to conventional machine-learning and data-mining techniques, profound information can generate high-level data representations from large raw data volumes. It, therefore, provides a solution for multiple applications in the real world. This chapter describes DL's state-of-the-art work achievements for SM analytics. DL's SM systems pose a variety of notable challenges. We include detailed definitions of different SM domains. DL-based approaches have considerable power to learn useful data representations from multi-domain SM systems, such as consumer market analysis, behavioral analysis, sentiment analysis, identification of anomalies, and more. In conclusion, deep learning, a modern and fast-growing process, offers numerous challenges and opportunities, and solutions to various SM domains.

References

1. What is Social Media? https://www.thebalancesmb.com/what-is-social-media-2890301. Last accessed 01 March 2020
2. Types of Social Media and How Each Can Benefit Your Business. https://blog.hootsuite.com/types-of-social-media/. Last accessed 2020/03/04
3. Social Media. https://whatis.techtarget.com/definition/social-media. Last accessed 07 March 2020
4. Important Reasons why you should Use Social Media Analytics. https://iag.me/socialmedia/6-important-reasons-why-you-should-use-social-media-analytics/. Last accessed 13 March 2020
5. What Is Social Media Analytics & Why Is It Important? https://www.netbase.com/blog/what-is-social-media-analytics-why-is-it-important/. Last accessed 15 March 2020
6. Wu, J.-L., Chang, P.-C., Tsao, C.-C., Fan, C.-Y.: A patent quality analysis and classification system using self-organizing maps with support vector machine. Appl. Soft Comput. **41**, 305–316 (2016)
7. Zhang, L., Zhang, Q.: A novel ant-based clustering algorithm using the kernel method. Inf. Sci. **181**, 4658–4672 (2011)
8. Martí, R., Martínez-Gavara, A., Sánchez-Oro, J., Duarte, A.: Tabu search for the dynamic Bipartite Drawing Problem. Comput. Oper. Res. **91**, 1–12 (2018)
9. Zhang, X.M., Han, Q.L., Ge, X., Ding, D.: An overview of recent developments in Lyapunov-Krasovskii functionals and stability criteria for recurrent neural networks with time-varying delays. Neurocomputing **313**, 392–401 (2018)

10. Zhu, S., Shen, Y.: Robustness analysis for connection weight matrix of global exponential stability recurrent neural networks. Neurocomputing **101**, 370–374 (2013)
11. Acharya, U.R., Fujita, H., Oh, S.L., Adam, M., Tan, J.H., Chua, C.K.: Automated detection of coronary artery disease using different durations of ECG segments with convolutional neural network. Knowledge-Based Syst. **132**, 62–71 (2017)
12. Hüllermeier, E.: Does machine learning need fuzzy logic? Fuzzy Sets Syst. **281**, 292–299 (2015)
13. Nguyen, T., Li, Z., Spiegler, V., Ieromonachou, P., Lin, Y.: Big data analytics in supply chain management: a state-of-the-art literature review. Comput. Oper. Res. **98**, 254–264 (2018)
14. Wang, Y., Kung, L.A., Wang, W.Y.C., Cegielski, C.G.: Integrated big data analytics-enabled transformation model: application to health care. Inf. Manage. **55**(1), 64–79 (2018)
15. Emin Yuksel, M., Sarikaya Basturk, N., Badem, H., Caliskan, A., Basturk, A.: Classification of high resolution hyperspectral remote sensing data using deep neural networks. J. Intell. Fuzzy Syst. **34**, 2273–2285 (2018)
16. Taherkhani, M., Safabakhsh, R.: A novel stability-based adaptive inertia weight for particle swarm optimization. Appl Soft Comput **38**, 281–295 (2016)
17. Yang, F., Dong, H., Wang, Z., Ren, W., Alsaadi, F.E.: A new approach to non-fragile state estimation for continuous neural networks with time-delays. Neurocomputing **197**, 205–211 (2016)
18. Huang, J., Yu, Z.L., Gu, Z.: A clustering method based on extreme learning machine. Neurocomputing **227**, 108–119 (2018)
19. Panda, M., Abraham, A.: Hybrid evolutionary algorithms for classification data mining. Neural Comput. Appl. **26**(3), 507–523 (2014)
20. Wan, Y., Song, S., Huang, G., Li, S.: Twin extreme learning machines for pattern classification. Neurocomputing **260**, 235–244 (2017)
