Lecture Notes in Social Networks

Mehmet Çakırtaş
Mehmet Kemal Ozdemir *Editors*

Big Data and Social Media Analytics

Trending Applications

Springer

Lecture Notes in Social Networks

Series Editors

Reda Alhajj, University of Calgary, Calgary, AB, Canada
Uwe Glässer, Simon Fraser University, Burnaby, BC, Canada
Huan Liu, Arizona State University, Tempe, AZ, USA
Rafael Wittek, University of Groningen, Groningen, The Netherlands
Daniel Zeng, University of Arizona, Tucson, AZ, USA

Advisory Board

Charu C. Aggarwal, Yorktown Heights, NY, USA
Patricia L. Brantingham, Simon Fraser University, Burnaby, BC, Canada
Thilo Gross, University of Bristol, Bristol, UK
Jiawei Han, University of Illinois at Urbana-Champaign, Urbana, IL, USA
Raúl Manásevich, University of Chile, Santiago, Chile
Anthony J. Masys, University of Leicester, Ottawa, ON, Canada
Carlo Morselli, School of Criminology, Montreal, QC, Canada

Lecture Notes in Social Networks (LNSN) comprises volumes covering the theory, foundations and applications of the new emerging multidisciplinary field of social networks analysis and mining. LNSN publishes peer-reviewed works (including monographs, edited works) in the analytical, technical as well as the organizational side of social computing, social networks, network sciences, graph theory, sociology, Semantics Web, Web applications and analytics, information networks, theoretical physics, modeling, security, crisis and risk management, and other related disciplines. The volumes are guest-edited by experts in a specific domain. This series is indexed by DBLP. Springer and the Series Editors welcome book ideas from authors. Potential authors who wish to submit a book proposal should contact Christoph Baumann, Publishing Editor, Springer e-mail: Christoph.Baumann@springer.com

More information about this series at http://www.springer.com/series/8768

Mehmet Çakırtaş • Mehmet Kemal Ozdemir
Editors

Big Data and Social Media Analytics

Trending Applications

Springer

Editors
Mehmet Çakırtaş
Bilkent yerleşkesi
Turkish Ministry of Health
Çankaya, Ankara, Turkey

Mehmet Kemal Ozdemir
Computer Engineering
Istanbul Medipol University
Istanbul, Turkey

ISSN 2190-5428 ISSN 2190-5436 (electronic)
Lecture Notes in Social Networks
ISBN 978-3-030-67043-6 ISBN 978-3-030-67044-3 (eBook)
https://doi.org/10.1007/978-3-030-67044-3

© The Editor(s) (if applicable) and The Author(s), under exclusive license to Springer Nature Switzerland AG 2021
This work is subject to copyright. All rights are solely and exclusively licensed by the Publisher, whether the whole or part of the material is concerned, specifically the rights of translation, reprinting, reuse of illustrations, recitation, broadcasting, reproduction on microfilms or in any other physical way, and transmission or information storage and retrieval, electronic adaptation, computer software, or by similar or dissimilar methodology now known or hereafter developed.
The use of general descriptive names, registered names, trademarks, service marks, etc. in this publication does not imply, even in the absence of a specific statement, that such names are exempt from the relevant protective laws and regulations and therefore free for general use.
The publisher, the authors, and the editors are safe to assume that the advice and information in this book are believed to be true and accurate at the date of publication. Neither the publisher nor the authors or the editors give a warranty, expressed or implied, with respect to the material contained herein or for any errors or omissions that may have been made. The publisher remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.

This Springer imprint is published by the registered company Springer Nature Switzerland AG
The registered company address is: Gewerbestrasse 11, 6330 Cham, Switzerland

Contents

Twenty Years of Network Science: A Bibliographic and Co-authorship Network Analysis

Roland Molontay and Marcell Nagy

Abstract Two decades ago three pioneering papers turned the attention to complex networks and initiated a new era of research, establishing an interdisciplinary field called network science. Namely, these highly-cited seminal papers were written by Watts and Strogatz, Barabási and Albert, and Girvan and Newman on small-world networks, on scale-free networks and on the community structure of complex networks, respectively. In the past 20 years – due to the multidisciplinary nature of the field – a diverse but not divided network science community has emerged. In this chapter, we investigate how this community has evolved over time with respect to speed, diversity and interdisciplinary nature as seen through the growing co-authorship network of network scientists (here the notion refers to a scholar with at least one paper citing at least one of the three aforementioned milestone papers). After providing a bibliographic analysis of 31,763 network science papers, we construct the co-authorship network of 56,646 network scientists and we analyze its topology and dynamics. We shed light on the collaboration patterns of the last 20 years of network science by investigating numerous structural properties of the co-authorship network and by using enhanced data visualization techniques. We also identify the most central authors, the largest communities, investigate the spatiotemporal changes, and compare the properties of the network to scientometric indicators.

Keywords Science of science · Network science · Bibliometrics · Scholarly data · Scholarly network analysis · Co-authorship network

R. Molontay (✉)
Department of Stochastics, Budapest University of Technology and Economics, Budapest, Hungary

MTA–BME Stochastics Research Group, Budapest, Hungary
e-mail: molontay@math.bme.hu

M. Nagy
Department of Stochastics, Budapest University of Technology and Economics, Budapest, Hungary
e-mail: marcessz@math.bme.hu

© The Author(s), under exclusive license to Springer Nature Switzerland AG 2021

M. Çakırtaş, M. K. Ozdemir (eds.), *Big Data and Social Media Analytics*, Lecture Notes in Social Networks, https://doi.org/10.1007/978-3-030-67044-3_1

1 Introduction

Complex networks have been studied extensively since they efficiently describe a wide range of systems, spanning many different disciplines, such as Biology (e.g. protein interaction networks), Information Technology (e.g., WWW, Internet), Social Sciences (e.g., collaboration, communication, economic, and political networks), etc. Moreover, not only the networks originate from different domains, but the methodologies of network science as well, for instance, it heavily relies on the theories and methods of graph theory, statistical physics, computer science, statistics, and sociology.

In the last two decades, network science has become a new discipline of great importance. It can be regarded as a new academic field since 2005 when the U.S. National Research Council defined network science as a new field of basic research [11]. The most distinguished academic publishing companies announce the launch of new journals devoted to complex networks, one after another (e.g. Journal of Complex Networks by Oxford University Press, Network Science by Cambridge University Press, Applied Network Science and Social Network Analysis and Mining by Springer). Network science also has its own prestigious conferences attended by thousands of scientists. Leading universities continuously establish research centers and new departments for network science, furthermore, launch Master and Ph.D. programs in this field (such as Yale University, Duke University, Northeastern University, and Central European University).

The significance of network theory is also reflected in the large number of publications on complex networks and in the enormous number of citations of the pioneering papers by Barabási and Albert [5], Watts and Strogatz [35] and Girvan and Newman [13]. Some researchers interpret network science as a new paradigm shift [16]. However, complex networks are not only acknowledged by the research community, but innovative textbooks aimed for a wider audience have also been published [7, 23], moreover, the concepts of network science have appeared in the popular literature [4, 34] and mass media [30] as well.

In the last two decades, complex networks became in the center of research interest thanks to – among many others – the aforementioned three pioneering papers and due to the fact that the prompt evolution of information technology has opened up new approaches to the investigation of large networks. This period of 20 years can be regarded as the golden age of network science. The first challenge was to understand network topology, to this end, structural properties were put under the microscope one after the other (small-worldness, scale-free property, modularity, fractality, etc.) and various network models were proposed to understand and to mathematically describe the architecture and evolution of real-world networks [33]. In recent years, there has been a shift from the structural analysis to studying the control principles of complex networks [3]. Remarkable computing power, massive datasets, and novel computational techniques keep great potential for network scientists for yet another 20 years [33].

This work is a tribute to the achievements of the network science community in the past 20 years. We provide a bibliographic analysis of 31,763 network science

papers and we also construct and investigate the co-authorship network of network scientists to identify how the network science community has been evolving over time.

The present study also extends the earlier conference version of this chapter [21] in several important directions. Namely, here we provide a more detailed literature review; we examine a longer time period; and we answer the question of how the network science community has evolved over time with respect to speed, diversity, and interdisciplinary nature by implementing novel analyses. Here we also provide an analysis of the co-occurrence network of the keywords. Moreover, we supplement our previous work with several other new methods and data visualizations that help to make insightful observations regarding the last two decades of network science.

The main contributions of this work can be summarized as follows:

- We collect 31,763 network science papers and provide a bibliographic analysis investigating various characteristics of the papers and showing how the discipline has developed over time.
- We construct the co-authorship network of 56,646 network scientists and undertake a scholarly network analysis study by analyzing its topology and dynamics.
- We answer the following major research questions:
  - What are the most important venues of network science and how have they changed over time?
  - How the publication patterns vary over research areas and time?
  - What are the most important topics of network science and how have they evolved through time? What relationships can we explore among the most frequent keywords of network science?
  - How the network science community has evolved over time with respect to speed, diversity, and interdisciplinary nature?
  - What are the most typical patterns in terms of international and interdisciplinary collaborations?
  - Who are the most central authors and how do the largest communities look like? How do these network properties compare to other scientometric indicators?

2 Scholarly Networks Analysis

The present paper joins the line of research focused on scholarly network analysis that is based on big scholarly data [28, 37]. Big scholarly data refers to the rapidly growing data accessible in digital libraries containing information on millions of scientific publications and authors [36]. The easily available data sources (Web of Science, Scopus, PubMed, Google Scholar, Microsoft Academic, the U.S. Patent and Trademark Office, etc.) together with novel powerful data analysis technologies have led to the emergence of science of science [12] that gives us a better understanding of the self-organizing rules and patterns of science, e.g. how

disciplines have emerged and evolved over time [19]. Various scholarly networks at many levels can be formed based on scholarly data; Pawar et al. identify the following forms of scholarly networks of great interest [28]:

1. co-authorship networks (a link is formed between scientists by their co-authorship of at least one scientific paper),
2. citation networks (a directed link is formed between documents referencing one another),
3. co-citation networks (a link is formed between documents if they are cited together),
4. bibliographic coupling (documents are linked if they share common references),
5. co-occurrence networks (keywords/topics are linked if they occur in the same document), and
6. heterogeneous networks (two or more coupled scholarly networks).

Among the aforementioned scholarly networks perhaps co-authorship networks have attracted the greatest deal of research interest, owing to the fact that co-authorship is one of the most important reflections of research collaboration, which is an essential mechanism that joins together distributed knowledge and expertise into novel discoveries. Furthermore, building the map of sciences is not only important for sociologists and other scholars to understand researchers' interaction but for policymakers as well to address sharing resources [36]. Co-authorship networks have been studied extensively in various ways and from various perspectives: e.g. the collaboration network determined by the articles of a certain journal, a specific country or a research community that cites a particular influential paper or author [1, 2, 6, 17, 24, 25]. In this chapter, we investigate the co-authorship network of network scientists as defined in the following section.

Keywords co-occurrence networks have also been investigated thoroughly [15, 20, 29, 31]. Keywords of academic articles can provide a concise overview of the content and the core idea of the body of the papers. In contrast to word clouds, co-occurrence networks do not only show the frequency of the keywords but also allow us to discover the relationship between them. Li et al. investigated 6,900 articles published between 1982 and 2013 which had been indexed using the keyword 'complex network(s)' and provided a co-keyword network and a keyword co-occurrence network analysis [20].

3 Preliminaries and Data

In this section, we describe how the co-authorship network of network scientists was constructed, how the examined set of academic publications was chosen and collected, what data preparation steps were conducted. Moreover, we also present some useful notions and preliminaries.

3.1 Co-authorship Network of Network Scientists

To the best of our knowledge, the co-authorship network of network scientists has been analyzed only by Newman et al. [26, 27]. However, their network consists of 1,589 authors, while this study investigates a much larger network (56,646 vertices and 357,585 edges) spanning a longer time horizon (1998–2019).

We construct the co-authorship network of network scientists as follows. We consider three ground-breaking papers around the millennium that can be regarded as the roots of the rise of network science: the paper of Watts and Strogatz [35] on small-world networks, the work of Barabási and Albert [5] about scale-free networks and the paper of Girvan and Newman [13] that reveals the community structure of complex networks. We selected the aforementioned three papers since they initiated new areas of research in network science by introducing pivotal concepts two decades ago, that had a huge impact on the network science community that is also demonstrated by the large number of citations they received in the past 20 years.

In this work, we consider a paper as a *network science paper* if it cites at least one of the three aforementioned pivotal articles (in addition, the three originating papers are also regarded as network science papers, obviously). Similarly, we call someone a *network scientist* if (s)he has at least one network science paper. The previous definitions of a network science paper and a network scientist are of course quite arbitrary. It is important to note that the papers that refer to one of the three seminal papers are not necessarily about network science and there certainly exist network science articles that do not refer to any of the aforementioned pioneering papers. On the other hand, we believe that this concept is a good proxy for our purposes and it is worth studying. We construct the co-authorship network of the network scientists where two of them are connected if they have at least one joint network science paper (see Fig. 1). In other words, this network is a one-mode projection onto scientists, from the bipartite network of scientists and the network science papers they authored. The anonymized data of the constructed network and some figures in high resolution are available in the supplementary material [22].

3.2 Glossary

Definition 1 (Graph) A simple (undirected) graph is an ordered pair $G = (V, E)$, where V is the set of vertices or nodes and E is the set of edges or links, which are two-element subsets of V. The vertex and edge sets of G are denoted by $V(G)$ and $E(G)$ respectively. The size of the graph is the number of its nodes, and it is usually denoted by n.

Definition 2 (Complex network) In network theory, the terms graph and network are used interchangeably, however, a complex network is a graph with non-trivial topological features, that characterize real-world networks.

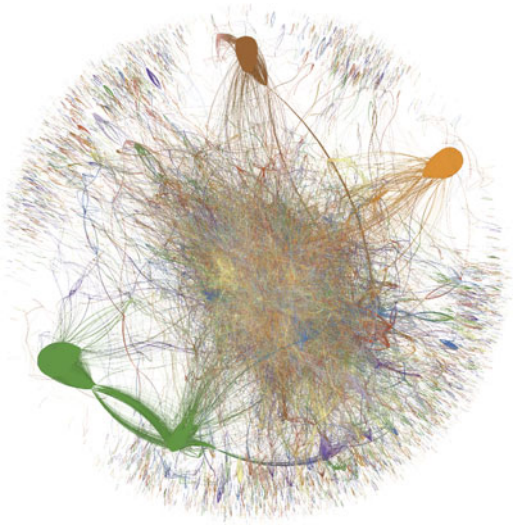


Fig. 1 The co-authorship network of network scientists colored by communities

Definition 3 (Average path length) A path is a sequence of edges which connect a sequence of vertices. The distance $d(u, v)$ between the vertices u and v is the length (number of edges) of the shortest path connecting them. The l_G average path length of a graph G of size n is defined as:

$$l_G = \frac{1}{n(n-1)} \sum_{\substack{u,v \in V(G) \\ u \neq v}} d(u, v).$$

Definition 4 (Small-world property) A network is said to be small-world, if the average path length is proportional to the logarithm of the size of the network i.e. $l_G \sim \log |V|$.

Definition 5 (Degree distribution) The degree $\deg(v)$ of a vertex v in a simple, undirected graph is its number of incident edges. The degree distribution P is the probability distribution of the degrees over the whole network, i.e. $P(k)$ is the probability that the degree of a randomly chosen vertex is equal to k.

Definition 6 (Scale-free property) A scale-free network is a connected graph which $P(k)$ degree distribution follows a power law asymptotically, i.e. $P(k) \sim k^{-\gamma}$, where $\gamma \geq 1$.

Definition 7 (Assortativity coefficient) The assortativity coefficient is the Pearson correlation coefficient of degree between pairs of linked nodes. The assortativity coefficient is given by

$$r = \frac{\sum_{j,k} j \cdot k(e_{j,k} - q_j q_k)}{\sigma_q^2},$$

where the term q_k is the mass function of the distribution of the remaining degrees (degree of the nodes minus one) and j and k indicates the remaining degrees. Furthermore, $e_{j,k}$ refers to the mass function of the joint probability distribution of the remaining degrees of the two vertices. Finally, σ_q^2 denotes the variance of the remaining degree distribution with mass function q_k i.e. $\sigma_q^2 = \sum_k k^2 q_k - \left(\sum_k k q_k\right)^2$.

Definition 8 (Local clustering coefficient) The local clustering coefficient of vertex v is the fraction of pairs of neighbors of v that are connected over all pairs of neighbors of v. Formally:

$$C_{\text{loc}}(v) = \frac{|\{(s,t) \text{ edges} : s,t \in N_v \text{ and } (s,t) \in E|\}}{\deg(v)(\deg(v)-1)},$$

where N_v is the neighborhood of the node v i.e. the vertices adjacent to v.

The average (local) clustering coefficient of a G graph is defined as:

$$\bar{C}(G) = \frac{1}{n} \sum_{v \in V(G)} C_{\text{loc}}(v),$$

where n is the size of the graph.

Definition 9 (Global clustering coefficient) The global clustering coefficient C of the graph G is the fraction of closed triplets (paths of length two in G that are closed) over all of the triplets (paths of length two) in G.

Definition 10 (Betweenness centrality) Betweenness centrality of a node v is given by the expression:

$$g(v) = \sum_{s \neq v \neq t} \frac{\sigma_{st}(v)}{\sigma_{st}},$$

where σ_{st} is the total number of shortest paths from node s to node t and $\sigma_{st}(v)$ is the number of those paths that pass through v.

Definition 11 (h-index) The h-index is an author-level metric defined as the maximum value of h such that the given author has published h papers that have each been cited at least h times.

Definition 12 (Harmonic centrality) Harmonic centrality of a node v is the sum of the reciprocal of the shortest path distances from all other nodes to v:

$$H(v) = \sum_{u \neq v} \frac{1}{d(u,v)}.$$

3.3 Data Collection and Preparation

We build our analysis on data collected from the Web of Science bibliographic database, retrieved on January 2, 2020. The collected data consist of 41,245 rows with multiplicity corresponding to the citing works of the three seminal articles [5, 13, 35]. For each citing paper we have information on the document title, publication name, publication type, publisher, publication year, authors' full name, authors' address, research area, keywords, cited reference count, total times cited count, page count, abstract, etc.

After the data were collected, various data preparation steps were conducted, including merging the files, handling missing fields, deleting duplicates, and indicating which of the three seminal papers were cited by the given article. These preparation steps reduced the dataset to 31,763 unique rows and the citation pattern of the corresponding articles is shown in Fig. 2.

The authors are represented by the full name field of Web of Science, however, this field is unfortunately not consistent, the author called John Michael Doe may appear as Doe, John; John Doe; Doe, J.; Doe, J. M.; Doe, John Michael, and other variants. To overcome this issue, we created a dictionary that defines the name variants that correspond to the same author. Furthermore, we cannot distinguish between different scientists with the same name, this issue is mainly relevant for Asian authors. However, the error introduced by this problem is negligible, as also pointed out by Newman [24] and by Barabási et al. [6].

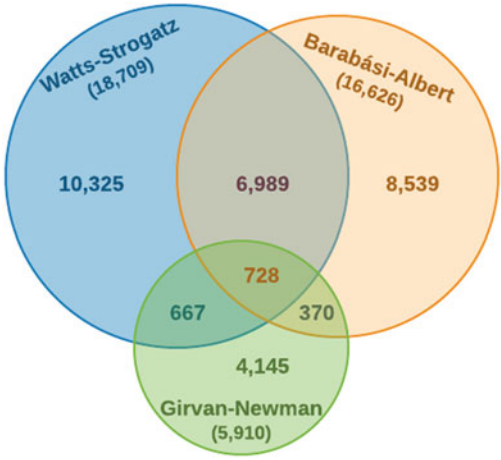


Fig. 2 Distribution of the citations among the three pioneering papers

4 Analysis of Network Science Papers

First, we analyze the enormous number of citing works, i.e. the network science papers. Figure 3 shows the top 10 research areas where the citing works belong to, illustrating the interdisciplinary nature of network science. We can see that the first decade was dominated by physics while later computer science took over. It is also clear from the figure that neuroscience has started to use tools of network science in the last decade. The journals that publish the most network science papers are shown in Fig. 4. Considering the number of publications, Physical Review E was the leading scientific forum of network science in the first half of the examined period, while PLOS One and Scientific Reports emerged in the last decade. Currently, Physica A can be regarded as the leading journal of network science in terms of the number of published network science articles.

Figure 5 shows the number of collaborating authors per citing works, the most typical numbers of co-authors in a network science paper are 2 and 3. Almost one-tenth of the network science papers are sole-authored, M. E. J. Newman has the highest number of sole-authored network science papers, namely 27. While the figure shows only up to 15 number of authors, there are a few papers with a high number of collaborating authors e.g. the paper with the highest number of authors (387) is a paper of the Alzheimer's Disease Neuroimaging Initiative [18]. The authors of this article emerge as a maximal clique of the co-authorship network of network scientists as it can be seen in Fig. 1.

We also investigate the distribution of network science papers written by network scientists. The authors with the highest number of network science papers together with the citation count of their network science papers are shown in Table 1.

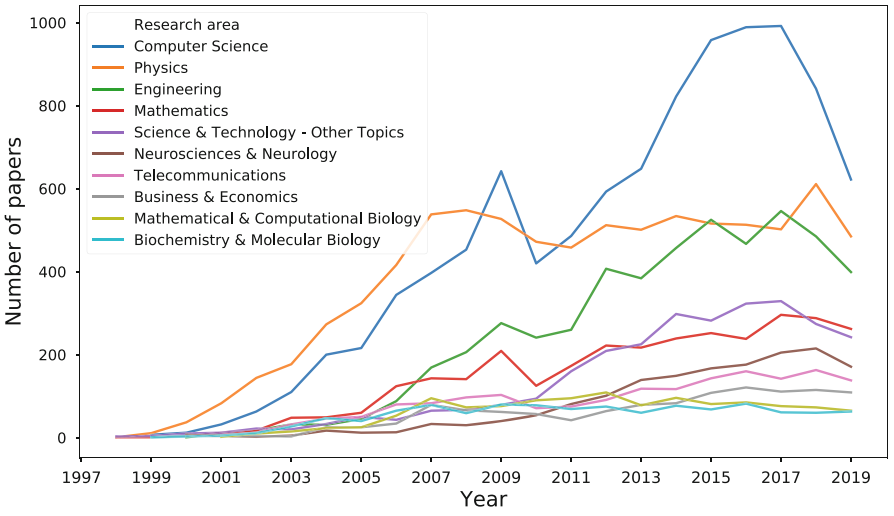


Fig. 3 Top 10 research areas of the network science papers

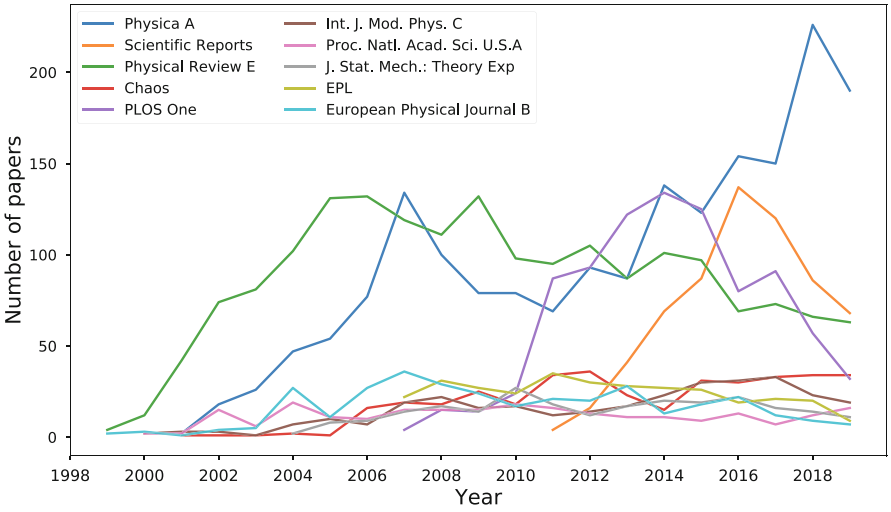


Fig. 4 Top 10 journals of the network science papers

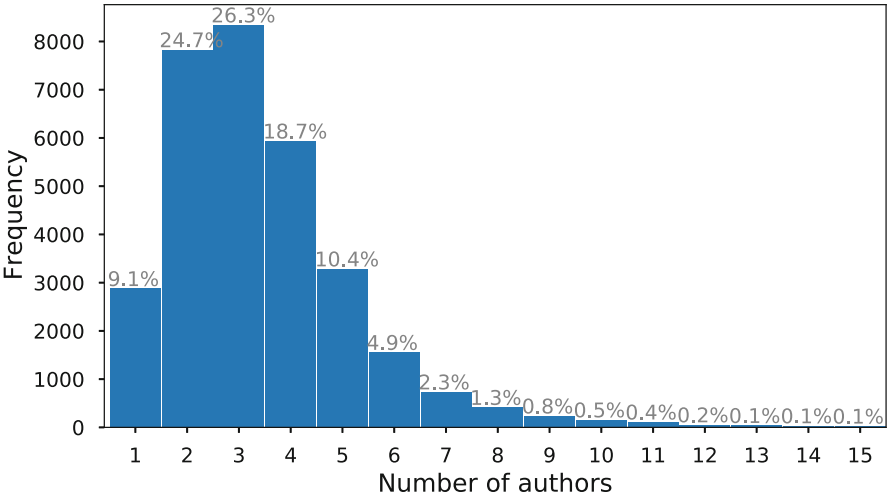


Fig. 5 Histogram of the number of authors per paper (truncated at 15)

Guanrong Chen has the highest number of network science papers, his research areas are nonlinear systems and complex network dynamics and control.

To gain some insights on how the publication patterns vary in network science depending on the research area, we show how the distribution of the number of cited references and the length of the papers differ across research areas (see Fig. 6). We can observe that in neuroscience and neurology authors typically cite a high number of articles while in computer science or engineering the typical number of

Table 1 Top 12 authors with the most network science papers

Name of author	Number of papers	Number of citations
Guanrong Chen	167	12,859
Bing-Hong Wang	145	5,341
Tao Zhou	138	9,911
Shlomo Havlin	124	13,377
Jürgen Kurths	118	9,249
Eugene H. Stanley	113	10,479
Zhongzhi Zhang	104	2,099
Ying-Cheng Lai	99	5,799
Albert-László Barabási	95	73,937
Luciano da Fontoura Costa	92	2,726
Matjaz Perc	89	8,634
Michael Small	88	2,345

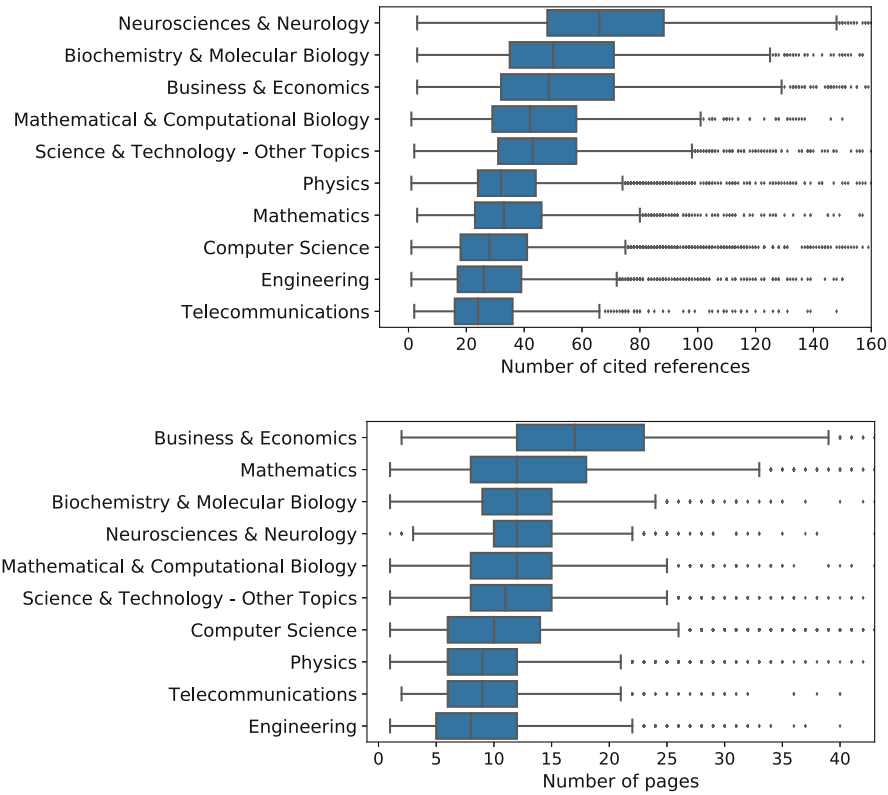


Fig. 6 Boxplots of the number of cited references and length of network science papers across research areas

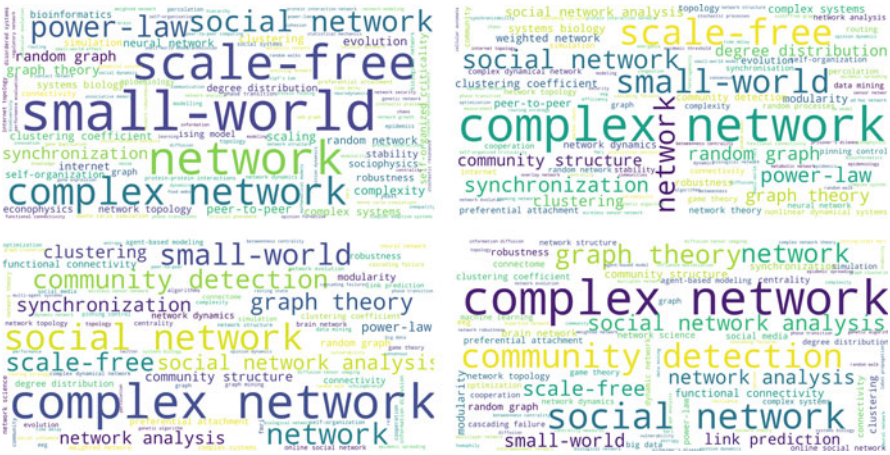


Fig. 7 Word cloud of the most frequent keywords of network science papers before 2006 (top right), between 2006 and 2010 (top left), between 2011 and 2015 (bottom left) and since 2016 (bottom right)

cited references is much smaller. Engineering together with telecommunications and physics are also the areas with the shortest articles, while it can also be due to the fact that in those disciplines double-column publication formats are quite typical. It is important to emphasize that these observations are not necessarily representative of the disciplines in general, only for those papers that were defined as network science papers.