21. Loebbecke, C., Picot, A.: Reflections on societal and business model transformation arising from digitization and big data analytics: a research agenda. J. Strat. Inf. Syst. **24**(3), 49–157 (2015)
22. Esposito, C., Ficco, M., Palmieri, F., Castiglione, A.: A knowledge-based platform for Big Data analytics based on publish/subscribe services and stream processing. Knowledge-Based Syst. **79**, 3–17 (2015)
23. Miao, Z., Wang, Y., Yang, Y.: Robust tracking control of uncertain dynamic nonholonomic systems using recurrent neural networks. Neurocomputing **42**, 216–227 (2014)
24. Bengio, Y.: Learning deep architectures for AI. Found Trends Mach. Learn. **2**(1), 1–127 (2009)
25. Larochelle, H., Bengio, Y.: Classification using discriminative restricted Boltzmann machines. In: Proceedings of the 25th International Conference on Machine Learning. Association for Computing Machinery, pp. 536–543. New York, NY, USA (2008)
26. Hinton, G.E.: Deep belief networks. Scholarpedia **4**(5), 5947 (2009)
27. Dean, J., Corrado, G.S., Monga, R., Chen, K., Devin, M., Le, Q.V., Mao, M.Z., Ranzato, M., Senior, A.W., Tucker, P.A., Yang, K., Ng, A.Y.: Large scale distributed deep networks. In: Proceedings of Advances in Neural Information Processing Systems, pp. 1223–1231. Curran Associates Inc. Red Hook, NY, USA (2012)
28. Adam, C., Brody, H., Tao, W., Wu, David, J., Ng, Andrew, Y., Bryan, C.: Deep learning with COTS HPC systems. In: Proceedings of the 30 th International Conference on Machine Learning, vol. 28, pp. 1337–1345. JMLR.org, Atlanta, GA, USA (2013)
29. Piller, F., Vossen, A., Ihl, C.: From social media to social product development: the impact of social media on co-creation of innovation. Die Unternehmung **65**(1), 1–22 (2011)
30. McCarthy, L., Stock, D., Verma, R.: How travelers use online and social media channels to make hotel-choice decisions [electronic article]. Cornell Hospitality Rep. **10**(18), 6–18 (2010)
31. Weinberg, B.D., Pehlivan, E.: Social spending: managing the social media mix. Bus Horiz. **54**(3), 275–282 (2011)
32. Glorot, X., Bordes, A., Bengio, Y.: Domain adaptation for largescale sentiment classification: a deep learning approach. In: Proceedings of the 28th International Conference on Machine Learning, ICML, pp. 513–520. Bellevue, WA, USA (2011).

33. Ding, X., Liu, T., Duan, J., Nie, J.-Y.: Mining user consumption intention from social media using domain adaptive convolutional neural network. In: Proceedings of the Twenty-Ninth AAAI Conference on Artificial Intelligence, pp. 2389–2395. AAAI Press, Austin, Texas (2015)
34. Lin, K., Yang, H.-F., Liu, K.-H., Hsiao, J.-H., Chen, C.-S.: Rapid clothing retrieval via deep learning of binary codes and hierarchical search. In: Proceedings of the 5th ACM on International Conference on Multimedia Retrieval. Association for Computing Machinery, pp. 499–502, New York, NY, US (2015)
35. Kiapour, M.H., Han, X., Lazebnik, S., Berg, A.C., Berg, T.L.: Where to buy it: matching street clothing photos in online shops. In: Proceedings of the IEEE International Conference on Computer Vision (ICCV). IEEE, Santiago, Chile, pp. 3343–3351 (2015)
36. Chen, Q., Huang, J., Feris, R., Brown, L.M., Dong, J., Yan, S.: Deep domain adaptation for describing people based on fine-grained clothing attributes. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), pp. 5315–5324. IEEE, Boston, MA, USA (2015)
37. Yuan, Z., Sang, J., Liu, Y., Xu, C.: Latent feature learning in social media network. In: Proceedings of the 21st ACM International Conference on Multimedia. Association for Computing Machinery, pp. 253–262. New York, NY, USA (2013)
38. Wang, D., Cui, P., Zhu, W.: Structural deep network embedding. In: Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining. Association for Computing Machinery, New York, NY, USA, pp. 1225–1234 (2016)
39. Guimarãs, R.G., Rosa, R.L., De Gaetano, D., Rodríguez, D.Z., Bressan, G.: Age groups classification in social network using deep learning. IEEE Access **5**, 10805–10816 (2017)