Figure 7 depicts separate word clouds of the most frequently used keywords of network science papers written in the four 5-year-long periods of the last two decades of network science. We can observe that in the first half of the examined period structure related (e.g., scale-free, small-world, topology) and modeling related keywords (e.g., preferential attachment, evolution, growing network, small-world model) dominated the study of complex networks, while in the last decade topics such as community detection, social network analysis, data-driven research (big data, data mining, link prediction, machine learning) have become more popular keywords. In the first decade of the examined period, the most studied real-world networks were Internet, peer-to-peer, and protein interaction networks, however, since 2010 the research has tended to focus on online social networks and brain networks. These word clouds can also be found in the supplementary material [22].

To gain a more comprehensive understanding of the keywords of network science papers, we also construct their co-occurrence network. Figures 8 and 9 together with Fig. 7 clearly shows that ‘complex network’ is the most important keyword of the examined set of papers that supports our hypothesis that our definition of network science paper was indeed a good proxy of our purposes. We can observe that the keyword ‘random graph’ often goes together with ‘network’ but less often with

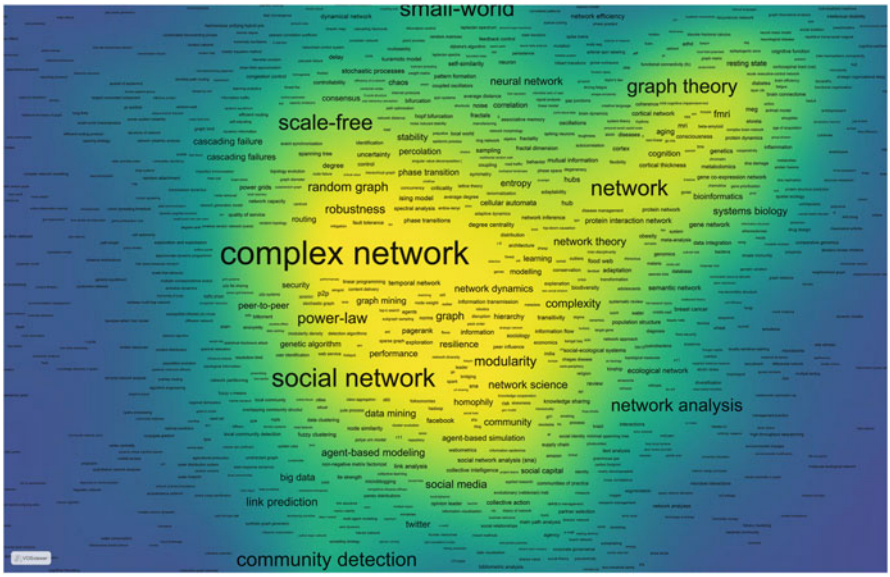


Fig. 8 Visualization of two-dimensional knowledge map of keywords. Keywords that have co-occurred more frequently are placed closer to each other on the map. The font size indicates the number and strength of the connections of a keyword. A more intense color implies a larger number of keywords and higher connectivity in the neighborhood of the point. The figure was created with VOSviewer [32]

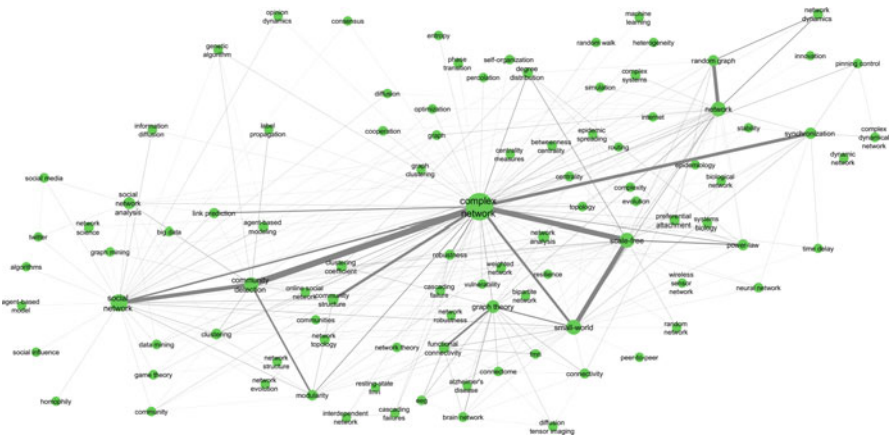


Fig. 9 Co-occurrence network of keywords. The size of the node indicates the frequency of keywords in network science papers, the edge width indicates their relative co-occurrence. Only keywords with frequency at least 100 and edges with weight at least 10 are shown in the figure

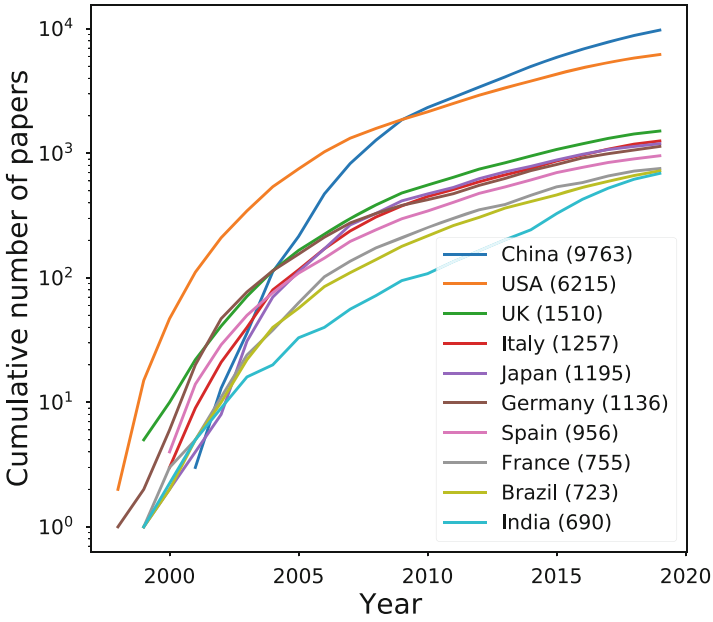


Fig. 10 Cumulative number of network science papers on a logarithmic scale by the country of the first author (only the Top 10 countries are shown)

'complex network' that suggests the term 'complex network' is not that widespread among mathematicians. Another observation is that community detection is rather popular in the social network domain, the bottom left side of the figures is dominated by terms associated with social network analysis and community detection. We can also observe that the keywords 'scale-free' and 'small-world' are frequently used together. The complete keyword co-occurrence network and the Figs. 8 and 9 can be found in the supplementary material [22].

Based on the address of the first author, we identify the network science hot-spots and investigate the spatiotemporal changes. Figure 10 demonstrates that China and the USA are the two leading countries of network science with a fast increase in Chinese network science papers in the last few years.

Figure 11 illustrates how the ratio of multidisciplinary and international papers varied over the years. We can observe that network science research gets both increasingly international and multidisciplinary. Here we consider a paper international if it has at least two co-authors who do not share an affiliation within the same country. The ratio of international papers in an important indicator, since it was also shown that scientific impact increases if researchers publish in international collaboration [8]. The multidisciplinary nature of papers is defined by the fact that more than one research area is attached to the document in Web of Science.

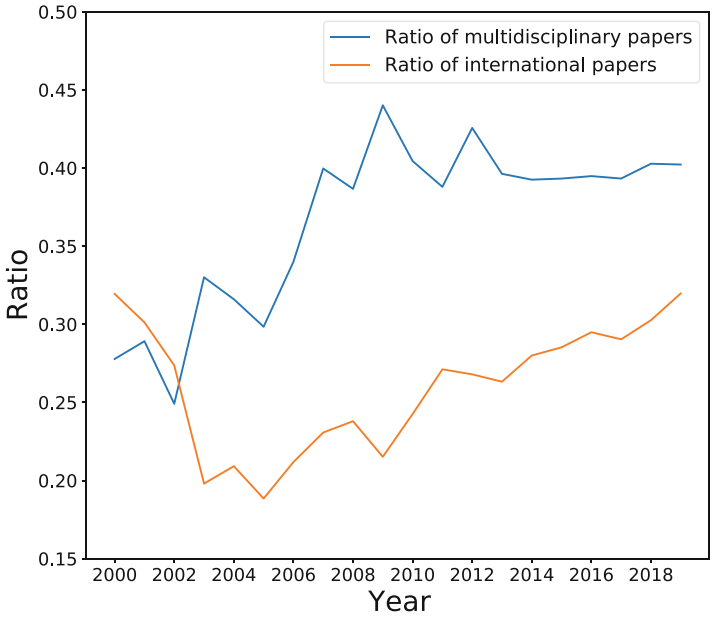


Fig. 11 Ratio of multidisciplinary and international papers since 2000

5 Analysis of the Co-authorship Network

The nodes of the co-authorship network of network scientists correspond to the authors who have at least one network science paper (i.e., a paper that cites at least one of the three seminal papers [5, 13, 35]), two of them are connected if they co-authored at least one network science paper. The network is simple, undirected, and unweighted meaning that here we ignore the strength of the connection between two scientists, i.e. the number of their joint papers. The network has 56,646 nodes and 357,585 edges with an average degree of 12.63, however, the median degree is just 4. The largest connected component consists of 35,716 nodes and it is depicted in Fig. 15.

The degree distribution of the network is illustrated in Fig. 12. There are 897 isolated nodes in the graph (nodes with zero degrees), i.e. scholars who have a single-authored network science paper but have not co-authored any network science papers. The most typical number of co-authors are between 2 and 4 and the tail of the distribution decays much slower than the number of authors per paper does (c.f. Fig. 5) since here the degree reflects all the number of co-authors who do not necessarily author the same paper. The highest degree is 546 corresponding to Roberto Bellotti, a medical physicist, who is also an author of the paper with the highest number of collaborating authors [18] and another many-authored paper [9]. While our network is unweighted by definition, a possible weight could be assigned to the edges corresponding to the number of joint papers written by the two authors

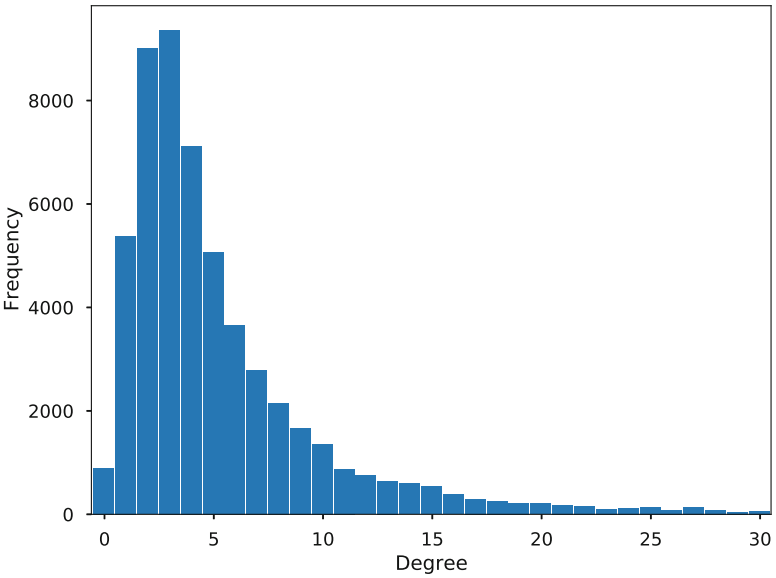


Fig. 12 The degree distribution of the network (truncated at 30)

Table 2 The most active links between authors

Authors		Number of joint papers
Shlomo Havlin	Eugene H. Stanley	52
Bing-Hong Wang	Tao Zhou	51
Jihong Guan	Shuigeng Zhou	50
Zhongzhi Zhang	Shuigeng Zhou	48
Jihong Guan	Zhongzhi Zhang	40
Zeng-Ru Di	Ying Fan	34
Sergey Dorogovtsev	José F.F. Mendes	32

at the endpoints of the edge. Table 2 shows the most 'active links', i.e. the edges with the highest weights in the edge-weighted version of the co-authorship network.

The network has a high assortativity coefficient of 0.53 that suggests that nodes tend to be connected to other nodes with similar degrees. The co-authorship network is highly clustered with a global clustering coefficient of 0.97 and an average local clustering coefficient of 0.8. The fact that the average shortest path length in the largest connected component is 6.6 also supports the small-world nature of co-authorship networks.

To identify the most central authors of the network science community as seen through the co-authorship network, we calculate centrality measures such as betweenness, harmonic and degree centralities of the nodes. The most central authors are shown in Table 3. We also compare the centrality measures of the authors with the citation count of their network science papers and with their *h*-

Table 3 The top 12 authors with the highest betweenness centrality. Their ranks with respect to each metric are shown in brackets

	Centralities				
Name	Betweenness	Harmonic	Degree	Number of citations	*h*-index
Jürgen Kurths	**0.025 (1)**	**0.169 (1)**	216 (1,017)	9,249 (30)	96 (2)
H. Eugene Stanley	0.024 (2)	0.168 (2)	220 (1,013)	10,479 (18)	57 (7)
Guanrong Chen	0.019 (3)	0.165 (4)	215 (1,018)	12,859 (15)	27 (30)
Albert-László Barabási	0.017 (4)	0.160 (12)	202 (1,023)	**73,937 (2)**	83 (3)
Yong He	0.014 (5)	0.163 (6)	**242 (1,012)**	9,104 (32)	61 (6)
Zhen Wang	0.014 (6)	0.163 (5)	155 (1,117)	3,306 (365)	39 (16)
Santo Fortunato	0.013 (7)	0.160 (13)	208 (1,021)	13,923 (12)	40 (14)
Shlomo Havlin	0.013 (8)	0.163 (7)	165 (1,042)	13,377 (13)	**110 (1)**
Tao Zhou	0.013 (9)	0.167 (3)	220 (1,013)	9,911 (20)	40 (14)
Edward T. Bullmore	0.012 (10)	0.151 (49)	210 (1,020)	17,915 (7)	50 (10)
Wei Wang[a]	0.012 (11)	0.161 (10)	145 (1,178)	467 (1511)	14 (188)
Stefano Boccaletti	0.112 (12)	0.162 (9)	130 (1,179)	9,609 (21)	22 (58)

[a]Sichuan University

indices (restricted only to their network science papers). Common characteristics of the most central authors that they are famous, well-established researchers, moreover, they are typically active in more research areas forming bridges between subdisciplines. The highest betweenness and harmonic centralities correspond to Jürgen Kurths, German physicist and mathematician whose research is mainly concerned with nonlinear physics and complex systems sciences. As we mentioned before, the highest degree corresponds to Roberto Bellotti, a medical physicist. Mark Newman English-American physicist has the highest number of citations on his network science papers (77,418), while Shlomo Havlin, Israeli physicist is ranked first with respect to *h*-index.

Figure 14 illustrates the relationship between centrality measures of network scientists and the scientometric indicators of their network science papers. On the left it shows the number of citations against the vertex betweenness centrality, colored by the harmonic centrality; on the right one can see the *h*-index against the vertex betweenness centrality, colored by the degree. We can conclude that there is a positive correlation between the authors' central role in the co-authorship network and their scientometric indicators. Figure 13 shows the Spearman's rank correlation heatmap of the aforementioned measures indicating positive correlations, with the highest positive correlation between citation count and *h*-index. Considering centrality measures against scientometric indicators, betweenness centrality and *h*-index has the highest correlation (Figs. 13 and 14).

Network scientists have become more connected as time has gone by, as it is illustrated in Fig. 16, since not only the size of the largest component has increased over the years but also the ratio of the size of the giant component to the size of the entire network, indicating the emergence of a diverse but not divided network

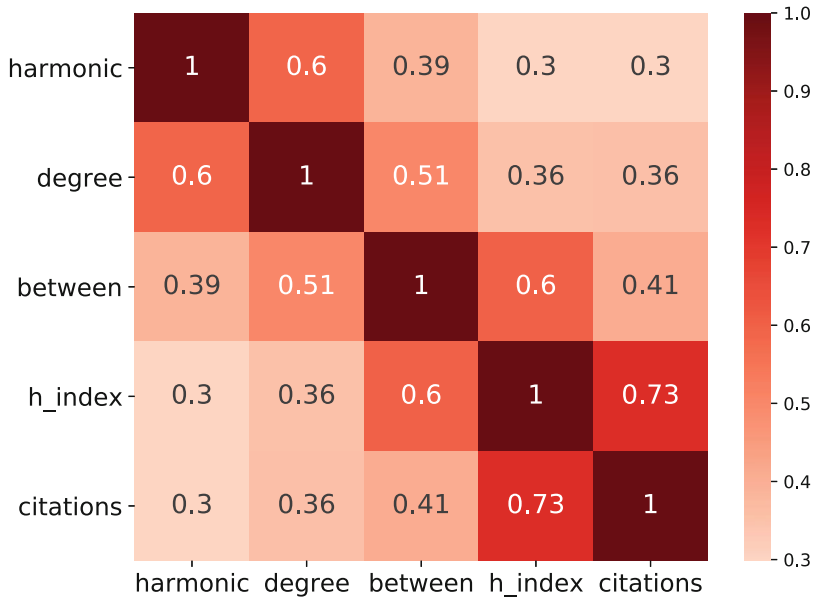


Fig. 13 Spearman correlation heatmap between various centrality measures and scientometric indicators

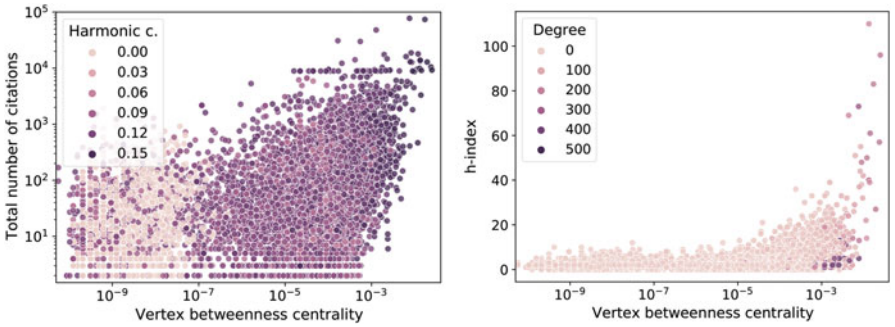


Fig. 14 Relationship between centrality measures of network scientists and the scientometric indicators of their network science papers

science community. The giant component consists of 35,716 nodes that is 63% of the entire network size and it is illustrated in Fig. 15.

Using Clauset-Newman-Moore greedy modularity maximization community detection algorithm [10], we identify the dense subgraphs of the network. To retrieve some important discipline and location-related characteristics of the largest communities, we assigned a research area and a country for each author as the majority of the research areas corresponding to their papers and the most frequent country of their affiliations respectively. The compositions of the ten largest

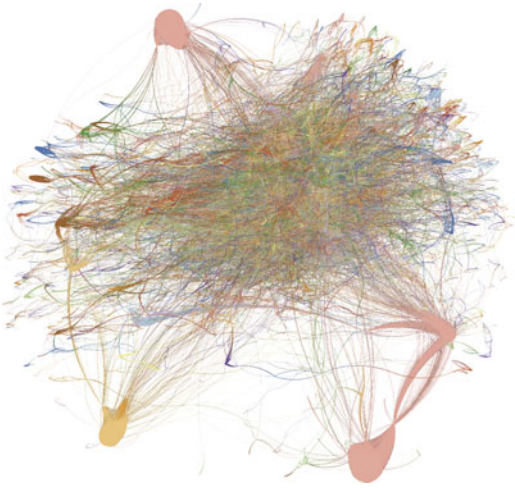


Fig. 15 The largest connected component of the co-authorship network of network scientists colored by communities

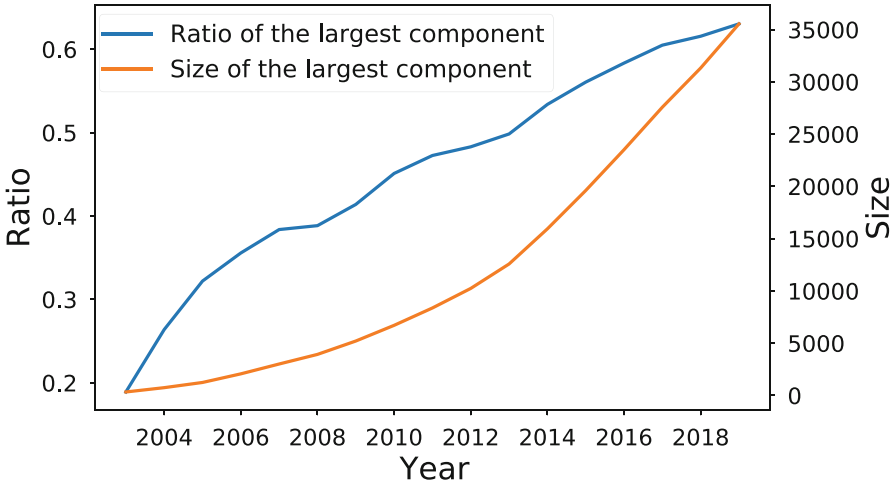


Fig. 16 The absolute and relative size of the largest connected component of the co-authorship network

communities are shown in Table 4. The largest community consists of 15,693 authors dominated by Chinese physicists and computer scientists. We can observe that the smaller the communities are, the more homogeneous they are. For example, the vast majority of the third-largest community are North American neuroscientists, moreover, there is a community with 53% EU scientists and 44% environmental scientists.

Network scientists come from 118 different countries which shows the international significance of network science. To illustrate the typical patterns of international collaborations, we created an edge-weighted network of countries where edge weights correspond to the number of network science papers that were

Table 4 Composition of the largest communities

Size	Research area			Country/region		
15,693	**PHY 31%**	**CS 30%**	NN 10%	**CHN 55%**	EU 14%	USA 12%
1,066	**CS 31%**	PHY 15%	BMB 12%	CHN 31%	USA 26%	EU 17%
812	**NN 72%**	CS 5%	PSY 4%	**USA 90%**	CAN 3%	CHN 2%
759	**CS 30%**	PHY 19%	BMB 16%	EU 36%	USA 29%	ISR 8%
756	NN 23%	CS 17%	PHY 16%	EU 34%	JPN 17%	KOR 15%
711	**CS 30%**	MAT 10%	LSB 7%	USA 35%	EU 29%	CHN 9%
633	CS 19%	PHY 18%	BMB 8%	CHN 27%	EU 23%	IND 13%
563	**ESE 44%**	CS 11%	LSB 8%	**EU 53%**	BRA 12%	USA 12%
559	**CS 42%**	ENG 14%	PHY 12%	USA 30%	EU 20%	IRN 11%
555	BMB 35%	CS 23%	MCB 8%	USA 30%	EU 24%	CHN 21%

ACS: Automations & Control Systems	**BE**: Business & Economics
BMB: Biochemistry & Molecular Biology	**CS**: Computer Science
ESE: Environmental Sciences & Ecology	**GH**: Genetics & Heredity
LSB: Life Sciences & Biomedicine	**MAT**: Mathematics
MCB: Mathematical & Computational Biology	**NN**: Neurosciences & Neurology
PHY: Physics	**PSY**: Psychiatry
SCT: Science & Technology	**TEL**: Telecommunication

The country abbreviations are the officially assigned ISO alpha-3 codes [14]

written in the collaboration of at least one author from both countries (see Fig. 17). We can observe that while China has the highest number of network science papers (see also Fig. 10), US scientists wrote the most articles in international collaboration. It is also apparent that EU countries collaborate with each other a lot.

Similarly to the network of international collaborations, we also created a network of multidisciplinary collaborations illustrating the importance of multidisciplinary research in network science. Figure 18 shows an edge-weighted network of research areas where the edge weights correspond to the number of network science papers that were written in the collaboration of authors whose main research areas are the ones at the endpoints of the edge. The main research area of the authors is not given in the Web of Science, so for each author, we assigned the most frequent research area associated with their papers. We can observe that computer scientists and physicists dominate network science. It is also clear that the collaboration of physicists and network scientists made huge progress in network science. We can conclude that – as far as network science papers are concerned – mathematicians collaborate the most with physicists, while engineers collaborate more with computer scientists. It is not surprising that telecommunication experts usually collaborate with engineers and computer scientists, while mathematical & computational biologists work a lot with biochemists & molecular biologists and computer scientists on network science papers.

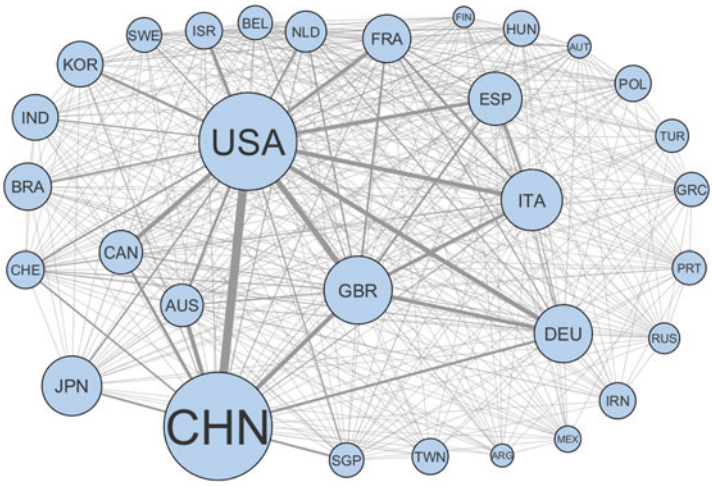


Fig. 17 Network of international collaborations. The size of the node corresponds to the number of network science papers authored by at least one scientist from the corresponding country, the edge width indicates the number of papers written in the collaboration of authors from the corresponding countries. Only countries with at least 100 network science papers are shown in the figure

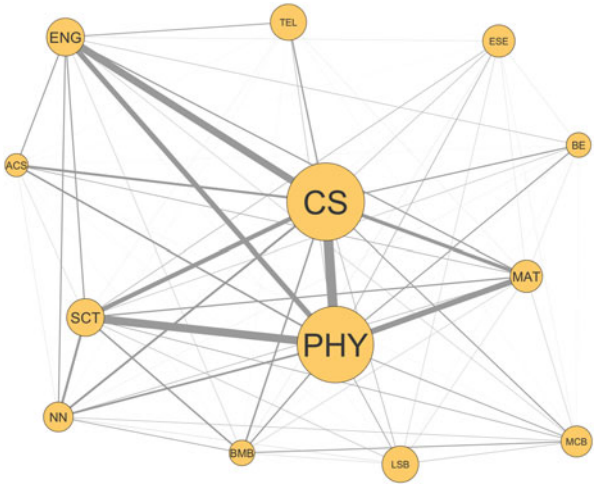


Fig. 18 Network of multidisciplinary collaborations. Only the research areas formed by at least 500 network scientists are shown in the figure. The full names of the research areas can be found in Table 4

6 Conclusion

Two decades ago a new multidisciplinary scientific field was born: network science. In this chapter, we paid tribute to the network science community by investigating the past 20 years of complex network research as seen through the co-authorship network of network scientists. We studied 31,763 network science

papers by extracting the distributions of research areas, journals, and keywords. We identified the most important publication venues and topics in network science and shed light on how they changed over time, we also explored the co-occurrence network of the keywords. Moreover, we constructed and extensively analyzed the co-authorship network of 56,646 network scientists, for example investigating its topological properties, namely its community structure, degree, and centrality distributions. We identified the most central authors of network science as seen through the co-authorship network. We also studied the spatiotemporal changes to provide insights on collaboration patterns. We can conclude that both international and interdisciplinary collaborations are on the increase and the network science community is getting more connected. Furthermore, we compared the centrality measures of authors with well-known scientometric indicators (e.g. citation count and h-index) and found a high correlation.

The present study also has its own limitations. Most importantly, our definitions of a network science paper and a network scientist are quite arbitrary but we believe our chosen notions are good proxies for the purpose of this study that is also supported by the distribution of the keywords of the examined papers. Additionally, the data set itself is not consistent due to different naming conventions that we aimed to resolve, furthermore, we cannot distinguish between different scientists with the same name. However, the error introduced by these problems is negligible.

After investigating the publication and collaboration patterns of network science and observing an increasing impact of complex networks, we are convinced that the next 20 years will produce at least as many fruitful scientific collaborations and outstanding discoveries in network science as the last two decades.

Acknowledgments We thank the anonymous reviewers for their observations and comment. The research reported in this chapter and carried out at the Budapest University of Technology and Economics has been supported by the National Research Development and Innovation Fund based on the charter of bolster issued by the National Research Development and Innovation Office under the auspices of the Ministry for Innovation and Technology. The research of Roland Molontay was partially supported by the NKFIH K123782 research grant.

References

1. Barabás, B., Fülöp, O., & Molontay, R. (2019). The co-authorship network and scientific impact of László Lovász. *Journal of Combinatorial Mathematics and Combinatorial Computing, 108*, 187–192.
2. Barabás, B., Fülöp, O., Molontay, R., & Pályi, G. (2017). Impact of the discovery of fluorous biphasic systems on chemistry: A statistical and network analysis. *ACS Sustainable Chemistry & Engineering, 5*(9), 8108–8118.
3. Barabási, A. (2019). Twenty years of network science: From structure to control. In *APS March Meeting Abstracts* (Vol. 2019, pp. S53–001).
4. Barabási, A. L. (2003). Linked: The new science of networks. *American Journal of Physics*.
5. Barabási, A. L., & Albert, R.: Emergence of scaling in random networks. *Science, 286*(5439), 509–512 (1999)

6. Barabási, A. L., Jeong, H., Néda, Z., Ravasz, E., Schubert, A., & Vicsek, T. (2002). Evolution of the social network of scientific collaborations. *Physica A: Statistical Mechanics and Its Applications, 311*(3–4), 590–614.
7. Barabási, A. L., et al. (2016). *Network science*. Cambridge: Cambridge University Press.
8. Breugelmans, J. G., Roberge, G., Tippett, C., Durning, M., Struck, D. B., & Makanga, M. M. (2018). Scientific impact increases when researchers publish in open access and international collaboration: A bibliometric analysis on poverty-related disease papers. *PloS one, 13*(9), e0203156.
9. Choobdar, S., Ahsen, M. E., Crawford, J., Tomasoni, M., Fang, T., Lamparter, D., Lin, J., Hescott, B., Hu, X., Mercer, J., et al. (2019). Assessment of network module identification across complex diseases. *Nature Methods, 16*(9), 843–852.
10. Clauset, A., Newman, M. E., & Moore, C. (2004). Finding community structure in very large networks. *Physical Review E, 70*(6), 066111.
11. Council, N. R., et al. (2005). Network science committee on network science for future army applications.
12. Fortunato, S., Bergstrom, C. T., Börner, K., Evans, J. A., Helbing, D., Milojević, S., Petersen, A. M., Radicchi, F., Sinatra, R., Uzzi, B., et al. (2018). Science of science. *Science, 359*(6379), eaao0185.
13. Girvan, M., & Newman, M. E. (2002). Community structure in social and biological networks. *Proceedings of the National Academy of Sciences, 99*(12), 7821–7826.
14. International Organization for Standardization. (2020). Officially assigned ISO 3166-1 alpha-3 codes. https://www.iso.org/obp/ui/.
15. Kastrin, A., & Hristovski, D. (2019). Disentangling the evolution of medline bibliographic database: A complex network perspective. *Journal of Biomedical Informatics, 89*, 101–113.
16. Kocarev, L., & In, V. (2010). Network science: A new paradigm shift. *IEEE Network 24*(6), 6–9.
17. Kumar, S. (2015). Co-authorship networks: A review of the literature. *Aslib Journal of Information Management, 67*(1), 55–73.
18. Lella, E., Amoroso, N., Lombardi, A., Maggipinto, T., Tangaro, S., Bellotti, R., Initiative, A. D. N. (2018). Communicability disruption in Alzheimer's disease connectivity networks. *Journal of Complex Networks, 7*(1), 83–100.
19. Leonidou, L. C., Katsikeas, C. S., & Coudounaris, D. N. (2010). Five decades of business research into exporting: A bibliographic analysis. *Journal of International Management, 16*(1), 78–91.
20. Li, H., An, H., Wang, Y., Huang, J., & Gao, X. (2016). Evolutionary features of academic articles co-keyword network and keywords co-occurrence network: Based on two-mode affiliation network. *Physica A: Statistical Mechanics and Its Applications, 450*, 657–669.
21. Molontay, R., & Nagy, M. (2019). Two Decades of Network Science as seen through the co-authorship network of network scientists. In *International Conference on Advances in Social Networks Analysis and Mining, ASONAM*. IEEE/ACM.
22. Nagy, M., & Molontay, R. (2020). Twenty years of network science – Supplementary material. https://github.com/marcessz/Twenty-Years-of-Network-Science.
23. Newman, M. (2018). *Networks*. Oxford: Oxford University Press.
24. Newman, M. E. (2001). The structure of scientific collaboration networks. *Proceedings of the National Academy of Sciences, 98*(2), 404–409.
25. Newman, M. E. (2004). Coauthorship networks and patterns of scientific collaboration. *Proceedings of the National Academy of Sciences, 101*(suppl 1), 5200–5205.
26. Newman, M. E. (2006). Finding community structure in networks using the eigenvectors of matrices. *Physical Review E, 74*(3), 036104.
27. Newman, M. E., & Girvan, M. (2004). Finding and evaluating community structure in networks. *Physical Review E, 69*(2), 026113.
28. Pawar, R. S., Sobhgol, S., Durand, G. C., Pinnecke, M., Broneske, D., & Saake, G. (2019). Codd's world: Topics and their evolution in the database community publication graph. In *Grundlagen von Datenbanken* (pp. 74–81).

29. Su, H. N., & Lee, P. C. (2010). Mapping knowledge structure by keyword co-occurrence: A first look at journal papers in Technology Foresight. *Scientometrics, 85*(1), 65–79.
30. Tálas, A. (2008). *Connected: The power of six degrees*.
31. Uddin, S., Khan, A., & Baur, L. A. (2015). A framework to explore the knowledge structure of multidisciplinary research fields. *PloS one, 10*(4), e0123537.
32. Van Eck, N., & Waltman, L. (2009). Software survey: VOSviewer, a computer program for bibliometric mapping. *Scientometrics, 84*(2), 523–538.
33. Vespignani, A. (2018). Twenty years of network science. *Nature, 558*, 528–529.
34. Watts, D. J. (2004). *Six degrees: The science of a connected age*. W. W. Norton & Company is based in New York.
35. Watts, D. J., & Strogatz, S. H. (1998). Collective dynamics of "small-world" networks. *Nature, 393*(6684), 440.
36. Xia, F., Wang, W., Bekele, T. M., & Liu, H. (2017). Big scholarly data: A survey. *IEEE Transactions on Big Data, 3*(1), 18–35.
37. Yan, E., & Ding, Y. (2014). Scholarly networks analysis. In *Encyclopedia of Social Network Analysis and Mining* (pp. 1643–1651). New York: Springer.

Impact of Locational Factors on Business Ratings/Reviews: A Yelp and TripAdvisor Study

Abu Saleh Md Tayeen, Abderrahmen Mtibaa, Satyajayant Misra, and Milan Biswal

Abstract The proliferation and the success of crowdsourcing reviews of businesses became a major indicator of the success or failure of establishments. Particularly, Yelp, Trip Advisor, and Zomato are using a social media platform where users can share feedback, scores, photos, and make reservations. Business owners keep a close eye on these crowd-sourced indicators in order to maintain or improve their ratings. While improvements can be related to many factors such as services, food, cleanliness, etc., in this chapter, we focus on the impact of the location of the business on its success based on these crowdsourced ratings. We perform an empirical study to quantify the impact of location characteristic indicators (*parameters*), such as cost of living, housing affordability, or tourism, on the success of restaurants as a business example using two datasets: 2019 Yelp, and TripAdvisor. We first, performed a state wise preliminary experiments to verify the correlation between location parameters and business success. We have verified that some location parameters alone, such as education index, can determine the success of a business with a 0.72 correlation ratio. Next, we propose a clustering method that group similar zip code locations to better estimate the influence of the said location parameters on the success scores.

Keywords Restaurant location success · Distance based social graph · Restaurant business · Yelp · TripAdvisor

A. S. M. Tayeen (✉) · S. Misra · M. Biswal
New Mexico State University, Las Cruces, NM, USA
e-mail: tayeen@nmsu.edu; misra@nmsu.edu; milanb@nmsu.edu

A. Mtibaa
University of Missouri Saint Louis, St. Louis, MO, USA
e-mail: amtibaa@umsl.edu

© The Author(s), under exclusive license to Springer Nature Switzerland AG 2021

M. Çakırtaş, M. K. Ozdemir (eds.), *Big Data and Social Media Analytics*, Lecture Notes in Social Networks, https://doi.org/10.1007/978-3-030-67044-3_2

1 Introduction

We often hear the cliché, used by business experts, that the three most important factors in determining the success of a business are "location; location; location". The location of the business may effect the business's success within the market [15]. Specifically, for restaurant businesses, it is widely speculated that being in a proper location is critically important since it can influence the success or failure of a restaurant in a number of ways, from attracting enough initial customer interest to being convenient to visit. Location is a complex construct encompassing not only physical site but also other facets such as population demographics, climate, accessibility, and competition. All of these variables might have a profound impact on the rise or fall of a restaurant business [14]. Studies have investigated recommendation systems for optimal location to expand an existing business utilizing online customer reviews [18] and check-in patterns [6] or labeling business popularity using customers online social connections and geographical proximity to businesses [5]. However, none of these works have tried to evaluate the impact of location on the success of a business.

Considering the purported importance of location, recently in our preliminary study [16], we performed a quantitative analysis to evaluate the impact of location on the success of restaurants. In [16], we mainly investigated the impact of different categories (*e.g.,* population demographics, tourism potential, access to businesses) of location characteristic parameters on restaurants' success, which in turn, is defined by the combination of a restaurant's online review counts and star ratings. As geographic location granularity, we used zip codes. We measured the location-by-location correlations between the success of specific restaurants (cuisines such as *American*, *Chinese*) and the parameters of its locations grouped by three individual states of U.S.A: Arizona (AZ), North Carolina(NC), and Ohio (OH) gathered from Yelp data released in 2018.

However, correlation analysis of few state-wise grouped zip code locations and some specific types of restaurants will not necessarily provide us with the right insight of a location's effect on restaurant success in general. In addition, only one restaurant dataset is not enough to make a general assessment. Moreover, the number of reviews and star ratings alone are not strong indicators of success of restaurants. There are other factors such as the sentiment of the reviews and the age (the duration a restaurant is open) of a restaurant which must be considered in defining the success metric of restaurants.

Novel Contributions To address the above issues, in this chapter, we extended our preliminary study [16] by performing a new correlation analysis of zip code locations grouped by similar characteristics. In our new experiments, we used an updated dataset from Yelp which includes more restaurants from seven states (including four new states) and a new restaurant dataset prepared from the popular travel website TripAdvisor [1]. We also introduced a new metric to measure the success of restaurants by combining review counts, star ratings, age of the restaurants, and sentiments associated with the reviews.

In this chapter, our contribution is mainly two fold. First, we grouped the zip codes of seven states in two ways: state-wise and cluster-wise where we used an unsupervised machine learning technique to cluster the zip code locations based on the demographic and location-specific features. Then we measured zip code by zip code correlations between the success of restaurants and the parameters of its locations of those two groups. Second, we defined a new metric to measure the success of restaurants and performed the same correlation analysis on the location groups using that metric. We also validated the influence of the location parameters on restaurant success by using a new restaurant dataset from TripAdvisor.

2 Related Work

In this section, we briefly overview different streams of studies related to predicting business reviews and ratings based on business services [8], features [5, 9, 12, 17], and the location of business [6, 18]. Researchers have used features e.g., degree centrality and clustering coefficient, derived from graph model of user rating and business category data in Yelp, to predict user ratings [17]. Using a Yelp dataset, Lu et al. [12] have shown that non-text features, such as review star loss and return guest count, are more significant than text (e.g., unigram, bigram) features to predict the future success (star rating) of restaurants. None of these have investigated the impact of location on business success.

More relevant to our study, Hu et al. [10] used matrix factorization model and evaluated the influence of geographical neighbors to business ratings prediction. Another study showed that time dependent features, such as the number of days since the first and last review, provide the best result to infer future attention (number of reviews) of businesses [9]. Wang et al. [18] proposed a solution to find an optimal location from some candidate locations for expanding a chain restaurant business to a new branch. They found that market attractiveness and competitiveness features which are generated solely from user review text are more accurate than geographic features to predict the number of likely visits to a restaurant in a prospective location. Eravci et al. [6] performed experiments using check-in data for New York from Foursquare and presented two solutions: Bayesian inference-based and collaborative filtering using neighborhood similarity to recommend a set of city neighborhoods as venues to successfully invest in a new specific business category.

In another study [5], researchers proposed an approach to label popular and unpopular businesses based on region-wise popularity metrics. They also demonstrated the influence of local and foreign customers on the popularity of businesses and proposed a model to predict popularity of businesses with an accuracy of 89%. Hegde et al. [8] identified restaurant properties such as “accept credit card”, “ambience” to be the most interesting to customers, found Monday as the most crowded day of the week for restaurants, and explored other restaurant properties (e.g., Wi-Fi, parking lot) that are missing in nearest restaurants to help setup a new restaurant business.

In our prior study [16], we performed correlation analysis between the success of few cuisine restaurants and the characteristic parameters of the locations those are situated in. We observed that location parameters such as life style and education index are strongly correlated with restaurant's success with correlation value closer to 0.5. In addition, we investigated how a cuisine restaurant's adjacency to neighboring locations such as landmarks or other restaurants can impact its success. We built city-wise restaurant vicinity graphs for different states and plotted the average success of cuisine restaurants as a function of maximum proximity to a landmark, minimum number of nearby fast-food franchises and same category restaurant density. We found from the city of Phoenix that the success of restaurants can reach up to 72% when its located close to city landmarks. We also observed that proximity to similar restaurants can increase restaurants' success up to 53.62% as verified in the city of Phoenix. Also, in the city of Cleveland, we noticed that cuisine restaurants surrounded by many fast-food franchises can be at most 16% successful.

The studies mentioned above mostly focus on recommending venues to investors for setting up new businesses utilizing multiple features including the location. However, in this chapter we mainly aim at getting a generalized assessment of a location's impact on the success of a business or its failure.

3 Methodology and Datasets

In this section, we briefly describe our methodology, define our restaurant success evaluation metric, and introduce the datasets used in our study.

3.1 Methodology

Our methodology to investigate the effect of location characteristic parameters on restaurant's success involves two different experiments: one with locations of all restaurants grouped by states and another with locations of all restaurants clustered by similar characteristics. For both of these experiments, we used a metric to assess the success of a restaurant. We interpret success of a restaurant in two ways: first as a function of its star rating and number of reviews received online and second as a function of its online star rating, review count, duration of business, and sentiments associated with reviews. We defined the metrics in Sect. 3.2. We make use of two Restaurant datasets described in detail in Sect. 3.3 to compute the values of these metrics for many restaurants.

To empirically quantify the impact of a given location parameter on the success of a restaurant, we find a zip-by-zip[1] correlation between the restaurant success and the

[1] Locations are represented by their corresponding zip codes.

location characteristic parameters. We are mainly interested in the inter-relationship of restaurant success and its location parameters rather than the corresponding cause and effect relationship. We first define parameter s illustrated in Sect. 4.1 to evaluate location characteristics. Second, we make use of a Location dataset, *LD* outlined in Sect. 3.4 to calculate the parameter values for each zip code. Finally, we utilize the values of a location parameter for each zip and restaurants' average success values for those zips to compute standard correlations.

We conduct our first experiment per state for all kinds of restaurants and every location parameter separately. However, the location features for a state might not be necessarily similar throughout and thus the state wise aggregation may dilute the inference. Therefore, we conduct a second experiment by using machine learning to cluster zip codes that share similar location characteristics. The zip-by-zip correlation between the location parameters and business success was then performed for each cluster.

3.2 Restaurant Success Metric

Online user reviews and star ratings are crucial factors determining the reputation or success of restaurants. While business success metrics may include income, service evaluation, awards, etc., we focus solely on the social media success, limited to reviews and ratings. Because customer's decision to visit a restaurant heavily depends on how many positive reviews and star rating that restaurant has.

Following our restaurant business success definition, we will quantify the success of a given restaurant based on both: (i) its star rating and (ii) the users review counts it received. We, therefore, define $SB_1(r)$, the success metric of a given restaurant business r using the following equation:

$$SB_1(r) = S(r)^{\log(RC(r))}, \tag{1}$$

where $S(r)$ is r's average star rating and $RC(r)$ is the number of reviews users gave to restaurant r. For each restaurant r, we extract $S(r)$ and $RC(r)$ values from our restaurant datasets and compute the restaurant success metric ($SB_1(r)$).

The number of reviews alone is not a reliable indicator of success for a restaurant. Each review could be positive, negative or neutral, which should be taken into consideration while assessing business success. To achieve this, we used machine learning, to quantify the sentiment associated with each review. We used the *CoreNLP* [13] tool from Stanford University to perform the sentiment analysis. This tool assigns to each sentence a sentiment value ranging from zero to four, where zero means the sentence is very negative and four means the sentence is extremely positive. Since each review may contain more than one sentences, we took the average of the sentiment values of all the sentences in the review to get a sentiment score for that review. Moreover, to compute the sentiment score $SC(r)$

for a restaurant r, we average the sentiment scores corresponding to all the reviews associated with that restaurant.

Another factor while accessing the success of a restaurant is its *age*. For instance, historic restaurants running for generations could not be compared on the same scale of online reviews with a relatively new restaurant. In addition, some restaurants might have been there for decades catering the local population, while lying low in the internet.

Considering these factors, we modified the success metric to include the sentiment score and age of the restaurant. The new metric is expressed as follows.

$$SB_2(r) = S(r)^{\log\left(\frac{RC(r)\times SC(r)}{N_{age}(r)}\right)}, \tag{2}$$

where, $SC(r)$ is the sentiment score of restaurant r and $N_{age}(r)$ is the age of restaurant r in months. We computed the age of restaurant r by taking the difference between current date and the date on which its first review was recorded.

3.3 Restaurant Dataset

Our restaurant datasets, RD_{Yp19} and RD_{TpAdv} consists of data collected from Yelp Dataset Challenge 2019 [2], and TripAdvisor website.[2] Each business in these data has unique identifier, name, address, city, state, average rating, and review count. In Yelp data, as attribute information, a business has categories such as 'Restaurant'. Each business in the Yelp data also has review text associated with it.

3.3.1 Yelp-2019 Dataset

The Yelp data released in 2019 is essentially a super set of the 2018 version of Yelp data we used in [16]. This version of Yelp data contains information of total 192,609 businesses and 6,685,900 reviews. We compiled our restaurant dataset, RD_{Yp19} with restaurants from *seven* different states of the United States: Arizona (AZ), North Carolina (NC), Ohio (OH), Pennsylvania (PA), Nevada (NV), Wisconsin (WI), and Illinois (IL) which are among the top seven states with large number of businesses in the 2019 version of Yelp data. We did not perform a cuisine wise segregation of this dataset as we did earlier in [16], to generalize the influence of location on the success of any restaurant, irrespective of their cuisine. The number of zip codes and restaurants per state in the RD_{Yp19} dataset is summarized in Table 1.

[2] https://www.tripadvisor.com/

Table 1 Number of zip codes and restaurants per state in RD_{Yp19} and RD_{TpAdv} datasets

State	Zip codes	Restaurants (RD_{Yp19})	Restaurants (RD_{TpAdv})
AZ	123	11,332	10,667
NC	57	4081	2486
OH	111	5498	1929
PA	109	3922	1751
NV	61	7651	8126
WI	31	1673	644
IL	20	667	15
Total	512	34,824	25,618

3.3.2 TripAdvisor Dataset

To improve the generalization we compiled another restaurant dataset by crawling the popular website TripAdvisor. We compiled information (e.g. average rating, and review count) of restaurants from the same seven states mentioned earlier during the period from Nov 2019 to Dec 2019. We did not crawl any reviews of the restaurants from TripAdvisor website. The number of restaurants per state in the TripAdvisor dataset, RD_{TpAdv} is summarized in Table 1.

3.4 Location Dataset

We constructed the Location dataset, *LD* by crawling the City-Data website[3] which contains demographics and tourism related information pertaining to United States cities and regions. Our *LD* dataset includes location-based information such as race, income, education, housing, transportation, crime, etc. Data was gathered for different zip code locations from the **seven** states mentioned earlier. The number of zip codes gathered per state are shown in Table 1. The *LD* contains raw location related data which independently may not define causal relationship with the success of businesses. And even if they do, it is difficult to encapsulate business intelligence. Therefore, we define location parameters considering the domain verbosity, that can be efficiently interpreted, in the following section.

4 Effect of Location Parameters on Restaurant Success

In this section, we will introduce several parameters to evaluate location of a restaurant and use the *LD* dataset to calculate the values of these parameters. We conduct two experiments. In the first, we discuss result of correlations between the

[3] http://www.city-data.com/

values of all kinds of restaurants' success and parameters of locations (zip codes) grouped by individual states. In the second, we form clusters of zip codes sharing similar location based features. Then, for each cluster, we find the correlation between restaurants' success and location parameters.

4.1 Location Characteristic Parameters

Different categories of location characteristics such as population demographics, tourism, accessibility might have a profound impact on the rise or fall of a restaurant business [14]. Note that our lowest location granularity is the zip code as per our Location dataset. However, we argue that the methodology taken in this chapter applies for a finer granularity, such as streets or full addresses.

Our analysis focus on three main location parameter categories, namely (i) the *living standard*, (ii) the *tourism significance*, and (iii) the *business convenience*, of a given location. Next we introduce these three categories in more detail and propose few location parameters per category.

4.1.1 Living Standard

The living standard of a location can be determined by looking at the job prospects, the trend in education, the cost of living and the housing affordability of that location. Our intuition is that the socio-economic status of the people who are the target customers of a restaurant in an area, will have a direct impact on a restaurant's success. Because when people have better education, have good jobs to earn enough money, and can afford to reside in a low cost area to maintain a certain level of living, they tend to prefer eating at restaurants and are more likely to write positive/negative reviews. Hence, we present four living standard parameters, namely *Education Index*, *Housing Affordability Index*, *Cost of Living Index*, and *Life Style*.

Education Index To measure job prospects and education standard in a zip code location z, we multiply the weighted average of the percentage of people educated in different levels such as high school, graduate, by the percentage of people employed in z. We name this parameter EI^z as defined by the following equation.

$$EI^z = (u_1 \times G^z + u_2 \times B^z + u_3 \times H^z) \times E^z \tag{3}$$

where, u_1, u_2, and u_3 are weights such that $\sum u_i = 1$; H^z, B^z, and G^z are the percentage of people educated up to High school or higher, educated up to Bachelor's degree or higher, and educated up to Graduate or professional degree respectively; and E^z is the percentage of people employed. We choose $u_i \geq u_{i+1}$ as

we assume that the higher the degree of education of a person, the higher his/her income thus the restaurant affordability. In our experiments, $u_1 = 0.5, u_2 = 0.3,$ and $u_3 = 0.2$.

Housing Affordability Index It is a parameter used by the US National Association of Realtors (NAR) to assess a typical middle income family's qualification for a mortgage loan on a median-priced home [3]. This parameter is a ratio between the median household income and the annual qualifying income needed to own a median-priced home. To calculate the qualifying income, NAR assumes that the homeowner uses no more than 25 percent of monthly household income for the mortgage payments. We use this parameter as an indicator for the living standard in a zip code location, z. We denote this parameter by HAI^z as defined by the following equation.

$$HAI^z = \frac{M^z}{MP^z \times 12 \times 4} \times 100, \tag{4}$$

where M^z is median household income, and MP^z is median monthly payment for housing units with a mortgage.

Cost of Living Index This index [4] is often used to quantify the expenses to live in a given area. We use Cost of Living Index, CLI^z as another parameter to estimate the living standard of a location z.

Life Style We also combine the parameters mentioned above to produce a new metric and name it *Life Style*, LS^z. It is defined by the following equation.

$$LS^z = \frac{EI^z \times HAI^z}{CLI^z}, \tag{5}$$

4.1.2 Tourism Significance

Tourism significance expresses how compelling a location is in terms of attracting tourists. Tourists often visit areas such as outdoor scenery (e.g., landscape, wildlife), historic or cultural venues, and recreation facilities. Next, we present two tourism related parameters, namely *Tourism Attraction Density* and *Neighboring Tourism Attractions*.

Tourism Attraction Density We argue that the number of touristic sights or attractions in a given location can be a good indicator for its tourism significance. Thus, we define the tourism attraction density parameter of a location z as:

$$TD^z = \frac{e^{TC^z}}{Area(z)}, \tag{6}$$

where, $Area(z)$ is the land area of location z, and TC^z is the weighted sum of touristic place counts of location z, defined by the following.

$$TC^z = v_1 \times (NP^z + BE^z) + v_2 \times TL^z + v_3 \times RV^z + v_4 \times LR^z + v_5 \times PK^z, \quad (7)$$

where NP^z is the number of National Parks, BE^z is the number of Beaches, TL^z is the number of Tourist Locations, RV^z is the number of Rivers, Streams or Creeks, LR^z is the number of Lakes and Reservoirs, PK^z is the number of Parks; v_i are weights such that $\sum v_i = 1$. We choose $v_i \geq v_{i+1}$ to give more weight to national parks and beaches compared to other tourism attractions such as rivers. In our experiments, $v_1 = 0.3$, $v_2 = 0.2$, and $v_3 = v_4 = v_5 = 0.1$.

Neighboring Tourism Attractions We argue that tourists may visit touristic attractions in a given location z and walk to neighboring locations nz for a meal. Thus, measuring the tourism attraction density only in the current location z may not be sufficient and can be augmented by looking at neighboring location, nz's attraction density as well.

$$NT^z = \alpha TD^z + (1 - \alpha)\frac{\sum_{nz} TD^{nz}}{t_z}, \quad (8)$$

where, TD^{nz} is the tourism attraction density of nz; t_z is the total number of neighboring locations for z and α is the influence factor between z and it's neighboring locations such that $\alpha > 0.5$. In our experiments, we let $\alpha = 0.8$.

4.1.3 Business Convenience

We refer to business convenience of a given location z, as the relative ease of access to z. While there maybe multiple factors to determine the business convenience of a given restaurant r, we consider only two: the presence of shopping malls and the availability of public transportation. Next, we present two business convenience related parameters, namely *Business Accessibility* and *Neighboring Business Accessibility*.

Business Accessibility To measure the business accessibility of z, we define the BA^z parameter by the following equation.

$$BA^z = \frac{e^{NS} \times TA^z}{Area(z)}, \quad (9)$$

where NS is the number of shopping centers, $Area(z)$ is the land area of z, and TA^z is the weighted average of the percentage of people using different transportation in

location z, defined by the following.

$$TA^z = w_1 \times BR^z + w_2 \times CT^z + w_3 \times CM^z + w_4 \times WB^z, \tag{10}$$

where BR^z is the percentage of people using subway or bus or railroad, CT^z is the percentage of people using carpool or taxi, CM^z is the percentage of people using car or motorcycle, WB^z is the percentage of people who walks or use bicycle; w_i are weights such that $\sum w_i = 1$. We choose $w_i \geq w_{i+1}$ to give more weight to public transportation convenience compared to other means that can be costly and/or inconvenient (e.g., finding parking). In our experiments, $w_1 = 0.7$, and$w_2 = w_3 = w_4 = 0.1$.