40. Zin, T.T., Tin, P., Hama, H.: Deep learning model for integration of clustering with ranking in social networks. In: Proceedings of the International Conference of Genetic Evolutionary Computation, pp. 247–254. Springer International Publishing (2017)
41. Aramo-Immonen, H., Jussila, J., Huhtamäki, J.: Exploring co-learning behavior of conference participants with visual network analysis of Twitter data. Comput. Hum. Behav. **51**, 1154–1162 (2015)
42. Zhang, Q., Yang, L.T., Chen, Z.: Deep computation model for unsupervised feature learning on big data. IEEE Trans. Serv. Comput. **9**(1), 161–171 (2016)
43. Jia, Y., Song, X., Zhou, J., Liu, L., Nie, L., Rosenblum, D.S.: Fusing social networks with deep learning for volunteerism tendency prediction. In: Proceedings of the 30th AAAI Conference on Artificial Intelligence. AAAI Press, Phoenix, Arizona, pp. 165–171 (2016)
44. Liu, F., Liu, B., Sun, C., Liu, M., Wang, X.: Deep belief network-based approaches for link prediction in signed social networks. Entropy **17**(4), 2140–2169 (2015)
45. Liu, F., Liu, B., Sun, C., Liu, M., Wang, X.: Deep learning approaches for link prediction in social network services. In: Proceedings of the International Conference on Neural Information Processing, pp. 425–432. Springer-Verlag, Berlin, Heidelberg (2013)
46. Lazreg, M. B., Goodwin, M., Granmo, O.-C.: Deep learning for social media analysis in crises situations. In: Proceedings of the 29th Annual Workshop of the Swedish Artificial Intelligence Society (SAIS), Malmö, Sweden, pp. 1–46 (2016)
47. Liu, X., Zhu, T.: Deep learning for constructing microblog behavior representation to identify social media user's personality. Peer J. Comput. Sci. **2**, e81 (2016)
48. Chen, Z., Gao, B., Zhang, H., Zhao, Z., Liu, H., Cai, D.: User personalized satisfaction prediction via multiple instance deep learning. In: Proceedings of the 26th International Conference on World Wide Web, pp. 907–915. International World Wide Web Conferences Steering Committee, Republic and Canton of Geneva, CHE (2017)
49. Elkahky, A. M., Song, Y., He, X.: A multi-view deep learning approach for cross domain user modeling in recommendation systems. In: Proceedings of the 24th International Conference on World Wide Web, pp. 278–288. International World Wide Web Conferences Steering Committee, epublic and Canton of Geneva, CHE (2015).
50. Wang, H., Wang, N., Yeung, D.-Y.: Collaborative deep learning for recommender systems. In: Proceedings of the 21st ACM SIGKDD Conference on Knowledge. Discovery Data Mining. Association for Computing Machinery, pp. 1235–1244, New York, NY, USA (2015)

51. Deng, S., Huang, L., Xu, G., Wu, X., Wu, Z.: On deep learning for trustaware recommendations in social networks. IEEE Trans. Neural Netw. Learn Syst. **28**(5), 1164–1177 (2017)
52. Poria, S., Cambria, E., Gelbukh, A.: Deep convolutional neural network textual features and multiple kernel learning for utterance-level multimodal sentiment analysis. In: Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing, pp. 2539–2544. EMNLP (2015)
53. Poria, S., Cambria, E., Gelbukh, A.: Aspect extraction for opinion mining with a deep convolutional neural network. Knowledge-Based Syst. **108**, 42–49 (2016)
54. Guan, Z., Chen, L., Zhao, W., Zheng, Y., Tan, S., Cai, D.: Weakly supervised deep learning for customer review sentiment classification. In: Proceedings of the Twenty-Fifth International Joint Conference on Artificial Intelligence (IJCAI-16), pp. 3719–3725. AAAI Press, New York, USA (2016)
55. Araque, O., Corcuera-Platas, I., Sánchez-Rada, J.F., Iglesias, C.A.: Enhancing deep learning sentiment analysis with ensemble techniques in social applications. Expert Syst. Appl. **77**, 236–246 (2017)
56. Badjatiya, P., Gupta, S., Gupta, M., Varma, V.: Deep learning for hate speech detection in tweets. In: Proceedings of the 26th International Conference on World Wide Web Companion, pp. 759–760. International World Wide Web Conferences Steering Committee, Republic and Canton of Geneva, CHE (2017)