Neighboring Business Accessibility We argue that people may visit neighboring shopping malls while shopping from a center in a given location z. Thus, we can augment the business accessibility of z with the business accessibility of its neighboring location, nz using the following equation.

$$NB^z = \beta BA^z + (1 - \beta)\frac{\sum_{nz} BA^{nz}}{t_z}, \tag{11}$$

where, BA^z is the business accessibility of z, BA^{nz} is the business accessibility of the neighboring location nz of z, t_z is the total number of neighboring locations for z and β is the influence factor between z and it's neighboring locations such that $\beta > 0.5$. We choose $\beta = 0.7$.

4.1.4 Combined Parameters

In addition to the individual location parameters, we attempted to form a combined parameter that aims to merge two or more salient parameters. As some of the defined location parameters are derived from one or more common independent parameters, we chose *LS* and *NB* as two most important parameters. To estimate the relative importance of these two parameters in the combined parameter, we performed a linear regression against the restaurant success metric values, for the *two* datasets. To ensure compatibility among datasets, we normalized all the individual location parameters and business successes in a particular dataset. The regression coefficients were then normalized to find the proportional contribution of each of these two parameters. We performed regression twice for each success metrics (SB_1 and SB_2), independently. Both the regression led to identical results, and the P-values for the regression coefficients were less than 0.05, indicating statistical significance. The combined parameter is expressed as:

$$CA^z = 0.4LS^z + 0.6NB^z \tag{12}$$

4.2 Correlation Metrics

Our objective is to measure the degree of statistical association between the restaurants' success values and the parameter values of the areas where these restaurants are located. Regression analysis and the degree of correlation are the two prominent choice to evaluate the statistical association. Linear regression is preferred when the objective is to deduce how independent variables affect each other to devise causal implications. However, the location parameters considered in this chapter are derived variables which encapsulate the domain countenance. Our objective is to primarily determine the degree of interrelation between the location parameters and the business success, and our focus is on the deduction rather than prediction. In addition, correlation effectively quantifies the strength and direction of the relationship between variables. Therefore, we preferred to use the correlation metric for the assessment.

To accomplish that, we first create two vectors: $X = (x_1, x_2, \cdots, x_n)$ and $Y = (y_1, y_2, \cdots, y_n)$, where x_i represents the value of a location parameter in the i-th zip code, y_i may represent the value of the success metric of restaurant located in the i-th zip code, and n, the size of both vectors X and Y, is the number of zip codes in a given state or cluster. Note that, since multiple restaurant may be co-located in the same zip code location, we consider the average restaurant success in location i (i.e., which represent a given zip code z_i) as y_i.

In our experiments, we are interested in the ranks of the data points rather than the value of the data points. Therefore, in our experiments we adopt two rank-order based correlation metrics, Spearman's correlation coefficient [7] and Kendall's correlation coefficient [11] to measure the correlation between location parameter and restaurant success metric values.

4.2.1 Spearman's Correlation

The Spearman's correlation coefficient can detect the strength and direction of the monotonic relationship between two variables. The idea is to first rank the data points of each variable separately. Ranking is obtained by assigning a rank of 1 to the lowest value, 2 to the next lowest and so on. Then for each pair (x_i, y_i) the difference between ranks of x_i and y_i are calculated. If the variables are correlated, then the sum of the squared difference between ranks of the data points will be small. The magnitude of the sum is related to the significance of the correlation. For example, if we consider LS^z parameter as X variable and success metric SB_1 as Y variable, then from the correlation coefficient of the values of X and Y, we can infer that how strongly living standard of location affects the success of restaurants. The Spearman's correlation coefficient, ρ is computed according to the following equation:

$$\rho = 1 - \frac{6 - \sum^{n} {d_i}^2}{n(n^2 - 1)}, \tag{13}$$

where d_i is the difference between ranks for each (x_i, y_i) data pair and n is the number of data pairs.

4.2.2 Kendall's Correlation

Unlike Spearman's correlation coefficient, Kendall's correlation does not take into account the difference between ranks, only directional agreement. The definition of Kendall's correlation relies on the notion of concordance. Two pairs of observations (x_i, y_i) and (x_j, y_j) where $i < j$, are called concordant pairs if the ranks (the sort order by x and y) for both elements agree. That is, if both $x_i > x_j$ and $y_i > y_j$; or if both $x_i < x_j$ and $y_i < y_j$. The pairs are called discordant, if $x_i > x_j$ and $y_i < y_j$; or if $x_i < x_j$ and $y_i > y_j$. The equation to compute the Kendall's correlation coefficient, τ, is as follows:

$$\tau = \frac{n_c - n_d}{\sqrt{(n_0 - n_1)(n_0 - n_2)}}, \tag{14}$$

where, $n_0 = n(n-1)/2$, n is the size of the X and Y, $n_1 = \sum_i t_i(t_i - 1)/2$, $n_2 = \sum_j u_j(u_j - 1)/2$, t_i is the number of tied values in the i-th group of ties for X variable, u_j is the number of tied values in the j-th group of ties for Y variable, n_c is the number of concordant pairs, and n_d is the number of discordant pairs.

The coefficient value varies between -1 and 1. A coefficient of 1 denotes that the rankings of the two variables are in perfect agreement. A coefficient equal to -1 signifies that one ranking is the opposite of the other. If the coefficient is 0, then X and Y are independent variables. To identify the statistical significance of the correlation coefficients, we computed the P-value. The null hypothesis for P-value computation considers that the correlation coefficient is zero, or there is no relationship between the variables. The null hypothesis is rejected if the P-value is less than 0.05, and indicates statistical significance [19]. The complete reliance on the P-value have competing views in the statistical community [20]. Particularly, the P-value is generally derived from an arbitrarily set and has no resemblance to the effect of one or more variables on another. Therefore, although we report all the correlation results in the summary tables, the values that pass the P-value test are underlined.

4.3 Correlation Results

We created X and Y vectors for each state and each cluster using all its zip codes that exist in our *LD* dataset. We removed the zip codes that did not have any restaurants

based on our restaurant datasets. We computed both Spearman's and Kendall's correlation coefficient by choosing values of each location parameter separately to build X.

4.3.1 State-Wise Correlation Using All Restaurants

In this analysis, we first use the RD_{Yp19} dataset with seven states. We show the correlation coefficient between values of each location parameter for all zip codes in each state and average values of success metric, SB_1 for all restaurants located in those zip codes in Table 2.

From Table 2, we find that among the three different categories, both the parameters corresponding to the living standard category yielded in the highest average state-wise correlation coefficients. As expected, we observe that the Spearman's correlation values are slightly greater than the Kendall's. The average Spearman's coefficient for Education index (EI^z) and Life style (LS^z) are $\rho = 0.56$ and $\rho = 0.48$, respectively. The Spearman's coefficient are more sensitive to error and discrepancies in the data, and their values being high indicate prominent statistical relationship. The average Kendall's correlations for both these parameters (EI^z and LS^z) are $\tau = 0.42$ and $\tau = 0.33$, respectively, indicating moderately concordant relationship. Overall, both the correlation coefficients lead to the inference that the living standard parameters positively affect the success of the business. This is further reinforced by the statistical significance indicated by the lower P-value (<0.05), for all the states except two states (WI and IL). We conjecture that this is due to the small number of zip codes for those states, available in our datasets.

The combined parameters did not indicate most prominent correlation for any of the states. Its lower relative importance indicate that a set of fixed proportional contribution of the location variables taken together may not necessarily improve their coupling with the business success.

The state-wise evaluation indicate that, living standard is the most prominent factor affecting business success. The highest value of correlation coefficients (EI^z : $\rho = 0.72, \tau = 0.54$, and LS^z: $\rho = 0.67, \tau = 0.47$), were obtained corresponding to living standard parameters of North Carolina. The same conclusion can be extended for AZ, OH, PA, NV, and IL, which have the highest correlation values for living standard among other parameters. However, for WI, tourism significance parameters (TD^z and NT^z) achieved maximum $\rho = 0.55, \tau = 0.39$ among all the other parameters.

In Table 3, we present the correlation coefficients computed state-wise using business success metric, SB_2. Similar to the results in Table 2, we observe in Table 3 that living standard category parameters also have the highest average $\rho = 0.60, \tau = 0.45$ (EI^z) and $\rho = 0.49, \tau = 0.35$ (LS^z). This implies that the inclusion of semantic review information and age of the restaurants into the success metric did not significantly affect the correlation coefficient values. Thus, the overall analysis of the results indicate that living standard of a location contributes most on the success of all kinds of restaurants.

Table 2 Correlation results of all the restaurants using SB_1, grouped by state for seven states in RD_{Yp19} dataset. Column-wise maximum values are bold faced and the statistically significant values are italicized

Category	Param.	AZ		NC		OH		PA		NV		WI		IL	
		ρ	τ	ρ	τ	ρ	τ	ρ	τ	ρ	τ	ρ	τ	ρ	τ
Living standard	EI^z	***0.56***	***0.40***	***0.72***	***0.54***	***0.54***	***0.38***	***0.60***	***0.42***	***0.44***	***0.31***	*0.53*	***0.43***	***0.53***	***0.45***
	LS^z	*0.45*	*0.31*	*0.67*	*0.47*	*0.42*	*0.30*	*0.49*	*0.34*	*0.36*	*0.25*	0.30	0.23	0.32	0.24
Tourism significance	TD^z	0.15	0.10	*0.38*	*0.27*	*0.25*	*0.17*	0.09	0.08	−0.03	−0.03	***0.55***	*0.39*	0.06	0.05
	NT^z	0.15	0.10	*0.38*	*0.27*	*0.25*	*0.17*	0.09	0.08	−0.03	−0.03	***0.55***	*0.39*	0.06	0.05
Business convenience	BA^z	*0.19*	*0.13*	0.22	0.15	0.04	0.03	0.09	0.07	−0.02	−0.02	*0.45*	*0.31*	−0.11	−0.09
	NB^z	*0.19*	*0.13*	0.22	0.15	0.04	0.03	0.09	0.07	−0.02	−0.02	*0.45*	*0.31*	−0.11	−0.09
Combined	CA^z	*0.23*	*0.16*	*0.37*	*0.27*	0.04	0.03	*0.41*	*0.29*	0.06	0.04	*0.51*	*0.38*	0.38	0.30

Table 3 Correlation results of all the restaurants using SB_2, grouped by state for seven states in RD_{Yp19} dataset. Column-wise maximum values are bold faced and the statistically significant values are italicized

Category	Param.	AZ		NC		OH		PA		NV		WI		IL	
		ρ	τ	ρ	τ	ρ	τ	ρ	τ	ρ	τ	ρ	τ	ρ	τ
Living standard	EI^z	***0.58***	***0.42***	***0.70***	***0.52***	***0.60***	***0.43***	***0.59***	***0.42***	***0.50***	***0.36***	***0.66***	***0.52***	***0.59***	***0.47***
	LS^z	*0.51*	*0.36*	*0.66*	*0.47*	*0.48*	*0.35*	*0.51*	*0.36*	*0.43*	*0.31*	*0.47*	*0.33*	0.38	0.28
Tourism significance	TD^z	0.05	0.03	*0.38*	*0.27*	*0.19*	0.12	−0.01	0.01	−0.05	−0.05	*0.56*	*0.40*	0.12	0.08
	NT^z	0.05	0.03	*0.38*	*0.27*	*0.19*	0.12	−0.01	0.01	−0.05	−0.05	*0.56*	*0.40*	0.12	0.08
Business convenience	BA^z	0.08	0.05	0.13	0.09	0.13	0.09	0.09	0.07	−0.04	−0.05	*0.47*	*0.32*	0.05	0.00
	NB^z	0.08	0.05	0.13	0.09	0.13	0.09	0.09	0.07	−0.04	−0.05	*0.47*	*0.32*	0.05	0.00
Combined	CA^z	0.14	0.09	*0.33*	*0.23*	0.13	0.09	*0.48*	*0.34*	0.03	0.02	*0.64*	*0.47*	*0.54*	*0.41*

We did not have any ground truth available to validate the definitive inference of these parameters on the business success. Therefore, we performed the same analysis over a restaurant dataset from a different source such as TripAdvisor, to make a comparative assessment. Table 4 summarizes the correlation coefficients for location parameter values and business success metric, SB_1 values using TripAdvisor dataset RD_{TpAdv}.

In Table 4, we notice that the living standard parameters have again obtained the highest average $\rho = 0.35, \tau = 0.24$ (EI^z) and $\rho = 0.30, \tau = 0.21$ (LS^z). Even though the correlation coefficient values of location parameters and SB_1 values are lower for RD_{TpAdv} dataset compared to the values in Table 2, the results are consistent with the result we found in Table 2. We speculate that the lower correlation values produced are due to less number of restaurants per state in the RD_{TpAdv} dataset compared to the RD_{Yp19} dataset. Thus, our assessment that living standard influences the most on the success of restaurants is validated.

4.3.2 Cluster-Wise Correlation Using All Restaurants

In the earlier experiments, we computed the correlation values between success of restaurants and parameters of zip code locations that were grouped by state. However, generalizing a state-wise study might shadow some of the inherent insights, as different regions of a state may exhibit different demographic and location specific profile. To address this issue, we used a data driven approach rooted in unsupervised machine learning, that creates clusters of zip codes based on demographic and location specific features. The rational behind this approach is that locations which have similar features are expected to show similar business success trend, if location is a key factor for business success.

Clustering algorithm: The k-means algorithm is one of the most profound clustering method that partitions the data into multiple voronoi cells. Each cell or cluster contain data points which exhibit closely related properties. These clusters were formed by minimization of the mean Euclidean distances of all the nearby data points from the center of the cluster. The choice of number of clusters is a critical factor that affects the efficacy of clustering. To determine the appropriate number of clusters, we used the elbow method. Figure 1 shows the plot of weighted error versus the number of clusters. This looks like an elbow and the optimal number of clusters is the point at which the weighted cumulative sum of squared error (WCSSE) do not decrease significantly with increase in the number of clusters. The WCSSE is calculated as:

$$WCSSE = \sum_{i=1}^{n} \sum_{j=1}^{k} w^{(i,j)} ||\mathbf{x}^i - \boldsymbol{\mu}^j||_2^2, \text{ where}$$

$\mathbf{x}^i$, is the data vector comprising of location features for ith zip code, $\boldsymbol{\mu}^j$ is the centroid for cluster j, and

Table 4 Correlation results of all the restaurants using SB_1, grouped by state for seven states in RD_{TpAdv} dataset. Column-wise maximum values are bold faced and the statistically significant values are italicized

Category	Param.	AZ		NC		OH		PA		NV		WI		IL	
		ρ	τ	ρ	τ	ρ	τ	ρ	τ	ρ	τ	ρ	τ	ρ	τ
Living	EI^z	***0.54***	***0.37***	***0.41***	***0.28***	***0.40***	***0.28***	**0.15**	**0.10**	**0.20**	**0.13**	*0.29*	*0.20*	*0.46*	*0.32*
standard	LS^z	*0.40*	*0.27*	*0.30*	*0.20*	*0.40*	*0.28*	0.07	0.05	0.08	0.06	***0.33***	***0.23***	***0.49***	***0.36***
Tourism	TD^z	−0.02	−0.02	0.15	0.10	−0.05	−0.03	−0.03	−0.02	−0.01	−0.01	0.02	0.02	−0.00	−0.00
significance	NT^z	−0.02	−0.02	0.15	0.10	−0.05	−0.03	−0.03	−0.02	0.00	−0.00	0.04	0.03	−0.00	−0.00
Business	BA^z	0.10	0.07	0.07	0.05	0.01	0.01	0.04	0.02	−0.01	−0.00	0.03	0.02	*0.19*	*0.13*
convenience	NB^z	0.10	0.07	0.07	0.05	0.01	0.01	0.04	0.02	0.01	0.01	0.05	0.04	*0.19*	*0.13*
Combined	CA^z	0.15	0.10	0.12	0.08	0.04	0.03	0.14	0.09	0.12	0.08	0.08	0.05	*0.30*	*0.21*

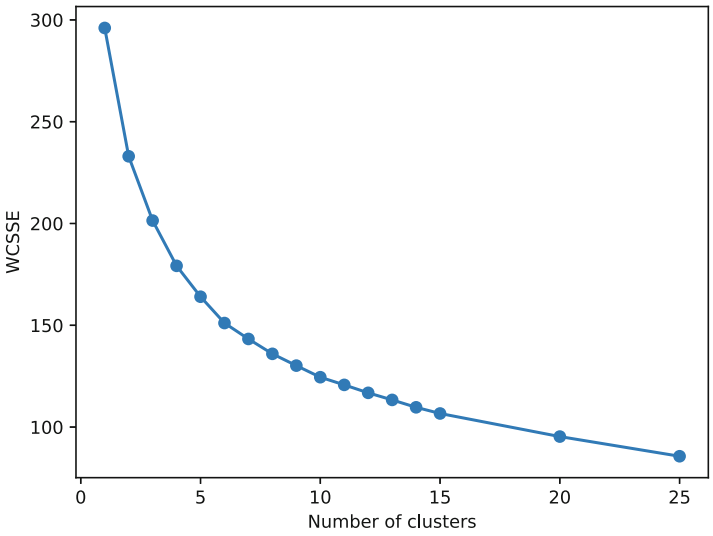


Fig. 1 Elbow method plot showing the reduction of steady state error as a function of the number of clusters

Table 5 Number of zip codes and restaurants for RD_{Yp19} and RD_{TpAdv} datasets in each location cluster

ClusterID	Zip codes	Restaurants (RD_{Yp19})	Restaurants (RD_{TpAdv})
1	135	8405	4640
2	103	13,070	12,275
3	181	6824	5158
4	40	4001	2399
5	53	2524	1146

$$w^{(i,j)} = \begin{cases} 1, & \text{if sample } \mathbf{x}^i \text{ is in cluster } j \\ 0, & \text{otherwise} \end{cases}$$

For our dataset, *five* clusters appeared to be a descent choice. Table 5 lists the number of zip-codes and the total number of restaurants in each cluster.

Cluster-wise correlation: The correlation coefficients for the five clusters with the two business success measures, SB_1 and SB_2 are summarized in Tables 6 and 7. From Table 6, we find that the average value of correlation coefficients across all the clusters, is maximum for living standard parameter EI^z with $\rho = 0.35$, $\tau = 0.23$; and for LS^z $\rho = 0.30$, $\tau = 0.20$, when SB_1 metric is used. For SB_2 metric, the maximum average correlation values were again observed for living standard parameters EI^z with $\rho = 0.32, \tau = 0.21$; and for LS^z with $\rho = 0.28, \tau = 0.18$ in Table 7. Unlike state-wise correlation, the correlation coefficient values observed in Table 6 for cluster-wise analysis are lower compared to the values in Table 2.

Table 6 Correlation results of all the restaurants using SB_1, grouped by cluster for RD_{Yp19} dataset. Column-wise maximum values are bold faced and the statistically significant values are italicized

Category	Parameter	$Cluster-1$		$Cluster-2$		$Cluster-3$		$Cluster-4$		$Cluster-5$	
		ρ	τ	ρ	τ	ρ	τ	ρ	τ	ρ	τ
Living standard	EI^z	***0.24***	***0.17***	*0.26*	*0.18*	*0.29*	*0.19*	*0.44*	0.26	***0.51***	***0.37***
	LS^z	0.09	0.05	0.14	0.10	*0.24*	*0.16*	***0.51***	***0.34***	***0.51***	*0.35*
Tourism significance	TD^z	0.16	0.12	0.13	0.09	***0.30***	***0.21***	0.30	0.21	−0.04	−0.01
	NT^z	0.16	0.12	0.13	0.09	***0.30***	***0.21***	0.30	0.21	−0.04	−0.01
Business convenience	BA^z	*0.22*	*0.15*	***0.31***	***0.21***	*0.19*	*0.12*	−0.10	−0.05	−0.22	−0.15
	NB^z	*0.22*	*0.15*	***0.31***	***0.21***	*0.19*	*0.12*	−0.10	−0.05	−0.22	−0.15
Combined	CA^z	*0.19*	*0.13*	***0.31***	***0.21***	0.07	0.05	−0.01	0.02	−0.03	−0.02

Table 7 Correlation results of all the restaurants using SB_2, grouped by cluster for RD_{Yp19} dataset. Column-wise maximum values are bold faced and the statistically significant values are italicized

Category	Parameter	$Cluster-1$		$Cluster-2$		$Cluster-3$		$Cluster-4$		$Cluster-5$	
		ρ	τ	ρ	τ	ρ	τ	ρ	τ	ρ	τ
Living	EI^z	***0.20***	***0.13***	*0.20*	*0.14*	***0.31***	***0.20***	0.37	0.22	***0.51***	***0.36***
standard	LS^z	0.09	0.05	0.09	0.07	*0.27*	*0.18*	**0.43**	**0.27**	*0.50*	*0.35*
Tourism	TD^z	0.11	0.08	0.09	0.05	*0.23*	*0.16*	0.28	0.19	−0.02	−0.00
significance	NT^z	0.11	0.08	0.09	0.05	*0.23*	*0.16*	0.28	0.19	−0.02	−0.00
Business	BA^z	0.15	0.10	***0.33***	***0.22***	0.15	0.08	−0.00	0.02	−0.17	−0.10
convenience	NB^z	0.15	0.10	***0.33***	***0.22***	0.15	0.08	−0.00	0.02	−0.17	−0.10
Combined	CA^z	0.13	0.09	***0.33***	***0.22***	0.09	0.05	0.06	0.05	−0.01	−0.01

This indicates that the state-wise abridging of the zip codes is still a better choice for studying the effect of location parameters. However, the relative importance of the location parameters remains unaltered. Like state-wise correlation, from cluster-wise correlation results we also infer that living standard parameters are the most important factor on the success of restaurants, when considering the average across all the clusters.

The clustering essentially groups the zip code with similar location related properties. So the relative importance of a location parameter for different clusters are expected to clearly reflect the existing variation. In case of cluster 2, the business convenience appears to have the highest relative importance. The zip codes that have higher significance for both living standard and tourism are grouped in cluster 3. This is evident from the close proximity (EI^z: $\rho = 0.29$, $\tau = 0.19$ and TD^z : $\rho = 0.30$, $\tau = 0.21$) of the corresponding correlation coefficient values, shown in Table 6.

Similar to the previous experiments, in the next step, we computed the cluster-wise correlation using the TripAdvisor RD_{TpAdv} dataset. Table 8 lists the correlation coefficients using business success metric SB_1. The maximum average correlation coefficients are $\rho = 0.29, \tau = 0.20$ (EI^z); and $\rho = 0.21, \tau = 0.14$ (LS^z), which correspond to the living standard. The TripAdvisor dataset essentially corroborates our conclusions derived from the Yelp-2019 dataset.

The cluster-wise analysis can be combined with the state-wise analysis to build confidence regarding relative importance of location feature, for a particular zip code. If both state and the cluster to which a zip code belongs, infer the same dominant location feature, the conclusion is more reliable.

5 Conclusion

In this chapter, we performed a correlational study to understand the relationships of different characteristics of locations represented by zip codes and the online success (defined by reviews, ratings etc.) of restaurant businesses of those locations. We used a location dataset to measure the location characteristic parameters such as living standard defined by cost of living and other parameters. We also quantified the success of restaurants using two restaurant dataset collected from two different sources: Yelp and TripAdvisor. We performed the correlation analysis on zip codes in two ways. First, we grouped the zip codes based on which state it belongs to. Second, we clustered the zip codes based on the geographic and demographic features using an unsupervised machine learning technique.

The cluster-wise and the state-wise segregation of the zip codes yielded a common conclusion that the living standard affects business success the most. This conclusion can be made in an aggregate sense as well. Looking into individual states reveals that, for six out of the seven states analyzed here, living standard played an important role in business success. The analysis was further refined by forming clusters of zip-codes based on location characteristics, which indicated that

Table 8 Correlation results of all the restaurants with SB_1, grouped by cluster for RD_{TpAdv} dataset. Column-wise maximum values are bold faced and the statistically significant values are italicized

Category	Parameter	$Cluster-1$		$Cluster-2$		$Cluster-3$		$Cluster-4$		$Cluster-5$	
		ρ	τ	ρ	τ	ρ	τ	ρ	τ	ρ	τ
Living	EI^z	0.12	0.08	*0.33*	*0.23*	***0.27***	***0.18***	0.26	0.17	***0.48***	***0.32***
standard	LS^z	−0.06	−0.03	*0.26*	*0.19*	*0.22*	*0.14*	0.33	0.21	*0.28*	0.17
Tourism	TD^z	−0.08	−0.04	0.02	0.01	−0.12	−0.08	***0.34***	***0.24***	−0.09	−0.07
significance	NT^z	−0.08	−0.04	0.02	0.01	−0.12	−0.08	***0.34***	***0.24***	−0.09	−0.07
Business	BA^z	**0.13**	**0.10**	***0.34***	***0.23***	−0.10	−0.07	0.13	0.09	−0.14	−0.09
convenience	NB^z	**0.13**	**0.10**	***0.34***	***0.23***	−0.10	−0.07	0.13	0.09	−0.14	−0.09
Combined	CA^z	0.08	0.06	***0.34***	***0.23***	0.05	0.02	0.14	0.10	0.01	−0.00

three out of five clusters have living standard as a predominant variable. Out of the remaining two clusters, we observed in case of one cluster that the tourism significance share similar weights as that of the living standard. The remaining cluster indicated business convenience as a principal factor. It should be noted that across datasets, living standard emerged either as the most dominant factor or as a significant factor for business success.

In this chapter, we empirically showed that when a location is considered singularly, different determinant categories such as demographics, tourism, and business convenience can have positive impact on restaurant success defined in terms of reviews and ratings. The success is particularly high when the restaurant caters to a location with a high living standard such as low living cost, high income of people, better education trend etc. There are other factors such as restaurant's ambience, and service quality which can also influence the success of a restaurant in addition to its location. In future work, we will try to quantify the impact of these factors on the success of restaurants.

References

1. Tripadvisor: Read reviews, compare prices & book. (2020). https://www.tripadvisor.com.
2. Yelp Dataset challenge. https://www.yelp.com/dataset/challenge. Accessed 01 June 2019.
3. HAI: housing affordability index. https://www.nar.realtor/research-and-statistics/housing-statistics/housing-affordability-index/methodology. Accessed 15 Nov 2018.
4. Cost of Living index. https://worldwidecostofliving.com/asp/wcol_HelpIndexCalc.asp. Accessed 15 Nov 2018.
5. Bhowmick, A. K., Suman, S., & Mitra, B. (2017). Effect of information propagation on business popularity: A case study on yelp. In *Proceedings of the 18th IEEE International Conference on Mobile Data Management (MDM)* (pp. 11–20). IEEE.
6. Eravci, B., Bulut, N., Etemoglu, C., & Ferhatosmanoğlu, H. (2016). Location recommendations for new businesses using check-in data. In *Proceedings of the 16th IEEE International Conference on Data Mining Workshop (ICDMW)* (pp. 1110–1117). IEEE.
7. Fieller, E. C., Hartley, H. O., & Pearson, E. S. (1957). Tests for rank correlation coefficients. I. *Biometrika, 44*(3/4), 470–481.
8. Hegde, S., Satyappanavar, S., & Setty, S. (2017). Restaurant setup business analysis using yelp dataset. In *Proceedings of the 6th International Conference on Advances in Computing, Communications and Informatics (ICACCI)* (pp. 2342–2348). IEEE.
9. Hood, B., Hwang, V., & King, J. (2013). Inferring future business attention. Yelp Challenge, Carnegie Mellon University.
10. Hu, L., Sun, A., & Liu, Y. (2014). Your neighbors affect your ratings: On geographical neighborhood influence to rating prediction. In *Proceedings of the 37th International ACM SIGIR Conference on Research & Development in Information Retrieval* (pp. 345–354). ACM.
11. Kendall, M. G. (1938). A new measure of rank correlation. *Biometrika, 30*(1–2), 81–93. https://doi.org/10.1093/biomet/30.1-2.81.
12. Lu, X., Qu, J., Jiang, Y., & Zhao, Y. (2018). Should I invest it?: Predicting future success of yelp restaurants. In *Proceedings of the Practice and Experience on Advanced Research Computing* (p. 64). ACM.

13. Manning, C. D., Surdeanu, M., Bauer, J., Finkel, J., Bethard, S. J., & McClosky, D. (2014). The Stanford CoreNLP natural language processing toolkit. In *Association for computational linguistics (ACL) system demonstrations* (pp. 55–60). http://www.aclweb.org/anthology/P/P14/P14-5010.
14. Parsa, H., Gregory, A., Terry, M., et al. (2011). Why do restaurants fail? Part III: An analysis of macro and micro factors. Institute for Tourism Studies.
15. Spaeder, K. How to find the best location. https://www.entrepreneur.com/article/73784. Accessed 15 Aug 2018.
16. Tayeen, A., Mtibaa, A., & Misra, S. (2019). Location, location, location! Quantifying the true impact of location on business reviews using a yelp dataset. In *IEEE/ACM International Conference on Advances in Social Networks Analysis and Mining* (pp. 1081–1088).
17. Tiroshi, A., Berkovsky, S., Kaafar, M. A., Vallet, D., Chen, T., & Kuflik, T. (2014). Improving business rating predictions using graph based features. In *Proceedings of the 19th International Conference on Intelligent User Interfaces* (pp. 17–26). ACM.
18. Wang, F., Chen, L., & Pan, W. (2016). Where to place your next restaurant?: Optimal restaurant placement via leveraging user-generated reviews. In *Proceedings of the 25th ACM International on Conference on Information and Knowledge Management* (pp. 2371–2376). ACM.
19. Wasserstein, R. L., & Lazar, N. A. (2016). The ASA statement on p-values: Context, process, and purpose. *The American Statistician, 70*(2), 129–133.
20. Ziliak, S., & McCloskey, D. N. (2008). *The cult of statistical significance: How the standard error costs us jobs, justice, and lives*. Michigan: University of Michigan Press.

Identifying Reliable Recommenders in Users' Collaborating Filtering and Social Neighbourhoods

Dionisis Margaris, Dimitris Spiliotopoulos, and Costas Vassilakis

Abstract Recommender systems increasingly use information sourced from social networks to improve the quality of their recommendations. However, both recommender systems and social networks exhibit phenomena under which information for certain users or items is limited, such as the cold start and the grey sheep phenomena in collaborative filtering systems and the isolated users in social networks. In the context of a social network-aware collaborative filtering, where the collaborating filtering- and social network–based neighbourhoods are of varying density and utility for recommendation formulation, the ability to identify the most reliable recommenders from each neighbourhood for each user and appropriately combine the information associated with them in the recommendation computation process can significantly improve the quality and accuracy of the recommendations offered. In this chapter, we report on our extensions on earlier works in this area which comprise of (1) the development of an algorithm for discovering the most reliable recommenders of a social network recommender system and (2) the development and evaluation of a new collaborative filtering algorithm that synthesizes the opinions of a user's identified recommenders to generate successful recommendations for the particular user. The proposed algorithm introduces significant gains in rating prediction accuracy (4.9% on average, in terms of prediction MAE reduction and 4.2% on average, in terms of prediction RMSE reduction) and outperforms other algorithms. The proposed algorithm, by design, utilizes only basic information from the collaborative filtering domain (user–item ratings) and the social network domain (user relationships); therefore, it can be easily applied to any social network recommender system dataset.

D. Margaris
Department of Digital Systems, University of the Peloponnese, Sparti, Greece
e-mail: margaris@uop.gr

D. Spiliotopoulos · C. Vassilakis (✉)
Department of Management Science and Technology, University of the Peloponnese, Tripoli, Greece
e-mail: dspiliot@uop.gr; costas@uop.gr

© The Author(s), under exclusive license to Springer Nature Switzerland AG 2021

M. Çakırtaş, M. K. Ozdemir (eds.), *Big Data and Social Media Analytics*, Lecture Notes in Social Networks, https://doi.org/10.1007/978-3-030-67044-3_3

Keywords Social networks · Recommender systems · Collaborative filtering · Limited information · Near neighbours · Evaluation · Rating prediction

1 Introduction

Nowadays, due to the blistering growth of the available information on the Internet, the task of searching and finding products that may be of interest to users has become an extremely difficult task. Recommender systems (RSs) aim to overcome the information overload problem, by investigating the preferences of online users and suggesting items they might like. Many commercial web services have implemented these systems to recommend products to their clients in a more accurate manner, to increase their profits.

The most widely used approach for making recommendations, stemming from user actions and behaviour, is collaborative filtering (CF). CF synthesizes the informed opinions of people in order to make accurate user rating predictions and personalized recommendations. Since traditional CF relies only on opinions expressed by humans on items, either implicitly (e.g. a user purchases an item or clicks on an advertisement banner, which indicates a positive assessment) or explicitly (e.g. a user submits a specific rating for a particular item), its biggest advantage is that the items' explicit content description is not required [23], since, contrary to content-based RSs, the CF RSs do not recommend items similar to the ones that the users have already experienced (and rated positively). CF works on the assumption that if users had similar tastes on some items in the past (rating assignment, buying, eating, watching, etc.), then they are likely to have similar interests in the future, too [10].

Traditional CF-based RSs assume that users are independent from each other and do not consider the social interactions among them, such as friendship and trust. This approach fails to incorporate important aspects that denote interaction, influence and tie strength among them, which can substantially enhance recommendation quality [4, 11].

Social network-aware RSs take into account both static data sourced from the user profiles (e.g. gender, age and residence), as well as static data sourced from the item profiles (e.g. item price, availability and colour). These features are complemented with dynamic aspects and contextual information stemming from social information, such as user mood and social influence, as well as the item general acceptance and trends in order to supplement the traditional CF data (e.g. the aforementioned static data, as well as user ratings). By taking this information into account in the recommendation process, the social network (SN) RSs manage to achieve more successful and targeted recommendations [16].

However, in some cases the SN- and CF-based information that a RS has at its disposal may be limited: some users may not consent to the use of their SN information for recommendations or may not have SN accounts at all, or the rating data (characteristics and categories of products) may be unavailable for the RS

service. And, conversely, in some cases, the CF-based near neighbours (NNs) of a user U may either be limited in number, or have low similarity, or have little utility, in the sense that they have rated very few items that U has not already rated. Generalizing, we can assert the successful combination of SN- and CF-based information effectively depends on identifying which rating prediction information source (the SN relations or the CF NNs) is considered as the most reliable and useful predictor for each individual user in a SN CF dataset. While evaluating available prediction information sources, we should take into account both (a) the characteristics of the information source (e.g. neighbourhood population, degrees of similarity and levels of influence between the user and her neighbourhood) and (b) the dynamics of the recommendation process, considering in particular the fact that—for many users—their SN relationships play an important role in their responses to recommendations, when compared to the CF NNs that traditional CF RSs use [4].

In this chapter, we propose an algorithm that can be applied to any SN-aware RS, which utilizes both the users' social relations (SN-based information) and the users' ratings on items (CF-based information) and combines them effectively to generate more successful rating predictions. The proposed algorithm addresses the issues of limited SN information or limited CF information for some users, by adapting its behaviour, taking into account the density and utility of each user's SN and CF neighbourhoods. In this context, we present and validate seven alternatives for evaluating the importance of each user's SN and CF neighbourhoods and combining the partial predictions produced by each user's SN and CF neighbourhoods.

Through this adaptation, the proposed algorithm achieves considerable improvement in rating prediction accuracy; this is verified by the results of our experiments, in which the performance of the proposed algorithm is evaluated against five contemporary SN CF datasets. In the same experiments, the performance of the proposed algorithm is compared against the performance of the algorithm presented by Margaris et al. [21], which also tackles the same problem; however assuming that all dataset users share the same prediction significance between CF and SN prediction information in RSs.

Notably, in our experiments we used:

1. Both *dense* and *sparse* SN datasets (a SN dataset density refers to the number of relations when compared to the number of users in it—[4])
2. Both *dense* and *sparse* CF datasets (a CF dataset density refers to the number of ratings when compared to the number of users and items in it—Herlocker et al. 2004)
3. Both *undirected edge* (friendships) and *directed edge* (trusts) SN datasets [11]

The experiment results show that the proposed algorithm introduces considerable prediction accuracy gains in terms of rating prediction error under all conditions (4.9% on average, in terms of prediction MAE reduction and 4.2% on average, in terms of prediction RMSE reduction). Since the proposed algorithm requires only basic SN information (user relationships), as well as basic CF information (user ratings on items), it follows that it can be applied to any SN RS dataset. It is also

worth noting that the proposed algorithm can be combined with other algorithms that have been proposed for improving prediction accuracy, rating recommendation quality or prediction coverage in CF-based systems, focusing either in traditional CF-based systems (e.g. concept drift and clustering techniques—[9]) or in SN CF-based systems (influence, trust, etc.—[5]).

The rest of the chapter is structured as follows: Section 2 overviews related work, while Sect. 3 presents the SN CF prediction formulation foundations. Section 4 presents the proposed algorithm, as well as the alternatives for combining the partial predictions produced by each user's SN and CF neighbourhoods. Section 5 evaluates the proposed algorithm and, finally, Section 6 concludes the chapter and outlines future work.

2 Related Work

RSs are increasingly utilizing SN data to improve the accuracy of the recommendations offered to their users and augment recommendation variety [4, 11], as well as alleviate the issues of cold start, where it is impossible to provide personalized recommendations due to lack of information, and grey sheep, that is, users whose opinions do not agree with any other user and hence a CF RS cannot produce a recommendation [13].

In the aforementioned works, Gilbert and Karahalios [11] present a predictive model that maps SN data to tie strength, differentiating between weak and strong ties with relatively good accuracy, and illustrate how the utilization of tie strength may enhance SN design elements, including friend introductions, message routing, information prioritization and privacy controls. On the other hand, Bakshy et al. [4] investigate the effect of social influence on the consumer responsiveness to online advertisements. More specifically, Bakshy et al. [4] analyse how the presence of cues from a user's social neighbourhood affects the user's responsiveness to online advertisements, taking into account the tie strength between the user and the social connections appearing in the cues. With this, they establish the sizable effect from the inclusion of minimal social cues in advertising and quantify the positive relationship between the consumer response rates and the connection strength among users and affiliated peers appearing in social cues. He and Chu [13] mention that even if a user has no past reviews, a RS still can make recommendations to him based on the preferences of his friends, if it integrates with SNs. They designed a framework that makes recommendations based on the user's own preferences, the general acceptance of a target item and his SN friends' opinions.

Other works utilize SN data to improve the accuracy of the recommendations offered to their users. Capdevila et al. [7] present GeoSRS, a hybrid RS for a location-based SN that enables users to write short reviews about places of interest that they visit. The presented RS uses text mining, as well as geographical location information in order to recommend locations. Margaris et al. [20] propose a query personalization algorithm that exploits the browsing and rating information of items

by users as well as the influence information from SNs used for personalized query adaptation. The queries were adapted by (re)writing the specification of the query sorting procedure to allow for re-ordering of data based on the projected user interest.

Yan et al. [30] propose an approach for managing the complexity of adding social relation networks to RSs. The proposed method, initially, generates an individual relationship network for each user and item, using a fitting algorithm of relationship networks to control the relationship propagation and contracting. Individual relationship networks are subsequently regularized by taking into account the taste diversity between relationship members, in order to capture the time-evolving nature of tastes and emphasize the aspect of homophily. Finally, the regularized individual relationship networks are fused into a matrix factorization algorithm to generate recommendations. Their method is generalized so it can also be applied to the item–item relationship network via item–user role switching. Pham et al. [25] introduce a social RS using memory-based CF models with user-oriented methods as basic models. This is conducted through analysis on the correlations between social relations and user interest similarities. Additionally, they employ sentiment analysis techniques to identify the top-K favourite products for each user, and this information is exploited by the social RS in the rating prediction computation process. Chamoso et al. [8] propose a relationship RS for business and employment-oriented SN. The proposed RS extracts the relevant information from the SN and utilizes it for recommendation on new contracts and job offers to users. The RS utilizes information scraped from user profiles, user activity and job offer descriptions. Then, metrics are applied to discover new ties that are likely to become relationships.

Seo et al. [26] introduce a method to calculate the friendship strength described by the closeness between users in a social circle. Moreover, they propose a personalized RS based on friendship strength to recommend topics or interests that users might have in order to analyse big social data, using Twitter. The measure that they propose can provide recommendations in multi-domain environments for a variety of topics. Zhao et al. [33] propose a rating prediction method for user-services by exploring the rating behaviour of users of social networks. They predict user-service ratings by focusing on the user rating behaviour and, more specifically, on additional rating information, such as the time the user rated the item, what kind of item it was, the user interest that could be mined from the user rating history and the manner that the user rating behaviour diffuses among the user SN relations. Moreover, they introduce the interpersonal rating behaviour diffusion factor for deep understanding of the users' rating behaviour. For the user-service rating prediction method, four factors are fused into a unified matrix-factorized framework: (a) user personal interest (related to user and item topics), (b) interpersonal interest similarity (related to user interest), (c) interpersonal rating behaviour similarity (related to user rating behaviour habits) and (d) interpersonal rating behaviour diffusion (related to user behaviour diffusions).

Yu et al. [31] present a social recommender that is based on the main idea that likability is reflected by distance. This work employs a distance metric learning

approach [29] to derive a distance metric representing the relationships between users and between users and items; these distances are jointly determined by ratings and social relations. This distance metric is combined with matrix factorization, mapping items and users into a unified low-dimensional space and supporting a spatial understanding of the latent factor space and how users and items are positioned inside the space. This approach increases the placement accuracy of users with few ratings, who are 'pulled' close to other users that are similar to them. Finally, the learned metrics and positions are used to generate understandable and reliable recommendations.

Mukamakuza et al. [24] examine the existence of observable relationships between rating behaviour and SN connections in social recommenders. More specifically, they investigate publicly available datasets that contain both traces of rating behaviour along with a social graph. Utilizing SN analysis and statistics techniques, they examine the correlation between high rating activity and multiple item feedback. They check whether high correlation leads to SN centrality and vice versa. Ma et al. [18] propose a CF algorithm based on SN relationship and geographic information as complementary conditions for solving fundamental RS problems, such as raw data sparsity and low accuracy/efficiency. Their proposed algorithm introduces the social relation data into the matrix complementation process. This results in reduced sparsity for the original user–item rating matrix and enhances the authenticity of the data complement. Then, the user geographic information is used for filtering the information that is used for the matrix complementation. This approach lowers the data complementation error and improves the data complementation accuracy. The improvement on the recommendation efficiency and accuracy is achieved through conditional selection of the item complements.

Recently, Amato et al. [3] proposed a RS based on a 'user-centred' approach for recommendations for big data applications. Their approach works by processing interactions between the users and the multimedia content generated in one or more social media networks. Alahmadi and Zeng [2] present an Implicit Social Trust and Sentiment-based RS framework that mines user preferences from online SNs. Their method utilizes the typically overlooked but widely available information from SNs in RSs. Based on the fact that a user opinion is influenced considerably by the opinions of their trusted SN relations, they present a framework to personalize recommendations through the application of new data sources from mining the short text posts of the users' friends from microblogs. The resulting Implicit Social Trust and Sentiment-based RS maps converted recommendations to numerical rating scales through three distinct measures: (1) calculation of the implicit trust between friends, based on intercommunication activities, (2) inference of the sentiment reflected from the information from friends' short posts, called micro-reviews, using natural language for sentiment analysis, enhanced with techniques for handling online social network language features such as emoticons and Internet jargon and (3) quantification of the degree to which the level of trust between friends and sentiment from micro-reviews from friend recommendations impacts each user's opinion, using machine learning regression algorithms, such as support-vector machines, random forests and linear regression.

All the aforementioned works necessitate the availability of additional information, either regarding the user profile (e.g. location, age and gender), the item description (e.g. price, taxonomical categorization and value for money) or the relationships between users (e.g. tie strength and social influence). In this sense, their applicability is limited when compared to the algorithm proposed in this work that requires the availability of just basic SN-sourced information (i.e. trust relationships or elementary friendship among users).

Notably, the work in Margaris et al. [21] presents an algorithm that also confines its needs for SN-sourced data to trust relationships or elementary friendship among users. It computes SN-aware CF-rating predictions by synthesizing a SN-based prediction with a CF-based one. However, the algorithm presented in Margaris et al. [21] uses the same weight coefficients for the SN- and CF-based predictions in the synthesis step, an approach that does not take into account the particular properties of each user's SN-based as well as CF-based neighbourhoods.

This chapter advances the state-of-the-art, by introducing an algorithm that is able to adapt its behaviour to the features of the users' SN- and CF-based neighbourhoods. More specifically, the proposed algorithm analyses the users' already entered ratings and computes a personalized set of weight coefficients associated with SN- and CF-based predictions for each user. Through this approach, the proposed algorithm significantly leverages prediction accuracy. In this chapter, we also present our experiments and findings that quantify the prediction accuracy improvement and establish that the proposed approach consistently achieves improved accuracy under two correlation metrics and across five contemporary datasets that contain both SN relations and CF ratings.

3 SN CF Prediction Formulation Foundations

In CF, predictions for a user U are computed based on a set of users that have rated items similarly to U, namely U's Near Neighbours (NNs). For the majority of the CF systems, the similarity metric between two users U and V is typically based on either the Pearson Correlation Coefficient (PCC) or the Cosine Similarity (CS) metrics [14].

The PCC metric, denoted as $sim\_pcc(U, V)$, is expressed as:

$$sim\,pcc\,(U,\,V) = \frac{\sum_{\mathrm{k}} \left(\mathrm{r_{U,k}} - \overline{\mathrm{r_U}}\right) * \left(\mathrm{r_{V,k}} - \overline{\mathrm{r_V}}\right)}{\sqrt{\sum_{\mathrm{k}} \left(\mathrm{r_{U,k}} - \overline{\mathrm{r_u}}\right)^2 * \sum_{k} \left(\mathrm{r_{V,k}} - \overline{\mathrm{r_V}}\right)^2}} \tag{1}$$

where k ranges over items that have been rated by both U and V, while $\overline{r_U}$ and $\overline{r_V}$ are the mean values or ratings entered by users U and V.

Similarly, the Cosine Similarity (CS) metric, denoted as $sim\_cs(U, V)$, is expressed as:

$$simcs\,(U, V) = \frac{\sum_{\mathrm{k}} \mathrm{r}_{\mathrm{U,k}} * \mathrm{r}_{\mathrm{V,k}}}{\sqrt{\sum_{\mathrm{k}} \left(\mathrm{r}_{\mathrm{U,k}}\right)^2} * \sqrt{\sum_{k} \left(\mathrm{r}_{\mathrm{V,k}}\right)^2}} \tag{2}$$

Afterwards, for user U, his NN users, NN_U, are selected out of the ones whom a positive similarity has been computed with. Then, the rating prediction $p_{U,i}$ for the rating of user U on item i is computed. The computation is expressed as:

$$p_{U,i} = \overline{r_u} + \frac{\sum_{\mathrm{V} \in NN_u} sim_{CF}\,(U, \mathrm{V}) * \left(r_{\mathrm{V},i} - \overline{r_{\mathrm{V}}}\right)}{\sum_{\mathrm{V} \in NN_u} sim\,(U, \mathrm{V})} \tag{3}$$

where the $sim_{CF}(U, V)$ denotes the similarity metric that the particular CF system implementation has selected.

The work in Margaris et al. [21] introduced the concept of *SN NN*s of a user U: user V is considered to be U's SN NN, if a social relation, such as friendship or trust, has been established between them in the context of a SN*S*.

The set of SN NNs of user U will be denoted as SN_NN_U and is formally expressed as:

$$SNNN_U = \{V \in users(S) : r\,(U, V) \in S_r\} \tag{4}$$

where *users(S)* is the set of users within social network S, r is a social relationship within S and S_r is the extension of relationship r in the context of S. Similarly, we denote the initial CF NNs of a user U as CF_NN_U.

Moreover, the algorithm presented in Margaris et al. [21] follows a metasearch score combination algorithm [19] in order to combine the two partial prediction scores. One score is based on the SN-based near neighbourhood of the user (SN_NN_U), while the second is based on the traditional CF near neighbourhood of the user (CF_NN_U). The score from the SN-based near neighbourhood is denoted as $\mathrm{p}_{U,i}^{SN}$ and computed as:

$$\mathrm{p}_{U,i}^{SN} = \frac{\sum_{\mathrm{V} \in SNNN_{U,i}} sim_{SN}\,(U, V) * \left(r_{\mathrm{V},i} - \overline{r_{\mathrm{V}}}\right)}{\sum_{\mathrm{V} \in SNNN_{\mathrm{U}}} sim_{SN}\,(U, V)} \tag{5}$$

As far as the computation of the $sim_{SN}(U, V)$ quantity is concerned, which represents the SN-based user similarity, in this work we adopt the following approach:

- If the SN dataset provides values representing the strength/weight of the relationship between users U and V, $sim_{SN}(U, V)$ is set to this value.
- If the SN dataset does not provide such values, then $sim_{SN}(U, V)$ is fixed to 1.0, for all user pairs *(U, V)* for which a relationship is established within the SN.

Similarly, the CF near neighbourhood-based score is denoted as $\mathrm{p}_{U,i}^{CF}$, computed as:

$$\mathrm{p}_{U,i}^{CF} = \frac{\sum_{\mathrm{V} \in CF_{NNU,i}} sim_{CF}(U,V) * \left(r_{\mathrm{V},i} - \overline{r_{\mathrm{V}}}\right)}{\sum_{\mathrm{V} \in CFNN_{\mathrm{U}}} sim_{CF}(U,V)} \tag{6}$$

$SN\_NN_{U,i}$ and $CF\_NN_{U,i}$ denote the SN- and CF-based NNs of user U, which have rated item i, respectively.

In previous research, certain features of SN structure and/or interaction among users have been shown to denote the strength of relationships between users. Such features include the number of common/mutual relations, tie strength, intimacy of message content and others [4, 20]. In our future work, we plan to investigate methods for exploiting these features, in order to compute or refine the $sim_{SN}(U,V)$ metric.

The partial predictions $\mathrm{p}_{U,i}^{CF}$ and $\mathrm{p}_{U,i}^{SN}$ are combined and the result is adjusted by the mean value of ratings entered by U, U $(\overline{r_{\mathrm{U}}})$ U $(\overline{\mathrm{r_U}})$, in order to formulate the rating prediction $p_{U,i}$, as shown in Eq. (7; [21]):

$$p_{U,i} = \begin{cases} \overline{r_{\mathrm{U}}} + \mathrm{p}_{U,i}^{CF}, & if\, SNNN_{U,i} = \varnothing \\ \overline{r_{\mathrm{U}}} + \mathrm{p}_{U,i}^{SN}, & if\, CFNN_{U,i} = \varnothing \\ \overline{r_{\mathrm{U}}} + \mathrm{w_{CF}} * \mathrm{p}_{U,i}^{CF} + \mathrm{w_{SN}} * \mathrm{p}_{U,i}^{SN}, & if\, SNNN_{U,i} \neq \varnothing \wedge \\ & CFNN_{U,i} \neq \varnothing \end{cases} \tag{7}$$

In Eq. (7), the $\mathrm{w_{CF}}$ parameter corresponds to the weight assigned to the (partial) prediction computed by considering only the *CF_NNs*. The $\mathrm{w_{SN}}$ parameter, which is complementary to the $\mathrm{w_{CF}}$ parameter ($\mathrm{w_{SN}} + \mathrm{w_{CF}} = 1.0$), denotes the weight assigned to the prediction computed by considering only the *SN_NNs* of U, respectively. If no *CF_NNs* of U's who have rated item i exist, then the prediction is based exclusively on the ratings of the user's *SN_NNs* and vice versa.

As shown in Eq. (7), the algorithm presented in Margaris et al. [21] uses the same $\mathrm{w_{SN}}$ and $\mathrm{w_{CF}}$ values (weights) to combine the partial predictions ($\mathrm{p}_{U,i}^{CF}$ and $\mathrm{p}_{U,i}^{SN}$) for all users within each dataset. However, such strategy may be suboptimal, since the properties of the CF- and SN-based neighbourhoods of each user U may vary significantly, necessitating the use of personalized weight assignments. For instance, a user U_1 may have a SN-based neighbourhood of high cardinality and a CF-based one of low cardinality, indicating that the $\mathrm{w_{SN}}$ for this particular user should be assigned a higher value than the respective value of $\mathrm{w_{CF}}$ for the same user.

In the following section, an algorithm that tackles the aforementioned problem is proposed. The proposed algorithm is able to adapt its behaviour by taking into account the density and utility of each user's SN and CF neighbourhoods. Furthermore, seven alternatives for combining the CF and SN partial predictions, calculated by Eqs. (5) and (6), are presented, which target to effectively replace the combination formula (Eq. 7) that the algorithm presented in Margaris et al. [21, 22] uses in order to produce the combined rating prediction value.

4 The Proposed Algorithm and the Partial Prediction Combination Alternatives

This section describes the proposed algorithm, as well as the seven alternatives for combining the CF and SN partial predictions. Finally, the time and space complexity of the proposed algorithm are assessed.

4.1 The Proposed Algorithm

The algorithm proposed in this chapter modifies Eq. (7) that was presented in the previous section, by catering for the use of personalized weights for the two partial predictions, $p_{U,i}^{SN}$ and $p_{U,i}^{CF}$, for the combination step. More specifically, the third case of Eq. (7), which corresponds to the condition (SN_{NNU}, $i \neq \varnothing \wedge CF_{NNU}$, $i \neq \varnothing$), is modified as shown in Eq. (8):

$$\overline{r_U} + w_U^{CF} * p_{U,i}^{CF} + w_U^{SN} * p_{U,i}^{SN} \tag{8}$$

where w_U^{SN} and w_U^{CF} denote the personalized weights for the SN- and the CF-based predictions, respectively.

Listings 1–3 present the aforementioned algorithm in detail; more specifically, Listings 1 and 2 present the computation of the CF- and SN-based partial pre-

```
FUNCTION compute_CF_Prediction(User X, Item it)
/* Implementation of equation (6), used for the CF prediction computation based on X's
    CF_NNs.
    INPUT: X is the user for whom the partial prediction will be computed; it is the respective
        item.
    OUTPUT: The CF partial rating prediction, or NULL if no CF_NN of X has rated it, and
        hence a prediction cannot be computed. */
      predictionNumerator = 0.0
      predictionDenominator = 0.0
      FOREACH Y∈ CF_NN_X
            IF (r_Y,it ≠ NULL) THEN
                  predictionNumerator += sim_CF(X,Y) * (r_Y,i − r̄_Y)
                  predictionDenominator += sim_CF(X,Y)
            ENDIF
      END /* FOREACH */
      IF (predictionDenominator = 0) THEN /* No CF_NN of X has rated it. */
            RETURN NULL
      ELSE /* At least one CF_NN of X has rated it. */
            return r̄_X + (predictionNumerator / predictionDenominator)
      END
END /* FUNCTION */
```

Listing 1 Computation of the CF-based partial rating prediction

```
FUNCTION compute_SN_Prediction(User X, Item it)
/*  Implementation of equation (5), used for the SN prediction computation based on X's
SN_NNs.
    INPUT: X is the user for whom the partial prediction will be computed; it is the respective
item.
    OUTPUT: The SN partial rating prediction, or NULL if no SN_NN of X has rated it, and
hence a prediction cannot be computed. */
      predictionNumerator = 0.0
      predictionDenominator = 0.0
      FOREACH Y∈ SN_NN_X
            IF (r_Y,it ≠ NULL)THEN
                  predictionNumerator += sim_SN(X,Y)* (r_Y,i − r̄_Y)
                  predictionDenominator += sim_SN(X,Y)
            ENDIF
      END /* FOREACH */
      IF (predictionDenominator = 0) THEN /* No SN_NN of X has rated it. */
            RETURN NULL
      ELSE /* At least one SN_NN of X has rated it. */
            return r̄_X + (predictionNumerator / predictionDenominator)
      END
END /* FUNCTION */
```

Listing 2 Computation of the SN-based partial rating prediction

```
FUNCTION compute_Prediction (User U, Item i)
/* INPUT: U is the user for whom the prediction will be computed; i is the respective item.
   OUTPUT: The prediction computed or NULL if no NN (CF or SN) of U has rated i. */
      p^CF_U,i = compute_CF_Prediction (U, i)
      p^SN_U,i = compute_SN_Prediction (U, i)

/* If no NN (CF or SN) of U has rated i, return NULL. */
      IF (p^CF_U,i = NULL && p^SN_U,i = NULL) THEN
            RETURN (NULL)
      ELSE IF (p^CF_U,i = NULL)
            /* If no CF NN exists, the prediction will be based only on the SN_NNs. */
            RETURN (p^SN_U,i)
      ELSE IF (p^SN_U,i = NULL)
            /* If no SN NN exists, the prediction will be based only on the CF_NNs. */
            RETURN (p^CF_U,i)
      ELSE /* Both CF and SN NNs of U have rated it. */
            RETURN (r̄_U + w^CF_U * p^CF_U,i + w^SN_U * p^SN_U,i)
      ENDIF
END /* FUNCTION */
```

Listing 3 Synthesizing the CF- and SN-based partial predictions into a comprehensive rating prediction

dictions, respectively, while Listing 3 presents the synthesis of the two partial predictions into a comprehensive prediction.

In the following subsection, the alternatives for combining the CF and SN partial predictions are presented and analysed, while Sect. 5 presents the experimentally deduced optimal combination and the evaluation.