57. Stojanovski, D., Strezoski, G., Madjarov, G., Dimitrovski, I.: Finki at SemEval-2016 Task 4: Deep learning architecture for Twitter sentiment analysis. In: Proceedings of the 10th International Workshop on Semantic Evaluation (SemEval-2016), pp. 149–154. Association for Computational Linguistics (2016)
58. Detecting offensive language in tweets using deep learning. https://arxiv.org/abs/1801.04433. Last accessed 15 March 2020
59. Alali, M., Sharef, N. M., Hamdan, H., Murad, M. A. A., Husin, N. A.: Multi-layers convolutional neural network for twitter sentiment ordinal scale classification. In: Proceedings of the International Conference on Soft Computing and Data Mining, pp. 446–454. SCDM (2018)
60. Li, C., Xu, B., Wu, G., He, S., Tian, G., Hao, H.: Recursive deep learning for sentiment analysis over social data. In: Proceedings of the 2014 IEEE/WIC/ACM International Joint Conferences on Web Intelligence (WI) and Intelligent Agent Technologies (IAT), vol. 2, pp. 180–185. IEEE Computer Society, USA (2014)
61. Tang, D., Wei, F., Yang, N., Zhou, M., Liu, T., Qin, B.: Learning sentiment-specific word embedding for twitter sentiment classification. In: Proceedings of the 52nd Annual Meeting of the Association for Computational Linguistics, vol. 1, pp. 1555–1565. ACL (2014)
62. Bian, J., Gao, B., Liu, T.-Y.: Knowledge-powered deep learning for word embedding. In: Proceedings of the Joint European Conference on Machine Learning and Knowledge Discovery in Databases, pp. 132–148. Springer-Verlag, Berlin, Heidelberg (2014)
63. Pennington, J., Socher, R., Manning, C.: Glove: Global vectors for word representation. In: Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP), pp. 1532–1543. EMNLP (2014)
64. Severyn, A., Moschitti, A.: Learning to rank short text pairs with convolutional deep neural networks. In: Proceedings of the 38th International ACM SIGIR Conference on Research and Development in Information Retrieval. Association for Computing Machinery, pp. 373–382, New York, NY, USA (2015)
65. LSTM-based deep learning models for non-factoid answer selection. https://arxiv.org/abs/1511.04108. Last accessed 17 March 2020
66. Chandola, V., Banerjee, A., Kumar, V.: Anomaly detection: a survey. ACM Comput. Surv. **41**(3), 15 (2009)
67. Hawkins, D.M.: Identification of Outliers. Springer Science+Business Media, The Netherlands, 11, (1980)
68. Ebrahimi, M., Suen, C.Y., Ormandjieva, O.: Detecting predatory conversations in social media by deep convolutional neural networks. Digit. Invest. **18**, 3349 (2016)

69. A sensitivity analysis of (and practitioners' guide to) convolutional neural networks for sentence classification, https://arxiv.org/abs/1510.03820. Last accessed 17 March 2020.
70. Ribeiro, M., Lazzaretti, A.E., Lopes, H.S.: A study of deep convolutional auto-encoders for anomaly detection in videos. Pattern Recognit. Lett. **105**, 3–22 (2017)
71. Xu, D., Yan, T., Ricci, E., Sebe, N.: Detecting anomalous events in videos by learning deep representations of appearance and motion. Comput. Vis. Image Understand **156**, 117–127 (2016)
72. Hayat, M.K., Daud, A.: Anomaly detection in heterogeneous bibliographic information networks using co-evolution pattern mining. Scientometrics **113**, 149–175 (2017)
73. Feng, Y., Yuan, Y., Lu, X.: Learning deep event models for crowd anomaly detection. Neurocomputing **219**, 548556 (2017)
74. Know What's Working on Social Media: 27 Paid and Free Social Media Analytics Tools. Accessed 18 March 2020. Available from: https://buffer.com/library/social-media-analytics-tools
75. Facebook: How deep learning benefits social media. https://www.bernardmarr.com/default.asp?contentID=696. Last accessed 20 March 2020.
76. Google: Using Deep Learning AI To Drive Success. https://www.bernardmarr.com/default.asp?contentID=1275. Last accessed 20 March 2020
77. How Twitter Uses Big Data And Artificial Intelligence (AI). https://www.bernardmarr.com/default.asp?contentID=1373. Last accessed 20 March 2020