4.2 Alternatives for Combining the CF and SN Partial Predictions

Regarding the computation of the personalized weights w_U^{SN} and w_U^{CF}, in this chapter we test the seven following alternatives:

1. The prediction is based only on the part (CF or SN) where each user has the largest number of NNs; In case of a tie, the weight is equally split between the two neighbourhoods. According to the above, the weights w_U^{SN} and w_U^{CF} are formulated as follows:

$$\begin{aligned} w_U^{CF} &= \begin{cases} 1, & if \left|CF_{NN_{U,i}}\right| > \left|SN_{NN_{U,i}}\right| \\ 0, & if \left|CF_{NN_{U,i}}\right| < \left|SN_{NN_{U,i}}\right| \\ 0.5, & if \left|CF_{NN_{U,i}}\right| = \left|SN_{NN_{U,i}}\right| \end{cases} \\ w_U^{SN} &= 1 - w_U^{CF} \end{aligned} \tag{9}$$

   and, effectively, the prediction for the rating that user U would assign to item i is computed as shown in Eq. (10):

$$p_{U,i} = \begin{cases} \overline{r_U} + p_{U,i}^{CF}, & if \ \left|CF_{NN_{U,i}}\right| > \left|SN_{NN_{U,i}}\right| \\ \overline{r_U} + p_{U,i}^{SN}, & if \ \left|CF_{NN_{U,i}}\right| < \left|SN_{NN_{U,i}}\right| \\ \overline{r_U} + 0.5 * p_{U,i}^{CF} + 0.5 * p_{U,i}^{SN}, & if \left|CF_{NN_{U,i}}\right| = \left|SN_{NN_{U,i}}\right| \end{cases} \tag{10}$$

   This alternative will be denoted as *max_NNs*.
2. The w_U^{CF} weight is computed as the relative number of the CF_NNs to the number of all the NNs taken into account for the prediction computation (CF_NNs and SN_NNs):

$$\begin{aligned} w_U^{CF} &= \begin{cases} 0, & if \ \left|CF_{NN_{U,i}}\right| = 0 \\ 1, & if \left|SN_{NN_{U,i}}\right| = 0 \\ \frac{CF_{NN_{U,i}}}{CF_{NN_{U,i}} + SN_{NN_{U,i}}}, & if \left|CF_{NN_{U,i}}\right| > 0 \wedge \left|SN_{NN_{U,i}}\right| > 0 \end{cases} \\ w_U^{SN} &= 1 - w_U^{CF} \end{aligned} \tag{11}$$

   This alternative will be denoted as *w_NNs*.
3. The prediction is based only on the part (CF or SN) for which the user has the largest cumulative similarity weight produced by his NNs; in case of a tie, the

weight is equally split between the two neighbourhoods. According to the above, the weights w_U^{SN} and w_U^{CF} are formulated as follows:

$$w_U^{CF} = \begin{cases} 1, & if \sum_{V \in CFNN_{U,i}} sim_{CF}(U,V) > \sum_{V \in SNNN_{U,i}} sim_{SN}(U,V) \\ 0, & if \sum_{V \in CFNN_{U,i}} sim_{CF}(U,V) < \sum_{V \in SNNN_{U,i}} sim_{SN}(U,V) \\ 0.5, & if \sum_{V \in CFNN_{U,i}} sim_{CF}(U,V) = \sum_{V \in SNNN_{U,i}} sim_{SN}(U,V) \end{cases}$$
$$w_U^{SN} = 1 - w_U^{CF} \tag{12}$$

and, effectively, the prediction $p_{U,i}$ for the rating that user U would assign to item i is computed as shown in Eq. (13):

$$p_{U,i} = \begin{cases} \overline{r_U} + p_{U,i}^{CF}, & if \sum_{V \in CFNN_{U,i}} sim_{CF}(U,V) > \sum_{V \in SNNN_{U,i}} sim_{SN}(U,V) \\ \overline{r_U} + p_{U,i}^{SN}, & if \sum_{V \in CFNN_{U,i}} sim_{CF}(U,V) < \sum_{V \in SNNN_{U,i}} sim_{SN}(U,V) \\ \overline{r_U} + 0.5 * p_{U,i}^{CF} + 0.5 * p_{U,i}^{SN}, & if \sum_{V \in CFNN_{U,i}} sim_{CF}(U,V) = \sum_{V \in SNNN_{U,i}} sim_{SN}(U,V) \end{cases} \tag{13}$$

This alternative will be denoted as *max_sim*.

4. The w_U^{CF} weight is computed as the ratio of the sum of similarities of U to her CF-neighbourhood to the sum of (a) the similarities of U to her CF-neighbourhood and (b) the similarities of U to her SN-neighbourhood, that is:

$$w_U^{CF} = \begin{cases} 0, & if \left|CF_{NN_{U,i}}\right| = 0 \\ 1, & if \left|SN_{NN_{U,i}}\right| = 0 \\ \frac{\sum_{V \in CFNN_{U,i}} sim_{CF}(U,V)}{\sum_{V \in CFNN_{U,i}} sim_{CF}(U,V) + \sum_{V \in SNNN_{U,i}} sim_{SN}(U,V)}, & if \left|CF_{NN_{U,i}}\right| > 0 \wedge \left|SN_{NN_{U,i}}\right| > 0 \end{cases}$$
$$w_U^{SN} = 1 - w_U^{CF} \tag{14}$$

This alternative will be denoted as *prop_sim*.

5. The w_U^{CF} weight is computed as the ratio of the average similarity of U to her CF-neighbourhood to the sum of (a) the average similarity between U and the members of her CF-neighbourhood and (b) the average similarity between U and the members of her SN neighbourhood. According to the above, the weights w_U^{SN} and w_U^{CF} are computed as follows:

$$w_U^{CF} = \begin{cases} 0, & if \left|CF_{NN_{U,i}}\right| = 0 \\ 1, & if \left|SN_{NN_{U,i}}\right| = 0 \\ \frac{\frac{\sum_{V \in CFNN_{U,i}} sim_{CF}(U,V)}{\left|CF_{NN_{U,i}}\right|}}{\frac{\sum_{V \in CFNN_{U,i}} sim_{CF}(U,V)}{\left|CF_{NN_{U,i}}\right|} + \frac{\sum_{V \in SNNN_{U,i}} sim_{SN}(U,V)}{\left|SN_{NN_{U,i}}\right|}}, & if \left|CF_{NN_{U,i}}\right| > 0 \wedge \left|SN_{NN_{U,i}}\right| > 0 \end{cases}$$
$$w_U^{SN} = 1 - w_U^{CF} \tag{15}$$

This alternative will be denoted as *prop_avg sim*.

6. The prediction is based only on the part (CF or SN) where each user has the largest ratio of NNs considering the item i (the item for which the prediction is computed) to the number of his overall NNs (this amount will be denoted as $rel_{NN_{U,i}}$). In case of a tie, the weight is equally split between the two neighbourhoods. According to the above, the weights w_U^{SN} and w_U^{CF} are formulated as follows:

$$\begin{aligned} w_U^{CF} &= \begin{cases} 1, & if \left| CFrel_{NN_{U,i}} \right| > \left| SNrel_{NN_{U,i}} \right| \\ 0, & if \left| CFrel_{NN_{U,i}} \right| < \left| SNrel_{NN_{U,i}} \right| \\ 0.5, & if \left| CFrel_{NN_{U,i}} \right| = \left| SNrel_{NN_{U,i}} \right| \end{cases} \\ w_U^{SN} &= 1 - w_U^{CF} \end{aligned} \tag{16}$$

where $CFrel_{NN_{U,i}} = \frac{\left| CF_{NN_{U,i}} \right|}{\left| CF_{NN_U} \right|}$ and $SNrel_{NN_{U,i}} = \frac{\left| SN_{NN_{U,i}} \right|}{\left| SN_{NN_U} \right|}$.

This alternative will be denoted as *max_rel_NNs*. Notably, in this variant the weights are tailored not only to the user for whom the recommendation is formulated for but also to the specific item through the consideration of item-specific neighbourhoods.

7. The w_U^{CF} weight is computed as the ratio of $CFrel_{NN_{U,i}}$ to the sum of $CFrel_{NN_{U,i}}$ and $SNrel_{NN_{U,i}}$, that is:

$$w_{U,i}^{CF} = \begin{cases} 0, & if\ CF_{NN_{U,i}} = 0 \\ 1, & if\ SN_{NN_{U,i}} = 0 \\ \frac{CFrel_{NN_{U,i}}}{CFrel_{NN_{U,i}} + SNrel_{NN_{U,i}}}, & if\ CF_{NN_{U,i}} > 0 \wedge SN_{NN_{U,i}} > 0 \end{cases} \tag{17}$$

while the value of the $w_{U,i}^{SN}$ weight is supplementary to the above (1 - $w_{U,i}^{CF}$).

This alternative will be denoted as *w_rel_NNs*. This alternative, similarly to the previous one, tailors the weights to both the user for whom the prediction is generated for and the item for which the rating is predicted.

In the next section, we will assess the performance of the aforementioned alternatives, in terms of prediction accuracy.

4.3 Complexity Analysis

In this subsection, we assess the complexity of the algorithms presented in the previous paragraphs.

The procedure computing the CF-based partial rating prediction presented in Listing 1 iterates over the user's CF neighbourhood and therefore its complexity is $O(|CF\_NN_U|)$, where CF_NN_U is the collaborative filtering-based near neighbourhood of the user for whom the rating is computed. Wang et al. [28] conclude that the

consideration of the 8 members of the user's CF neighbourhood having the highest similarity with the user suffices to compute accurately this partial prediction, since no notable effect on the rating prediction is observed when more than 8 members are considered. Therefore, we can consider an upper bound for the complexity of this step.

Similarly, the procedure computing the SN-based partial rating prediction presented in Listing 2 iterates over the user's SN neighbourhood and therefore its complexity is $O(|SN_U|)$, where SN_U is the social neighbourhood of the user for whom the rating is computed. The work in Margaris et al. [19] asserts that considering the 8–10 members of each user's social neighbourhood with the strongest influence on the user (as influence is quantified by tie strength [4]) suffices to compute this metric accurately, since considering more members of the social neighbourhood has a negligible effect on the recommendation formulation. Wang et al. [28] concur this finding, further limiting the number of social neighbours that need to be considered to 6. Therefore, we can consider an upper bound for the complexity of this step.

Finally, the partial score synthesis presented in Listing 3 does not involve any iterations, and hence its complexity is equal to *O(1)*.

Taking into account all the above, we conclude that the complexity of the proposed algorithm is

$$O\left(|CFNN_U| + |SN_U| + 1\right) \tag{18}$$

With $|CF\_NN_U|$ and $|SN_U|$ being capped by values 8 and 10, respectively.

Regarding space complexity, the overhead introduced by the proposed algorithm is negligible, compared to a plain CF-SN algorithm, since the additional information required by the proposed algorithm is the social network-based similarity between each user U and each member of U's social neighbourhood. Since, according to the discussion presented above, it suffices to maintain only up to 10 members of the social neighbourhood, the space overhead introduced by the algorithm is also capped to up to 10 real numbers per user.

5 Experimental Evaluation

In this section, we report on the experiments that were designed for the quantification of the achieved rating prediction improvement, from the deployment of the proposed algorithm. The results are compared against the results from:

1. The SN RS algorithm presented in Margaris et al. [21], denoted as *same weights,* which utilizes the same w_U^{SN} and w_U^{CF} weights for all users in each dataset. The *same weights* dataset has been shown to achieve improvements ranging from 1.35% to 3.25% for the dataset listed in Table 1, notably however the optimal values for the w_U^{SN} and w_U^{CF} weights are dataset-specific (e.g. the optimal value

Table 1 Datasets summary

Dataset name	#Users	#Items	#Ratings	Avg. #Ratings /User	Density	#Social Relations	Avg. #Social Relations /User	Type of items	Type of relations
Ciao [12]	30 K	73 K	1.6 M	53.4	0.07%	40K	1.3	General	Trust
FilmTrust [12]	1.5 K	2.1 K	35 K	23.5	1.13%	1.8 K	1.2	Movies	Trust
Epinions [27]	49 K	134	665 K	13.5	0.01%	487 K	9.9	General	Trust
LibraryThings [6]	83 K	506 K	1.7 M	20.5	0.004%	130 K	1.6	Books	Trust
Dianping SocialRec 2015 [17]	148 K	11 K	2.1 M	14.5	0.13%	2.5 M	17	Restaurants	Friendship

for w_U^{SN} is 0.3 for the Ciao dataset, while for the Epinion dataset the respective value is 0.6), hence a training phase must be executed for each dataset.
2. The plain CF algorithm [14], which does not take into account the SN relations.

For all cases, the plain CF algorithm is used as a baseline. In order to quantify the rating prediction accuracy of the contending algorithms, we have used two well-established error metrics, namely the Mean Absolute Error (MAE) and the Root Mean Squared Error (RMSE) that amplifies the importance of large deviations.

To compute the MAE and the RMSE, we employed the standard 'hide one' technique [32], where each time one rating in the database was hidden. Then, based on the ratings of other non-hidden items, its numeric value was tried to be predicted. Furthermore, in our experiments both the PCC and the CS metrics were used.

For hosting the datasets and running the rating prediction algorithms, we used a laptop equipped with a quad core Intel N5000 @ 1.1GHz CPU, 8GB of RAM and one 256GB SSD with a transfer rate of 560 MBps.

In the experiments, we have used five datasets that exhibit the following properties:

1. They contain both user-item ratings, as well as SN user relations.
2. They vary with respect to the type of dataset item domain (music, books, movies, restaurants, etc.), CF-density and SN-density, and size.
3. They are widely used for benchmarking in SN CF research and they are up to date; published the last 10 years.

Table 1 summarizes the basic properties of the considered datasets.

In our first experiment, random ratings from each user are hidden (5 rating selections per user) and then their values are predicted. To further validate our results, we conduct an additional experiment in every dataset containing the timestamps of the ratings, where the last rating from each user in the database is hidden and then its value is predicted. The results of these two experiments were in close agreement (less than 2.5% difference in results) and, therefore, we report only on the results of the first experiment, for conciseness.

In the remainder of this section, we present and discuss the results obtained from applying the algorithm presented in the previous section to the five datasets, using the two aforementioned errors metrics, as well as the two similarity metrics (PCC and CS). From the presentation of the results, we have excluded the variant *prop_avg sim*, since it was found to yield lower rating prediction accuracy in comparison to the baseline algorithm.

5.1 Prediction Accuracy Experiments Using the PCC as the Similarity Metric

Figure 1 illustrates the performance regarding the MAE reduction when the PCC similarity metric is used to quantify the similarity between two users. We can

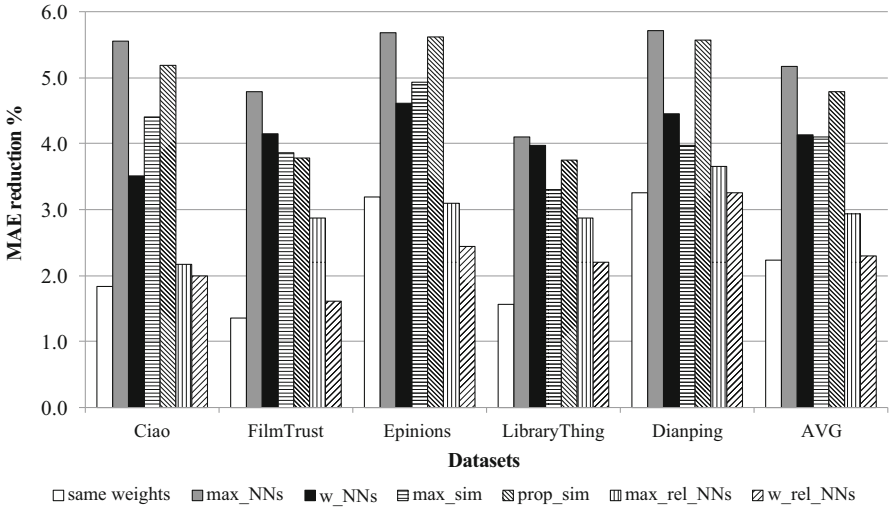


Fig. 1 MAE reduction for all datasets, using the PCC as the similarity metric

observe that the proposed algorithm, using the first alternative (*max_NNs)*, is the one achieving the best results for all five datasets tested. It achieves an average MAE reduction over all datasets equal to 5.2%, surpassing by approximately 2.3 times the corresponding improvement achieved by the *same weights* algorithm (2.2%) presented in [21], which uses the same weights for the CF NNs and SNNNs of all users. At the individual dataset level, the performance edge of the proposed algorithm against the *same weights* algorithm ranges from over 75% for the 'Dianping SocialRec 2015' dataset to 260% higher for the 'FilmTrust' dataset.

It has to be mentioned that the lowest MAE improvement for the proposed algorithm is observed for the 'Filmtrust' and the 'LibraryThing' datasets, which have relatively low #Social Relations / #Users ratio among the five datasets (1.2 and 1.6, respectively). In contrast, the highest MAE improvements for the proposed algorithm are observed for the 'Epinions' and the 'Dianping SocialRec 2015' datasets, which have the highest #Social Relations / #Users ratio among the five datasets (9.9 and 17, respectively). This fact clearly demonstrates the power of the proposed algorithm to exploit the available SN information to improve the RS prediction accuracy.

Considering the other alternatives for computing the weights for the CF and SN neighbourhoods, we can observe that the *prop_sim* algorithm is the runner up, achieving an average improvement of 4.8% against the baseline algorithm, lagging this behind the *max_NNs* algorithm by 0.4%. Interestingly, the biggest gap between the performance of *max_NNs* and *prop_sim* is observed for the 'Filmtrust' and the 'LibraryThing' datasets, which have relatively low #Social Relations / #Users ratios, indicating that the *prop_sim* algorithm achieves good results in more dense social neighbourhoods, but its performance in sparse social neighbourhoods declines.

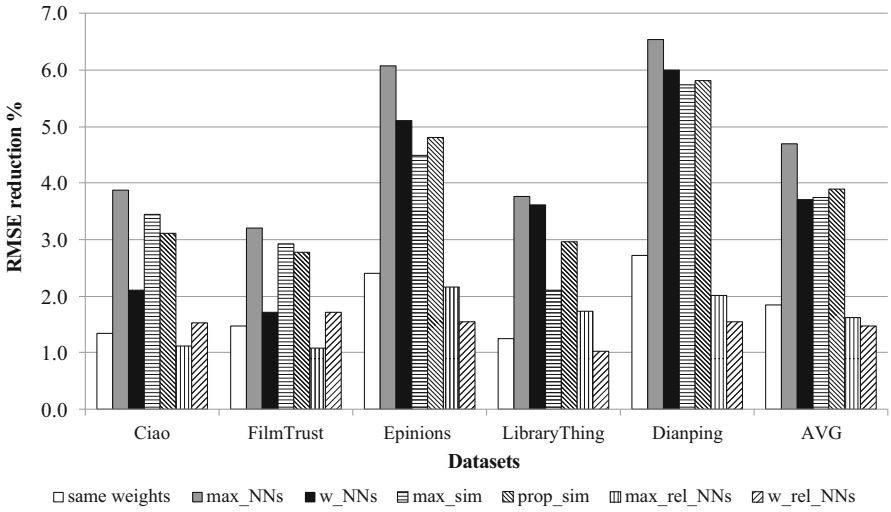


Fig. 2 RMSE reduction for all datasets, using the PCC as the similarity metric

Interestingly, the *max_rel_NNs* and *w_rel_NNs* alternatives, which are the two methods that consider item-specific neighbourhoods, tailoring the weights *both* to the user for whom the recommendation is formulated for *and* to the specific item for which the prediction is formulated, are found to achieve the lowest improvements among all variants discussed in this section. This indicates that for any individual user U, the effect of U*'s* CF and SN neighbourhoods on the rating predictions formulated for U is generally uniform across all items, and the consideration of item-specific neighbourhoods merely adds noise to the rating prediction procedure. This issue will be investigated further in our future work.

Figure 2 demonstrates the performance regarding the RMSE reduction when similarity between users is measured using the PCC.

We can observe that the proposed algorithm, again using the first alternative (*max_NNs)*, achieves the best results for all five datasets tested, with an average RMSE reduction over all datasets equal to 4.7%, surpassing the improvement achieved by the *same weights* algorithm (1.8%), by approximately 2.6 times. At the individual dataset level, the performance edge of the proposed algorithm against the *same weights* algorithm ranges from 120% for the 'FilmTrust' dataset to 300% higher for the 'LibraryThing' dataset.

Furthermore, we can again clearly see that the 'Epinions' and the 'Dianping SocialRec 2015' datasets, having the highest #Social Relations / #Users ratio among the five datasets tested, achieve the highest RMSE reduction, while the other three datasets ('Ciao', 'Filmtrust' and 'LibraryThing'), which have the lowest #Social Relations / #Users ratio achieve the lowest MAE improvement. This fact, again, clearly demonstrates the power of the proposed algorithm to exploit the available SN information, in order to improve the RS prediction accuracy.

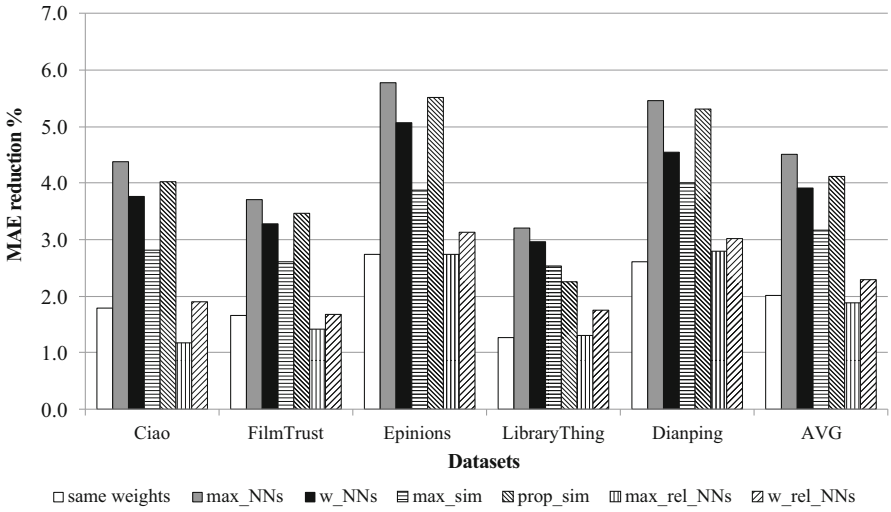


Fig. 3 MAE reduction for all datasets, using the CS as the similarity metric

Regarding the other alternatives for computing the weights for the CF and SN neighbourhoods, we can observe that the *prop_sim* algorithm is again ranked second, albeit with a wider margin from the *max_NNs* alternative than the one observed in the reduction of the MAE (0.8% against 0.4%). This indicates that the *max_NNs* alternative corrects more large errors than the *prop_sim* variant (the RMSE metric penalizes more severely large errors). Again, the two alternatives that consider item-specific neighbourhoods achieve the lowest improvements to the RMSE among all alternatives discussed in this section.

5.2 Prediction Accuracy Experiments Using the CS as the Similarity Metric

Figure 3 illustrates the performance regarding the MAE reduction when the CS metric is used to quantify the similarity between users. We can observe that the proposed algorithm, using the first alternative (*max_NNs)*, is again the one achieving the best results for all five datasets tested, with an average MAE reduction over all datasets equal to 4.5%, surpassing the improvement achieved by the *same weights* algorithm (2%), by approximately 2.2 times. At the individual dataset level, the performance edge of the proposed algorithm against the algorithm that sets the same weights for all users in each dataset ranges from 110% for the 'Dianping SocialRec 2015' dataset to 150% higher for the 'LibraryThing' dataset.

Yet again, that accuracy improvement achieved by the proposed algorithm is positively correlated to the density of available SN relations. Similarly to the case

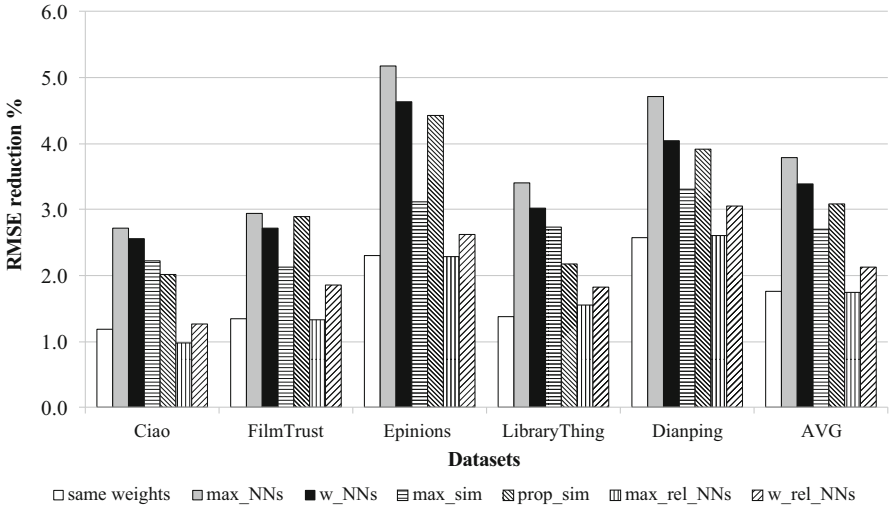


Fig. 4 RMSE reduction for all datasets, using the CS as the similarity metric

of using the PCC similarity metric, the *prop_sim* algorithm is the runner up: it achieves a MAE improvement of 4.1% against the baseline, lagging behind the performance of the *max_NNs* by 0.4%. Moreover, the two alternatives that consider item-specific neighbourhoods achieve the lowest improvements to the RMSE among all alternatives discussed in this section.

Finally, Fig. 4 illustrates the performance regarding the RMSE reduction when similarity between users is measured using the CS metric.

We can observe that the proposed algorithm, using the first alternative (*max_NNs)*, is the one achieving the best results for all five datasets tested, with an average RMSE reduction over all datasets equal to 3.8%, surpassing the improvement achieved by the *same weights* algorithm (1.8%) by approximately 2.1 times. At the individual dataset level, the performance edge of the proposed algorithm against the *same weights* algorithm ranges from 80% higher for the 'Dianping SocialRec 2015' dataset to 150% higher for the 'LibraryThing' dataset.

Furthermore, we can clearly see that the 'Epinions' and 'Dianping SocialRec 2015' datasets, having relatively high #Social Relations / #Users ratio among the five datasets tested, achieve the highest RMSE reduction, while the other three datasets ('Ciao', 'Filmtrust' and 'LibraryThing'), which have the lowest #Social Relations / #Users ratio, achieve the lowest MAE improvement. This clearly demonstrates the power of the proposed algorithm to exploit the available SN information, in order to improve the RS prediction accuracy.

Considering the other alternatives, we can observe that in this case the *w_NNs* variant is ranked second, while the *prop_sim* algorithm is ranked third. Again, the *prop_sim* algorithm appears to mostly remedy prediction errors of smaller magnitude, while the *max_NNs* and *w_NNs* algorithms manage to correct larger

errors, and hence the latter two algorithms surpass *prop_sim* in this case. Once more, the two alternatives that consider item-specific neighbourhoods achieve the lowest improvements to the RMSE among all alternatives discussed in this section.

6 Conclusions and Future Work

Nowadays, where the available information on the Internet is chaotic, the task of recommending interesting products to the users is more difficult than ever. The core task RSs is the investigation of the preferences of online users and suggestion of items they might. CF, which is the most widely used RSs method, synthesizes the people's opinions to make accurate user rating predictions, which will lead to personalized recommendations, under the assumption that if users liked (bought, ate, listened, etc.) common items in the past, they are likely to do so in the future, as well.

However, traditional CF-based RSs assume that users are independent from each other and do not take into account the social interactions among them, such as trust and friendship. As a result, they fail to incorporate important aspects that denote influence and interaction among the users, which can enhance recommendation quality.

The aforementioned drawback has been recently overcome by SN-aware RSs, which take into account information derived from the user profiles and from the item profiles, as well as dynamic aspects and contextual information stemming from social information. With this information in hand, the SN-aware RSs achieve more targeted and hence more successful recommendations.

However, the success of a SN-aware RS greatly depends on the combination of the SN- and the CF-based information, in the sense of identifying which rating prediction information source (the SN relations or the CF NNs) is considered as the most reliable and useful predictor each time.

In this chapter, we proposed an algorithm that effectively combines SN information, specifically user social relations, with CF information, that is user ratings for items. The proposed algorithm formulates two partial prediction scores, from the SN and the CF neighbourhood, and then combines the two partial predictions using a weighted average metascore combination approach.

In contrast to the algorithm presented in Margaris et al. [21], which set the same weights to the two partial predictions for all the users within each dataset, the algorithm proposed in this chapter sets personalized weights for each individual user, based on the density and utility of each individual user's SN- and CF-based neighbourhoods.

In this direction, we have tested seven weight calculation alternatives. The one based only on the partial result, either SN or SF, where each user has the largest number of NNs for the item whose rating is about to be predicted, proved to be the optimal.

The proposed algorithm was validated through a set of experiments, aiming to quantify the improvement obtained in prediction accuracy, due to the consideration of the SN NNs in the recommendation process. In these experiments, five datasets containing both SN information (user–user relation) and CF information (user–item rating) were used. Measurements were taken under the two similarity metrics most widely used in RSs, namely the PCC and the CS. Additionally, two types of social relations, friendship (undirected) and trust (directed), were considered, in order to examine the behaviour of the proposed algorithm under several settings commonly encountered in SN RSs. The algorithm was proven to be highly adaptive to the characteristics of the datasets, yielding promising results in all cases.

The evaluation results have shown that the proposed algorithm may provide substantial improvement on rating prediction quality, across all datasets. The MAE decreases by 5.2% and the RMSE declines by 4.7%, on average, when the PCC metric is used (the respective reductions of the algorithm proposed in Margaris et al. [21] were 2.2% and 1.8%), and by 4.5% and 3.8%, respectively, when the CS metric is used (the respective reductions of the algorithm proposed in Margaris et al. [21] were 2% and 1.8%). In both cases, the performance of the plain CF algorithm is taken as a baseline. Furthermore, the proposed algorithm outperforms the algorithm proposed in Margaris et al. [21] for all cases, by an average of 2.3 times.

Since the proposed algorithm takes into account only each user's CF NN cardinality, as well as his SN NN cardinality, it does not introduce any extra overhead to the prediction calculation procedure, compared to the *same weights* algorithm; on the contrary, we can note that while the algorithm presented in Margaris et al. [21] always calculates both the CF- and the SN-based partial prediction, the algorithm proposed in this chapter can only calculate one partial prediction being thus more efficient (except for the case that the SN and CF neighbourhoods of the user have the same cardinality, where both partial scores need to be computed, thus the prediction formulation cost is similar to that of the algorithm presented in Margaris et al. [21]).

Moreover, the fact that the proposed algorithm achieves the highest error improvement for the datasets that have the highest #Social Relations / #Users ratios among the five datasets tested, under both metrics (PCC and CS), clearly proves the capacity of proposed algorithm to successfully exploit the available SN information to improve the RS prediction accuracy.

The proposed algorithm requires the availability of typical CF information (i.e. a user-rating matrix) and elementary social relation information (bidirectional friendships or unidirectional trusts). However, due to the fact that no additional information, such as users' demographic information (age, gender, nationality, location, etc.), items' characteristics (price, category, reliability, etc.) or SN's contextual information (tie strength, influence, etc.) is required, it can be applied to any SN CF dataset, standalone or in combination with other algorithms that have been proposed for improving rating prediction accuracy and/or coverage, such as matrix factorization models [15] and concept drift techniques [9].

This study has two limitations. Additional SN information (such as users' influence, tie strength, common–mutual relations, demographic data, contextual

information, etc.) is not taken into account for tuning the $sim(U, V)_{SN}$ parameter. Furthermore, the proposed algorithm does not take the age of each user rating into account, in the sense that aged user ratings may not accurately reflect the current state of users regarding their likings and tastes, which may produce inaccurate predictions, due to the concept drift phenomenon [9].

Our future work will focus on investigating the computation-tuning of the $sim(U, V)_{SN}$ parameter value, considering additional information derived from the *SNs* domain. Furthermore, we are planning to evaluate the presented algorithm under additional user similarity metrics, such as the Euclidean Distance, the Hamming Distance, and the Spearman Coefficient [14], for the cases which those metrics are proposed by the literature as more suitable for the additional information. The above can also be utilized in broader applications of prediction methods that utilize social media data, such as textual reviews [22] or user-contributed data for the creation of detailed user profiles [1]. Finally, the combination of the proposed method with concept drift techniques [9] will also be investigated.

References

1. Aivazoglou, M., Roussos, A., Margaris, D., Vassilakis, C., Ioannidis, S., Polakis, J., & Spiliotopoulos, D. (2020). A fine-grained social network recommender system. *Social Network Analysis and Mining, 10*(20), 1–18.
2. Alahmadi, D., & Zeng, X. (2016). ISTS: Implicit social trust and sentiment based approach to recommender systems. *Expert Systems with Applications, 42*(22), 8840–8849.
3. Amato, F., Moscato, V., Picariello, A., & Piccialli, F. (2019). SOS: A multimedia recommender System for Online Social networks. *Future Generation Computer Systems, 93*, 914–923.
4. Bakshy, E., Eckles, D., Yan, R., & Rossen, I. (2012a). Social influence in social advertising: Evidence from field experiments. In *Electronic Commerce 2012 13th ACM Conference* (pp. 146–161). ACM. New York, NY, USA.
5. Bakshy, E., Rosenn, I., Marlow, C., & Adamic, L. (2012b). The role of social networks in information diffusion. In *World Wide Web (WWW) 2012 21st International Conference* (pp. 519–528). ACM. New York, NY, USA.
6. Cai, C., He, R., & McAuley, J. (2017). SPMC: Socially-aware personalized Markov chains for sparse sequential recommendation. In *Artificial Intelligence (IJCAI '17) 2017 International Joint Conference* (pp. 1476–1482).
7. Capdevila, J., Arias, M., & Arratia, A. (2016). GeoSRS: A hybrid social recommender system for geolocated data. *Information Systems, 57*, 111–128.
8. Chamoso, P., Rivas, A., Rodríguez, S., & Bajo, J. (2018). Relationship recommender system in a business and employment-oriented social network. *Information Sciences, 433–434*, 204–220.
9. Gama, J., Zliobaite, I., Bifet, A., Pechenizkiy, M., & Bouchachia, A. (2014). A survey on concept drift adaptation. *ACM Computing Surveys, 46*(4), 1–37.
10. Ge, M., Delgado-Battenfeld, C., & Jannach, D. (2010). Beyond accuracy: Evaluating recommender systems by coverage and serendipity. In *Recommender systems (RecSys) 2010 4th ACM International Conference on* (pp. 257–260). ACM. New York, NY, USA.
11. Gilbert, E., & Karahalios, K. (2009). Predicting tie strength with social media. In *Human Factors in Computing Systems 2009 SIGCHI Conference* (pp. 211–220). ACM. New York, NY, USA.

12. Guo, G., Zhang, J., Thalmann, D., & Yorke-Smith, N. (2014). ETAF: An extended trust antecedents framework for trust prediction. In *Advances in Social Networks Analysis and Mining (ASONAM) 2014 IEEE/ACM International Conference* (pp. 540–547). IEEE/ACM. Beijing, China.
13. He, J., & Chu, W. (2010). A social network-based recommender system (SNRS). *Annals of Information Systems, 12*, 47–74.
14. Herlocker, J. L., Konstan, J. A., Terveen, L. G., & Riedl, J. T. (2004). Evaluating collaborative filtering recommender systems. *ACM Transactions on Information Systems, 22*(1), 5–53.
15. Koren, Y., Bell, R., & Volinsky, C. (2009). Matrix factorization techniques for recommender systems. *IEEE Computer, 42*(8), 42–49.
16. Li, L., Zheng, L., Yang, F., & Lii, T. (2014). Modelling and broadening temporal user interest in personalized news recommendation. *Expert Systems with Applications, 41*(7), 3168–3177.
17. Li, H., Wu, D., Tang, W., & Mamoulis, N. (2015). Overlapping community regularization for rating prediction in social recommender systems. In *Recommender Systems (RecSys 2015) 9th ACM International Conference* (pp. 27–34). ACM. New York, NY, USA.
18. Ma, D., Dong, L., & Li, K. (2018). Collaborative filtering recommendation algorithm based on social relation and geographic information. In *Computer Science and Application Engineering (CSAE '18) 2nd International Conference* (pp. 1–7). ACM. New York, NY, USA.
19. Margaris, D., Vassilakis, C., & Georgiadis, P. (2016). Recommendation information diffusion in social networks considering user influence and semantics. *Social Network Analysis and Mining, 6*(108), 1–22.
20. Margaris, D., Vassilakis, C., & Georgiadis, P. (2018). Query personalization using social network information and collaborative filtering techniques. *Future Generation of Computer Systems, 78*(P1), 440–450.
21. Margaris, D., Spiliotopoulos, D., & Vassilakis, C. (2019a). Social relations versus near neighbours: Reliable recommenders in limited information social network collaborative filtering for online advertising. In *Advances in Social Networks Analysis and Mining (ASONAM 2019) 2019 IEEE/ACM International Conference* (pp. 1–8). IEEE/ACM. New York, NY, USA.
22. Margaris, D., Vassilakis, C., & Spiliotopoulos, D. (2019b). Handling uncertainty in social media textual information for improving venue recommendation formulation quality in social networks. *Social Network Analysis and Mining, 9*(64), 1–19.
23. Massa, P., & Avesani, P. (2004). Trust-Aware collaborative filtering for recommender systems. In *OTM 2004: On the Move to Meaningful Internet Systems 2004: CoopIS, DOA, and ODBASE* (pp. 492–508). Springer: Agia Napa, Cyprus.
24. Mukamakuza, C., Sacharidis, & Werthner, H. (2018). Mining user behaviour in social recommender systems. In *Web Intelligence, Mining and Semantics (WIMS '18) 8th International Conference* (pp. 1–6). ACM. New York, NY, USA.
25. Pham, T., Vuong, T., Thai, T., Tran, M., & Ha, Q. (2016). Sentiment analysis and user similarity for social recommender system: An experimental study. In *Science and Applications (ICISA), Lecture Notes in Electrical Engineering* (Vol. 376). Singapore: Springer.
26. Seo, Y., Kim, Y., Lee, E., & Baik, D. (2017). Personalized recommender system based on friendship strength in social network services. *Expert Systems with Applications, 69*, 135–148.
27. Simon, M., Guillot, E., Médini, L., & Laforest, F. (2019). *RED: A rich epinions dataset for recommender systems, archives-ouvertes.*https://hal.archives-ouvertes.fr/hal-01010246. Accessed 11 Nov 2019.
28. Wang, B., Huang, J., Ou, L., & Wang, R. (2015). A collaborative filtering algorithm fusing user-based, item-based and social networks. In *2015 IEEE International Conference on Big Data* (pp. 2337–2343). IEEE. Washington, DC, USA.
29. Xing, E. P., Jordan, M. I., & Russel, S. (2002). A generalized mean field algorithm for variational inference in exponential families. In *Proceedings of the nineteenth conference on uncertainty in artificial intelligence (UAI'03)* (pp. 583–591). Morgan Kaufmann Publishers: San Francisco, CA, USA

30. Yan, S., Lin, K., Zheng, X., Zhang, W., & Feng, X. (2017). An approach for building efficient and accurate social recommender systems using individual relationship networks. *IEEE Transactions on Knowledge and Data Engineering, 29*(10), 2086–2099.
31. Yu, J., Gao, M., Rong, W., Song, Y., & Xiong, Q. (2017). A social recommender based on factorization and distance metric learning. *IEEE Access, 5*, 21557–21566.
32. Yu, K., Schwaighofer, A., Tresp, V., Xu, X., & Kriegel, H. P. (2004). Probabilistic memory-based collaborative filtering. *IEEE Transactions on Knowledge and Data Engineering, 16*(1), 56–69.
33. Zhao, G., Qian, X., & Xie, X. (2016). User-service rating prediction by exploring social users' rating behaviours. *IEEE Transactions on Multimedia, 18*, 496–506.

Safe Travelling Period Recommendation to High Attack Risk European Destinations Based on Past Attack Information

Dimitris Spiliotopoulos, Dionisis Margaris, and Costas Vassilakis

Abstract Terrorism is a significant deterrent for tourism. It affects both visitors and local citizens and personnel of a country or area. On one hand, the potential visitor will probably avoid travelling to a high attack risk country, due to safety reasons, hence will miss the opportunity to visit it, and, on the other hand, the country's tourism will decline. This work addresses the aforementioned problem by (1) showing that relatively safe visiting periods for high attack risk European countries can be predicted with high accuracy, using limited information, comprising of attack and fatality data from the past years, which are widely available, and (2) developing an algorithm that recommends relatively safe periods to potential travellers.

The results of this work will be useful for tourists, visitors, businesses and operators, as well as relevant stakeholders and actors.

Keywords Terrorist attacks · Tourism · Safety perception · Risk calculation · Safety prediction · Recommendation algorithm · Evaluation

1 Introduction

Terrorist attack reports that are on the news are taken into account by travelling agencies and individual travellers that consider their options to travel. The term 'safe destination' is used to denote countries and cities where crime is not likely to happen against visitors. Crime, in general, and terrorism, in particular, are factors that organisations devoted to global peace, such as The Organisation for Economic Co-operation and Development (OECD) that publishes the Better Life Index [56],

D. Spiliotopoulos · C. Vassilakis (✉)
Department of Management Science and Technology, University of the Peloponnese, Tripoli, Greece
e-mail: dspiliot@uop.gr; costas@uop.gr

D. Margaris
Department of Digital Systems, University of the Peloponnese, Sparti, Greece
e-mail: margaris@uop.gr

© The Author(s), under exclusive license to Springer Nature Switzerland AG 2021

M. Çakırtaş, M. K. Ozdemir (eds.), *Big Data and Social Media Analytics*, Lecture Notes in Social Networks, https://doi.org/10.1007/978-3-030-67044-3_4

and the Institute for Economics and Peace [19] that publishes the Global Terrorism Index (GTI; [15]), use to rank countries for safety.

The country safety indices take into account several geopolitical factors as well as expert opinions based on very recent political events or even terrorist attacks on areas or nearby locations. Terrorism is one of the most impactful factors that affect the citizen perception of safety. According to Pain [42], inciting widespread fear among the global population is a key objective behind terrorist attacks. According to the Institute for Economics and Peace, global terrorism peaked in 2014, with an unprecedented increase of 80% between 2013 and 2014. While terrorist attacks had then receded to the levels of 2013, the attacks are still very widely spread between countries. This is reflected by incidents of terrorism happening in relatively low-risk countries, maintaining terrorism as a prime public fear factor for many countries.

The influence of terrorism in the decisions of potential visitors, domestic or international, is undeniable [52]. Countries and prospective visitors, being dependent on perceived safety, may utilise such information to gradually build trust between countries and visitors for economic recuperation and tourism viability, allowing for stable, unhindered economic growth [51]. On the other hand, authorities may use this information by opting to examine the unsafer time periods and prepare accordingly to shield against and ultimately prevent terrorism in tourist destinations [24]. The news may also use such information, to inform tourists, citizens and businesses on safety [22].

Perceived destination safety is dependent on the recent past events. The frequency and severity of such events is measurable. In regard to tourism, the concerned parties are the travellers and tourists, as well as businesses, such as tour operators, hotel managers and others. Risk perception and risk estimate clearly differ among these groups; hence, an aggregated and uniform risk assessment is inherently of limited utility. Since terrorism is a targeted act, past data may prove useful for the derivation of patterns that create the terrorism attack footprint for a specific country [17]. Figure 1 depicts the frequency of recorded attacks and fatalities for Turkey in 2017. As seen, the terrorist activity is not uniform over time in a year. Moreover, there are periods of time that the terrorism activity is lower, shown in blue in Fig. 1. Therefore, for a traveller, it would be useful to find out the statistically low terrorism activity season as that would be the optimal time to travel.

The above consideration is useful for travellers that plan to visit a particular European country that has experienced recent high terrorism activity. Prospective visitors would still wish to visit; however, they would also like to feel as safe as possible.

Previous work [55] utilised past terrorism information, found in *Global Terrorism Database* (GTD; [27]), and more specifically, the number of tourism-related attacks, target types and attack types, in order to estimate the number of attacks a country may suffer in the following years, targeting at upgrading a country's safety measures and vice-versa. More specifically, the two main findings of our previous work are:

1. The tourism-related attack patterns mostly followed the general attack patterns, despite the tourist and non-tourist terrorist attack ratio for a country.

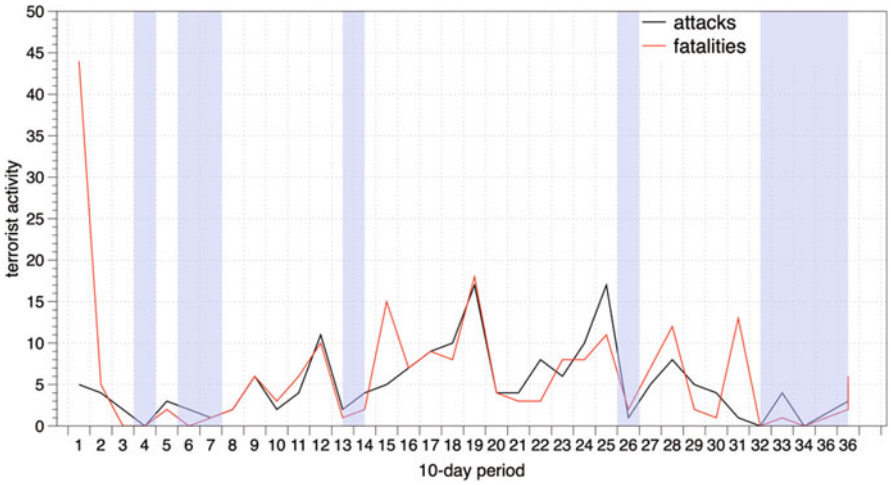


Fig. 1 Terrorist activity in Turkey in 2017 (source: GTD); the blue ribbon shows periods of lower terrorist activity

2. The number of attacks of the last three years proved to be a simple yet effective predictor for the following year's attack number, and irrelevant to the number of attacks recorded for a country.

Further analysis of these two findings is given in Sect. 3.

The present work extends the aforementioned work by proposing an algorithm that recommends optimal time slots to visit high terrorist activity European countries, providing valuable information to the query 'show me the safest time to visit country Y' a visitor will probably ask, hence elevating the state-of-the-art in this research area from safety evaluation to active safe period recommendation.

More specifically, the present work:

1. Studies how terrorist attacks disperse over the year-long periods per country
2. Proposes an algorithm that recommends the safest periods for travellers to visit a high terrorist activity European country (the previous work focused on the data analysis to understand how past information can be useful but did not recommend safe periods for travel)
3. Validates the proposed algorithm, by applying it to the European countries that have been targeted severely in the last years

The following hypotheses are examined in this work:

- H1—Is the algorithm applicable to high- and low-density data alike?
- H2—Is the algorithm accuracy retained for recommendations for earlier years?
- H3—Is the algorithm accuracy retained for non-yearly special events, such as the Olympics, which may shift tourism flow and terrorism focus in specific periods?

The experimental results show that the proposed algorithm introduces considerable prediction accuracy, achieving to predict low-risk attack visiting periods for the majority of the cases applied. It has to be mentioned that since the proposed algorithm requires only basic information (date and target type of the attack, fatalities, etc.), it is applicable to most cases.

It is also worth noting that the proposed algorithm can be enriched with additional input data that affect the terrorist attack events of a country, such as political, economic and social information.

The rest of the chapter is structured as follows: Sect. 2 overviews related work, while Sect. 3 describes the algorithm prerequisites that are used in our work. Section 4 presents the proposed recommendation algorithm. Section 5 evaluates the proposed algorithm and, finally, Sect. 6 concludes the chapter and outlines future work.

2 Related Work

In general, terrorism is negatively correlated to tourism in the literature [6, 47]. Terrorism inspires fear, which spans across citizens and visitors alike [22]. Earlier works studied the importance of safety on the attractiveness of tourist destinations, finding that safety is an important, marketable factor for tourist destinations [18]. That risk is calculated by potential visitors as a quality factor, which is as important as the natural beauty of the destination [16]. Tourism is a major source of income for tourist destination countries and terrorists know that targeting tourist destinations may force their respective governments to be involved in the international politics [2, 14, 46].

Targeting tourist destinations affects people that are directly involved in the tourist sector and businesses within the tourist industry, as well as a large portion of the economy partners that supports that industry indirectly, such as food suppliers; in total, it affects countries' economies heavily [25, 39]. Moreover, it has been observed that terrorist attacks in one tourist destination have an effect on tourism for other tourist destinations, based on correlations of places, geography, politics and other factors [5].

Studies have found that there is a difference on the level of impact of terrorism attacks between places with high and low tourist activity [7]. High tourist activity places are affected heavily in the short term but may recover over time based on factors such as media coverage, such as reports on life returning to normal, advertisements or tourist resilience to terrorism [9, 26], while low activity places are greatly affected in the long term, resulting in the tourist operations going out of business [28, 59]. The effects of terrorist attacks on an area result in tourism decline and may take a period of six months to a year for the local tourism industry to recover [44], based on how the perception of safety by potential visitors is restored [23].

Tourists, as opposed to local population, develop expectations and make decisions on whether to visit a destination based on past experiences and general perception of what they expect to experience [50]. Due to that fact, tourists are easy targets to be affected by terrorism [49]. Another business that is also involved in tourism and terrorism is the media, which reports on the impact internationally [8].

Since the recent increase in terrorist attacks, predicting future terror attacks is an important aim of the society and the individual governments. The public opinion steadily supports actions that aim to deter and shield the citizens from terrorism [45]. Understanding the dynamics of terrorism is a major step towards terrorist event prediction [21, 57]. Several works use machine learning, statistics and other big data analysis techniques to find patterns and predict terrorist events [11, 40, 41, 43, 58]. Kalaiarasi et al. [21] used the GTD and machine learning to predict terrorist threat, while Xia and Gu [60] utilised the GTD data to build a terrorism knowledge graph. Yang et al. [61] built a model to predict the lethality of terrorist attacks and tested it using the GTD data. All the aforementioned works try to predict terrorism, as in high terrorism activity or events and related parameters, such as fatalities. Recently, the work in Spiliotopoulos et al. [55] utilised limited past terrorism information, such as the number of tourism-related attacks, target types and attack types, to estimate the number of attacks a country may suffer in the following years, proving that terrorism attack estimation is possible.

None of the aforementioned works addresses the aspect of low terrorism activity prediction, and cannot be directly helpful to the prospective traveller, who plans to visit a country. The prospective visitor would benefit from knowing a predicted safe period, especially for countries that have had a significant number of terrorist attacks in the recent past.

This chapter advances the state-of-the-art on terrorism attack prediction, by analysing patterns of terrorism attacks, in yearly periods, in order to recommend relatively safe travelling periods to visit high attack risk European countries. Being able to provide specific time periods when the safety may prove to be much higher than the generic perception of safety (or non-safety), provided by international summits and politics, is a unique service to travellers and tourism businesses. The findings of this work support the fact that specific limited information from past terrorism data can be utilised to provide recommendations on timeslots that have a higher chance of being safe (higher predicted safety) for high terrorist attack risk destinations. Apart from the apparent immediate effect on visitor option for a safer travel time, such information may be used to filter time-series data such as social media data streams or surveillance data, may those be gathered through social networks [3, 34, 35, 48, 54] or other data sources, such as the IoT [33], to focus on specific points in time.

3 Algorithm Prerequisites

The analysis of the GTD by Spiliotopoulos et al. [55] was tourism centred. It utilised a subset of the GTD data that contained the tourist-related information. As an example, Fig. 2 depicts the attacks in France in the span of 18 years (2000–2017). A very high proportion of the attacks in France was tourist-related, that is, the attacks targeted locations where tourists would be, as opposed to targeting empty government facilities in remote locations. However, it is also shown that the tourism-related attack patterns mostly followed the general attack patterns. Other examples of this dispersion are Spain (geographically belonging to Western Europe), depicted in Fig. 3, and Greece (geographically belonging to Southeast Europe), depicted in Fig. 4.

France, Spain and Greece are bordering countries in Europe, but the terrorist attack pattern among them is quite opposite. One common factor, however, is that for all three countries, the patterns of the tourist and the overall attacks relatively match. For France and Spain, the number of tourist-related attacks was higher than the non-tourist-related ones. For Greece, the opposite was true. However, for all three countries the aforementioned attack matching pattern holds, albeit for Greece the non-tourist attack information is of greater value, due to the sheer difference in volume. Prediction algorithm worked well for the tourist data subset and since the subset pattern is very similar to the full data pattern, we opted to utilise the full data for the proposed algorithm, which is data driven.

The second finding of the work by Spiliotopoulos et al. [55] was the predictor data range. The results of that work showed that the data (tourist-related) of the last three years, and more specifically the accumulated attack number, are simple,

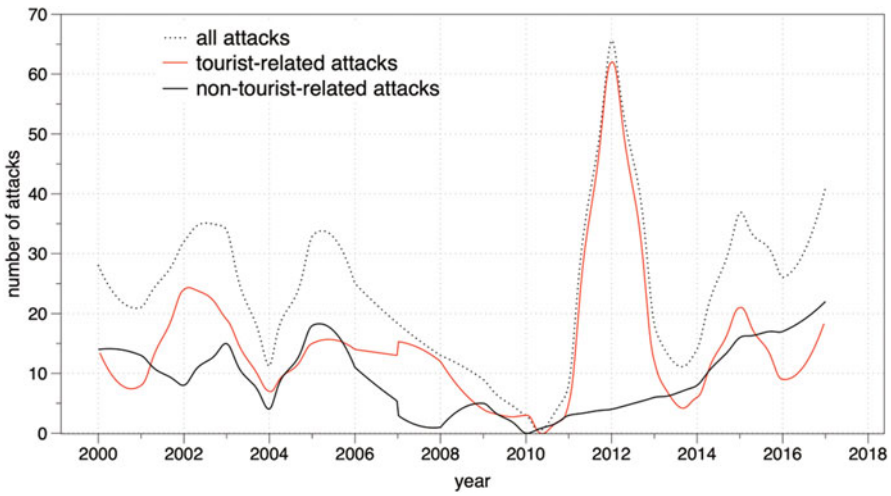


Fig. 2 Tourist- vs non-tourist-related attacks for France, 2000–2017

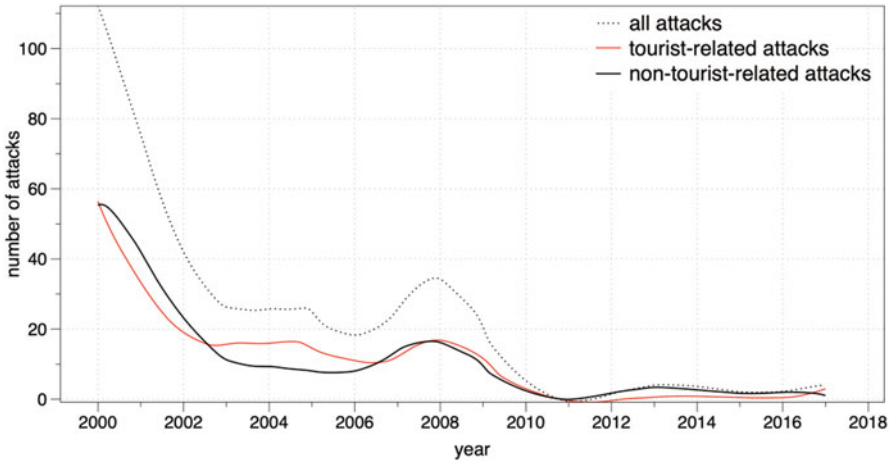


Fig. 3 Tourist- vs non-tourist-related attacks for Spain, 2000–2017

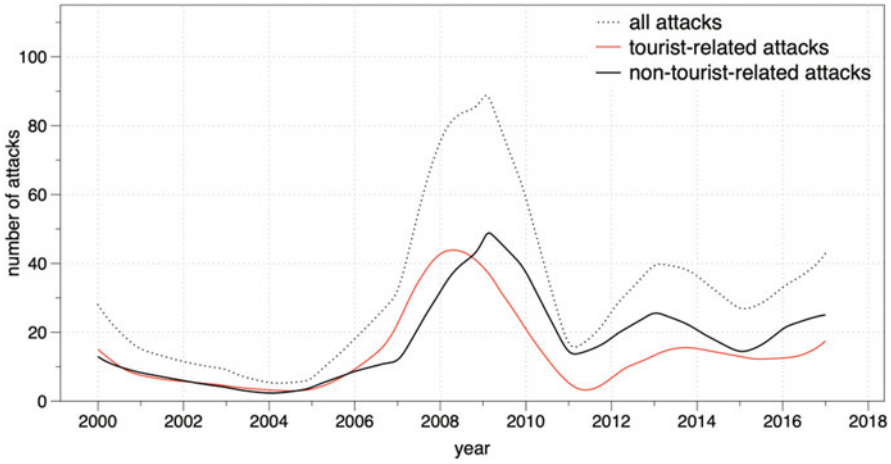


Fig. 4 Tourist- vs non-tourist-related attacks for Greece, 2000–2017

yet effective and reliable, predictors. That result was derived by using pruning techniques for enhancing prediction accuracy in recommender systems [29, 31, 32].

The proposed algorithm, therefore, is designed to use the last three years of data from the GTD. For the purpose of this work, no additional information or data source is considered. Although economic, political, social and intelligence information could result in higher accuracy or more reliable results, this work aims to develop a recommender utilising very limited information (number of attacks and time), which can be easily acquired from the GTD or similar widely available source. Therefore, the proposed algorithm is designed for standalone, broad use and is highly adaptive due to the limited information requirements. The proposed

approach, however, can also be combined with other works that use multi-source data or enriched information.

4 Prediction Algorithm

The aim of this work is to design an algorithm to predict safe periods for travel to a European country, based on the limited information described in the previous sections. Tourism statistics show a variation in the number of days that visitors spend per country. Eurostat [13] reports that for the EU-28 the number of days span from 1.4 to 20.8, with an average of 6.1 days per visitor. Therefore, a ten-day window for safe timeslot prediction is a suitable baseline for our work.

Listing 1 presents the recommendation algorithm, which accepts as input the European country for which the recommendation will be generated and produces as output the recommended period of the year, computed as the period with the fewest incidents. The idea behind the proposed algorithm is to create a total of 36 ten-day slots (three per month, marking the start, middle and end of each month) and recommend the period corresponding to the slot having the smallest number of accumulated attacks over the last three years (following the work in [55]).

In the next section, we assess the performance of the aforementioned recommendation algorithm, in terms of predicting safe visiting periods for high terrorist attack risk countries.

5 Experimental Results

In this section, we report on the experiments that were designed to measure the accuracy of the proposed algorithm on recommending safe periods for visiting countries with high terrorism attack risk.

More specifically, from the GTD dataset, we selected the ten European countries that have had the most attacks over the last years, under the condition that a country had no more than two consecutive zero attacks per year, for the three years prior to the one that was predicted (since the experiments compute data from the three years prior to the target year), so that a prediction can be formulated based on the algorithm presented in Spiliotopoulos et al. [55]. Since (1) the GTD dataset contains attacks until 2017, and (2) based on the work in Spiliotopoulos et al. [55], taking into account the attacks of the last three years proved to be the optimal predictor for the volume of attacks that will happen in the next year, we store in a separate file the attacks that occurred between 2014 and 2016, for the aforementioned ten countries, targeting at recommending a relatively safe period for the year 2017, for each of the ten countries. The data for year 2017 are then used to assess the quality of the recommendation, that is, whether the algorithm achieved to recommend a period with ‘few’ incidents. This method is analogous to the ‘hide one’ technique,

```
FUNCTION populateIncidentHistogram(incidentSet): IncidentHistogram
/* Populates a histogram regarding the number of incidents that have occurred in each period of the year in a
    specific European country. Periods of years correspond to a duration equal to one third of month
    (BEGINNING, MIDDLE, END), thus totalling to 36 periods per year.
    Input: The set of incidents that have occurred in the country. Each has a "date" field, indicating when the
    incident occurred.
    Output: The histogram. The histogram elements are indexed by (month, monthPeriod) pairs, where month-
    Period∈ {BEGINNING, MIDDLE, END}. */
/* Initialize all histogram elements to 0 */
FOR month = 1 TO 12
    FOREACH monthPeriod {BEGINNING, MIDDLE, END}
        histogram[(month, monthPeriod)] = 0
    END /* FOREACH */
END /* FOR */
/* Select incidents that have occurred in the last three years. */
consideredIncidents = {i∈incidentSet: 1 ≤ CURRENT_YEAR - extractYear(i.date) ≤ 3}
FOREACH incident ∈consideredIncidents
    month = extractMonth(incident.date)
    day = extractDay(incident.date)
    /* Days 1-10 are mapped to BEGINNING; days 11-20 to MIDDLE; and all other days to END. */
    monthPeriod = mapDayToMonthPeriod(day)
    histogram[(month, monthPeriod)]++
END /* FOREACH */
RETURN histogram
END /* FUNCTION */

FUNCTION createAllHistograms(European_countries, incidentSet): IncidentHistogram[]
/* Populates a histogram array for all countries.
    Input: The set of countries, and the incident dataset. Each incident has a "date" field, indicating when the
    incident occurred and a "country" field, designating the country.
    Output: The array of histograms, indexed by the country. */
FOREACH country ∈ European_countries
    incidentsInCountry = {incident ∈incidentSet: incident.country = country}
    result[country] = populateIncidentHistogram(incidentsInCountry)
END /* FOREACH */
RETURN result
END /* FUNCTION */

FUNCTION generateRecommendation(country, histogramArray): PeriodOfYear
/* Generates a recommendation for the safest period of the year to visit a country.
    Input: The country for which the recommendation will be generated, and the array of histograms computed
    by function createAllHistograms.
    Output: Recommended period of the year, computed as the period with the fewest incidents. */
countryHistogram = histogramArray[country]
recommendation = NULL
/* Iterate over the elements of the histogram; period is the period index and value is the number of incidents in
the period. */
FOREACH (period, value) ∈countryHistogram
    IF ((recommendation == NULL) OR (value <countryHistogram[recommendation].value)) THEN
        recommendation = period
    END /* IF */
END /* FOREACH */
RETURN recommendation
END /* FUNCTION */
```

Listing 1 The proposed recommendation algorithm

Table 1 The ten European countries that suffered the highest number of terrorist attacks (2014–2016)

Country	Attacks	Fatalities
Turkey	1058	1535
United Kingdom	322	10
Russia	124	152
Germany	122	28
Greece	88	67
France	77	258
Ireland	76	2
Italy	23	1
Kosovo	14	2
Spain	8	0

commonly used in recommender systems for prediction evaluation [12, 20, 38]. The same process was repeated for safe period prediction for the year 2016, in this case by pruning the 2017 information and using the data from the three years prior to 2016 (2013–2015). Finally, the algorithm was also validated for 2015, using data from the three years prior (2012–2014) to test for H2.

Notably, the ten European countries examined in our experiment are depicted in Table 1, along with the number of attacks and the number of fatalities from those attacks between 2014 and 2016. These ten cases include EU countries with high tourism activity, such as Spain (no. 1), Italy (no. 2), the UK (no. 3) and France (no. 4) as the top European tourism destinations, according to Eurostat [13].

In order to provide a better service for the potential visitors, we adjust the algorithm to recommend a total of two periods (instead of just one), as an alternative option recommendation for the period of visit.

In this experiment, we use the following evaluation metrics for the periods recommended by our algorithm:

1. The number of the incidents (attacks or fatalities) that took place in the proposed period, when compared to the average attack number per period per country (#attacks in 2017 over 36 ten-day periods). This metric will be denoted as RNoI—Relative Number of Incidents. A recommendation is considered successful when its RNoI value is less than 100.0% (i.e. less than the average number of attacks per ten-day period).
2. The quartile that the recommended period belongs to, for each country. A recommendation is considered successful when a period belongs to either Q1 (very successful) or Q2 (successful).

In the remainder of this section, we present and discuss the results obtained from applying the algorithm presented above to the ten countries listed in Table 1, using the two aforementioned metrics, as well as two evaluation parameters, the number of attacks (Sect. 5.1) and the number of fatalities (Sect. 5.2). The reasons behind the evaluation of the proposed algorithm with two different parameters are (1) because both of these parameters are extremely important for the safety of tourists and (2)

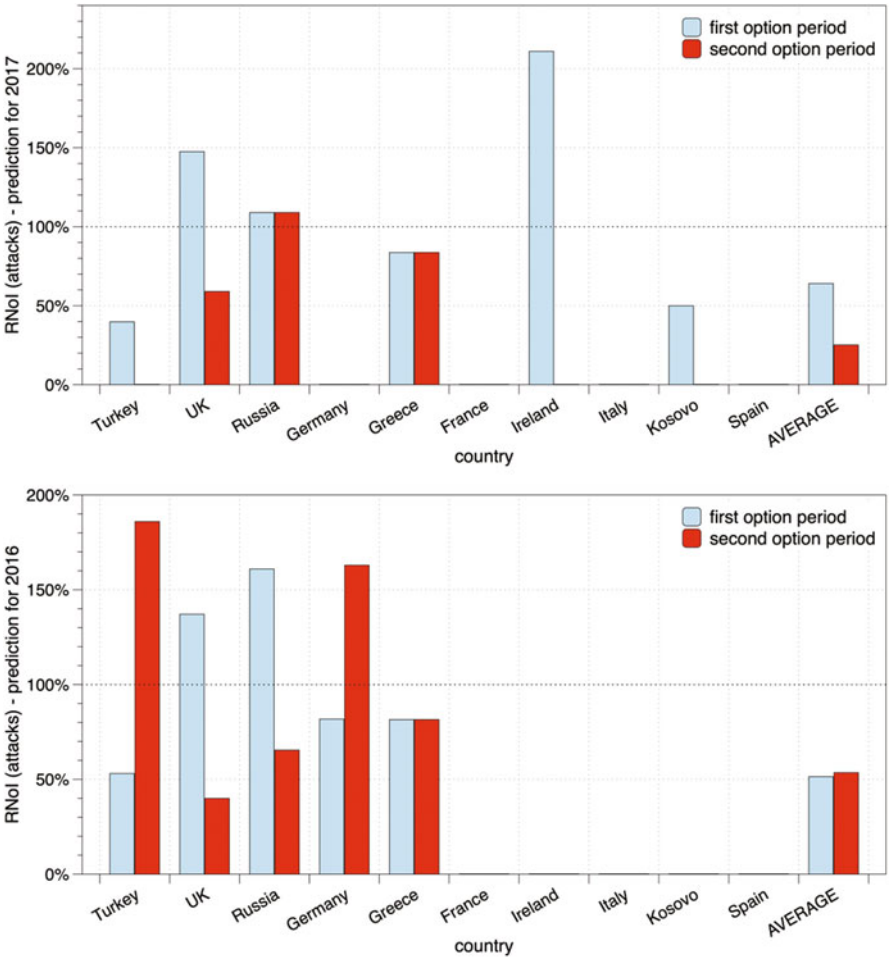


Fig. 5 RNoI results when taking the number of attacks as the evaluation parameter

to prove that the algorithm is parameter-independent, hence it can be applied in many-use cases.

5.1 Number of Attacks as the Evaluation Parameter

Figure 5 illustrates the results of the experiments, regarding the RNoI metric for the ten countries, when using the number of attacks as the evaluation parameter, for the 2017 prediction (top) and for the 2016 prediction (bottom). The average value for all the countries tested in our experiment is also included.

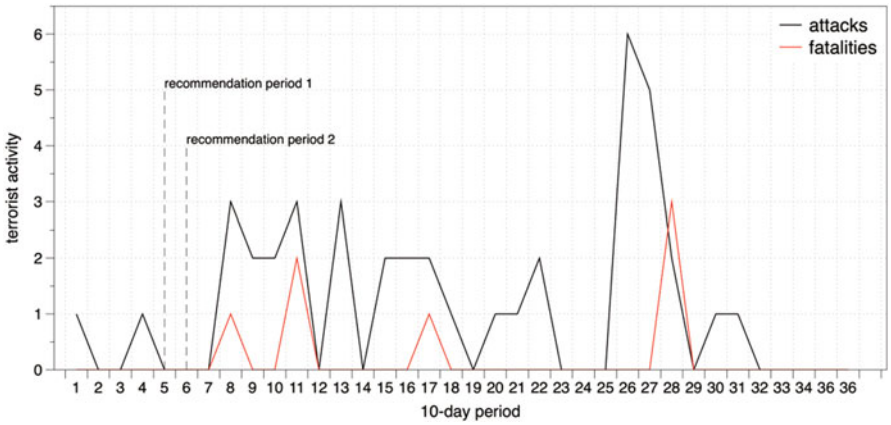


Fig. 6 Recommended periods for France (prediction year: 2017; predictor: *attacks*)

Regarding the predictions for year 2017, we can observe that the average RNoI value for the 10 countries is 45% (64% for the first recommended period and 25% for the second), while for 80% of the 20 recommendations produced (10 countries * 2 recommendations per country), the RNoI values were found to be less than 100% (i.e. less than the average number of attacks over the 36 ten-day periods in a year), indicating successful recommendations. Furthermore, we can clearly see that for the cases of Germany, France, Italy and Spain, the proposed algorithm achieved to recommend two periods with zero attacks for the four countries.

We repeated the same experiment for prediction of 2016 (after pruning the attacks of 2017 and using the attacks that occurred between 2013 and 2015 as predictor input) and the results are shown in the lower part of Fig. 5. More specifically, the average number of the RNoI value of the ten countries is 53% (51% for the first recommended period and 54% for the second). As before, for 80% of the 20 total recommendations produced, the RNoI values were found to be less than 100% (threshold), indicating successful recommendations. In this case, the algorithm achieved to recommend two periods with zero attacks for five countries (France, Ireland, Italy, Kosovo and Spain).

Similarly, for the 2015 predictions, the average was 47% for the primary and 50% for the secondary prediction. This shows that the algorithm worked successfully for past target years, using data from the three years prior to the target year.

More specifically, in the case of France, which was referred to in the introduction of this chapter, the proposed algorithm recommended, as safe visiting periods, the middle and the end of February, that is, the fifth and sixth ten-day period of the year (as indicated in Fig. 6).

Moreover, as far as Turkey is concerned, a country where 181 attacks occurred in 2017, the proposed algorithm successfully recommended periods with only 1 attack and 0 attacks, for the first and second recommendation, respectively. Considering that the average number of attacks in a 10-day period is 29.39 (i.e. 1058/36),

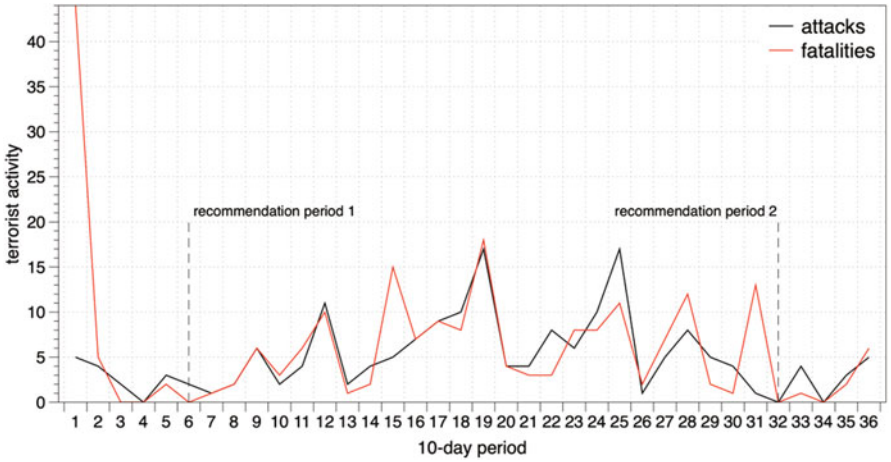


Fig. 7 Recommended periods for Turkey (prediction year: 2017; predictor: *attacks*)

these recommendations are deemed as very successful. Figure 7 illustrates the two periods proposed for visiting Turkey (end of February and middle of November, corresponding to the 6th and 32nd 10-day period of the year), along with a graph depicting the number of attacks and fatalities throughout the year.

Figure 8 illustrates the results of the evaluation under the second metric, that is, the quartile that each recommended period belongs to, for the ten countries. The average value for all countries considered in our experiment is also calculated and shown to the far right of the figure.

Figure 9 presents the aggregate results for all recommendations generated (20 in total = 10 countries * 2 recommendations per country). For each aggregated quartile value, the left column represents the algorithm prediction and the right column represents the aggregated values from ten randomly chosen ten-day periods averaged for each country.

Regarding the 2017 prediction (top of Fig. 9), we can observe that the vast majority of the recommendations (85%) are considered successful (Q1 and Q2), while only the 10% belongs to the Q4 (meaning that the recommendation is considered very unsuccessful, actually recommending a period with high attack risk to the visitors). The investigation and handling of this phenomenon will be part of our future work. At country level, we can see that only 3 out of the 20 cases that Fig. 9 presents are categorised in Q3 and Q4; however, 12 of them are categorised in Q1, while all the others in Q2. In the same figure, the results from the randomly chosen periods are significantly worse, showing safe periods (Q1 and Q2) for only 60% of the cases.

For the 2016 prediction (bottom of Fig. 9), 80% of the recommendations are successful (Q1 and Q2). As for 2017, the results from the randomly chosen periods are successful for 60% of the cases.

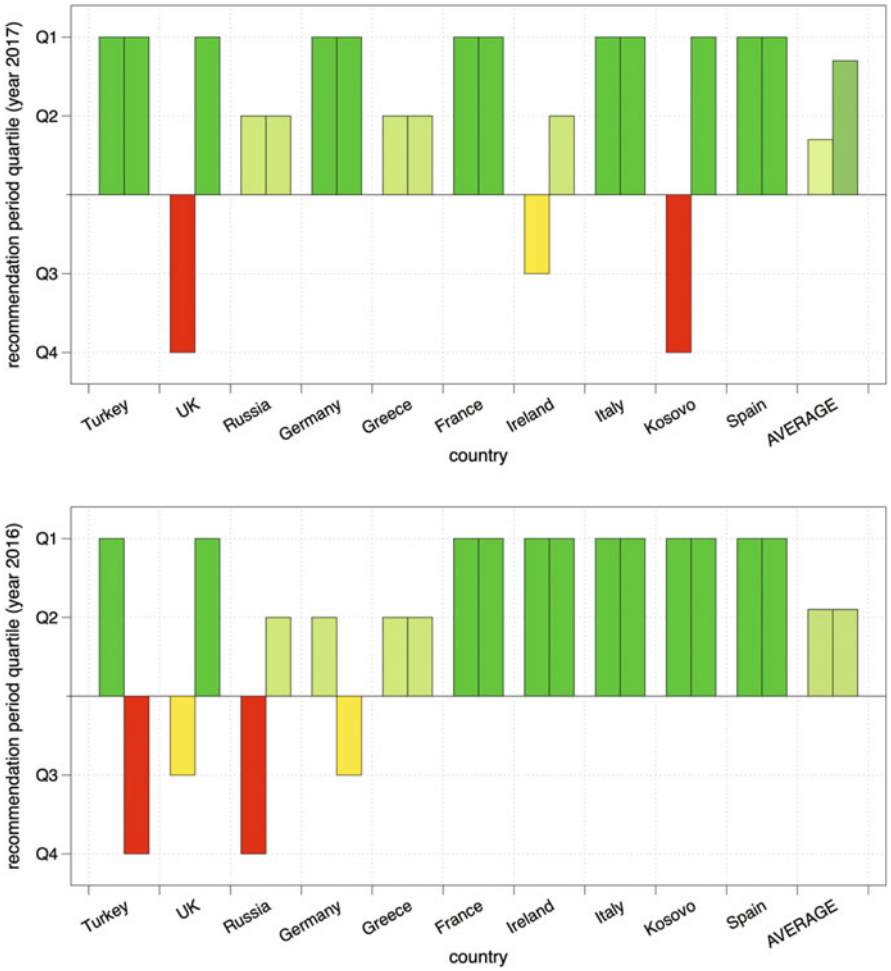


Fig. 8 Recommended period success for various countries when classifying the number of attacks into quartiles. For each country, the left value is the first recommended period and the right value is the second recommended period

Similarly, for the 2015 (two years back) recommendation experiments, Q1 was achieved for 60% of the cases, Q2 for 20%, Q3 for 15% and Q4 for 5%. The results from the randomly selected periods, Q1 and Q2, were achieved for only 60% of the cases.

To examine the cases when non-yearly periodic events (see H3), such as the Summer Olympics, occur, we evaluated the algorithm for 2012 UK and 2004 Greece. Both were Summer Olympics years for the respective countries. The quartiles for the predicted values were Q2 (primary) and Q1 (secondary) for UK and Q1 for both primary and secondary for Greece (Fig. 10). This shows that there

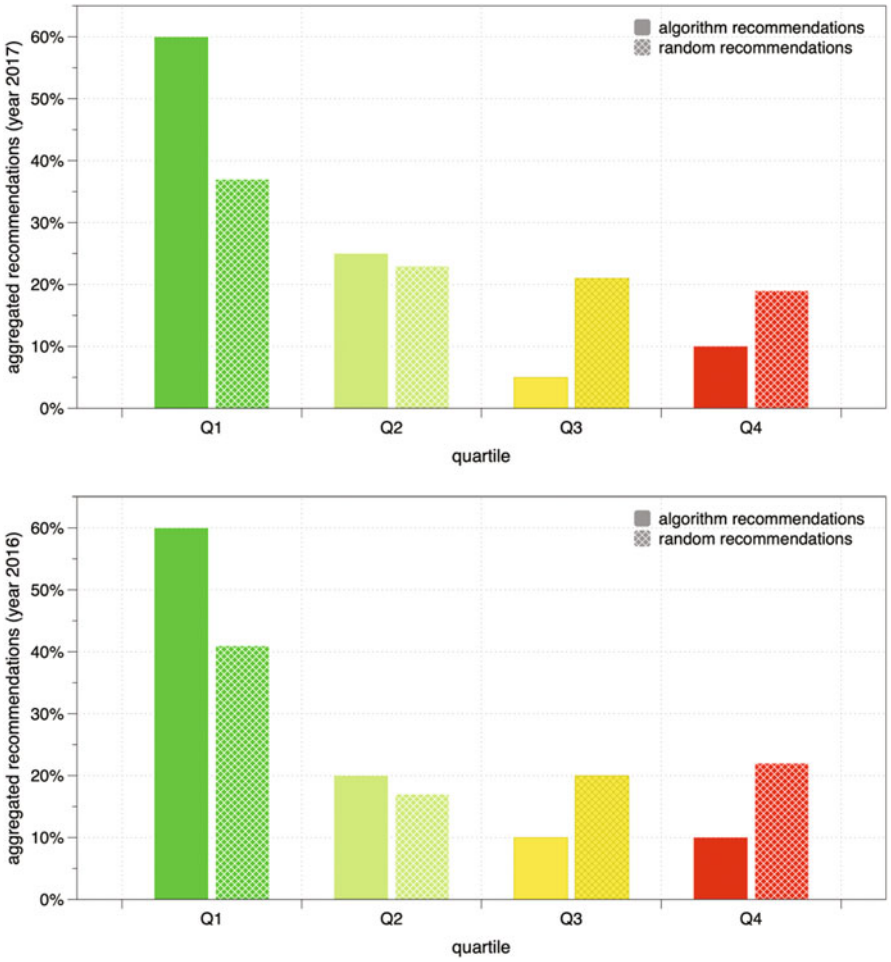


Fig. 9 Aggregated recommendations into quartiles for all countries (attacks)

are no exceptions for the algorithm applicability for special event years, since it resulted in recommendations with high accuracy.

5.2 Number of Fatalities as the Evaluation Parameter

This subsection analyses the results regarding the formulated recommendations' success when the number of fatalities is used as the evaluation parameter.

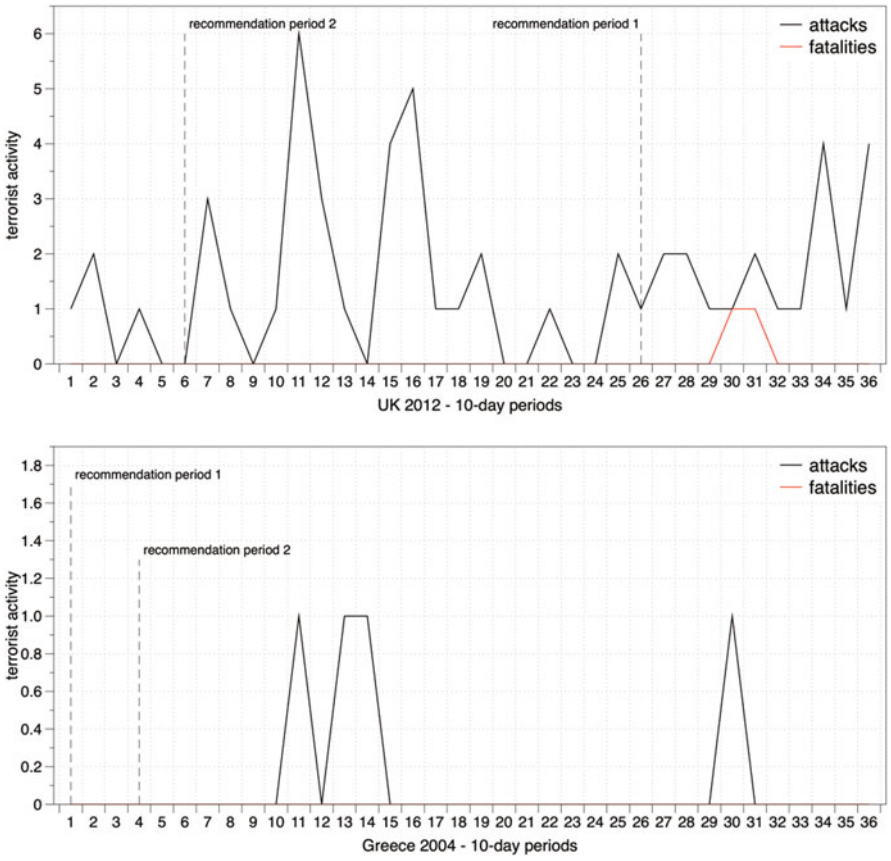


Fig. 10 Recommended periods for UK (prediction year: 2012; predictor: *attacks*) (top) and Greece (prediction year: 2004; predictor: *attacks*) (bottom)

Figure 11 illustrates the results of the experiments regarding the RNoI metric for the same ten countries. The average value of all the countries tested in our experiment is also depicted in Fig. 11.

Regarding the 2017 prediction (top of Fig. 11), we can observe that the average RNoI value of the ten countries is 17% (0% for the first recommended period and 34% for the second), while for the 90% of the recommendations generated, the relative number of fatalities is zero (indicating a successful recommendation). Similarly, for the 2016 prediction (bottom of Fig. 11), the average RNoI value of the ten countries is 16% (2% for the first recommended period and 29% for the second). For the 2015 prediction, the average of RNoI values was 22.5% for the primary prediction and 14% for the secondary. This clearly shows that the algorithm accuracy is retained for the earlier years.

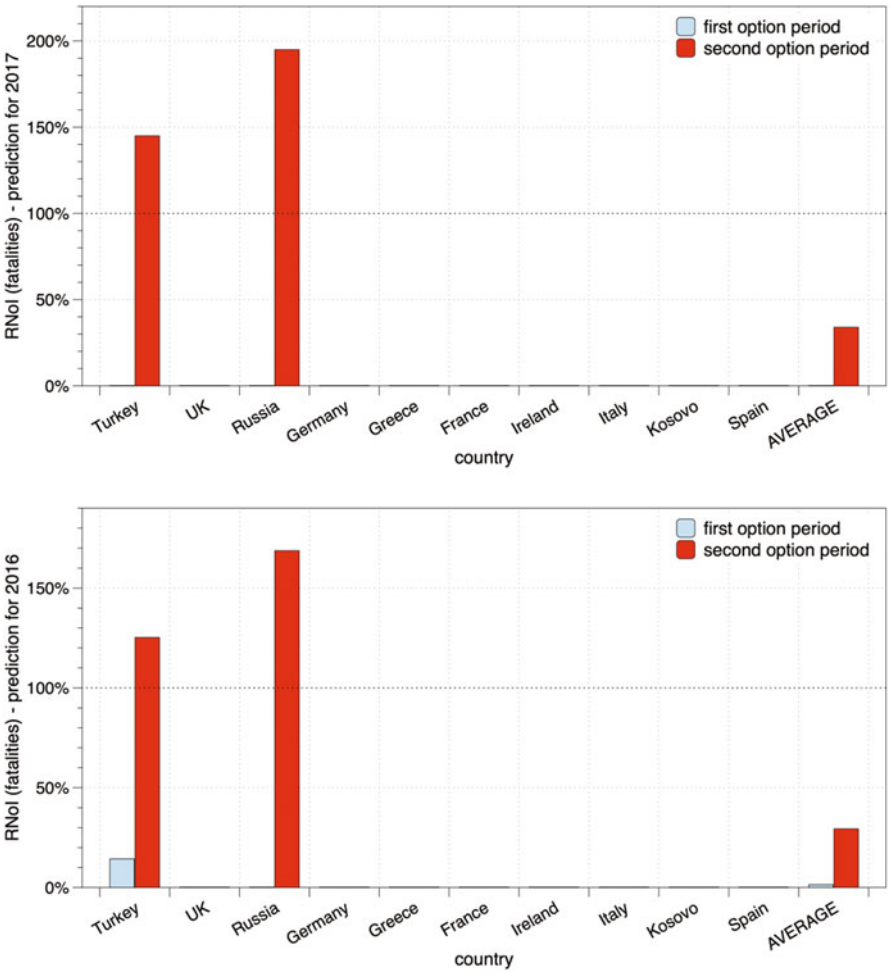


Fig. 11 RNoI results when taking the number of fatalities as the evaluation parameter

More specifically, in the case of France (Fig. 12), which was referred to in the introduction of this chapter, the proposed algorithm recommended, as safe visiting periods, the middle and the end of January, that is, the second and third ten-day period of the year (indicated by the dashed lines).

Figure 13 illustrates the results of the evaluation when considering the quartile that the recommended period belongs to, for the same ten cases. The average value of all the ten countries tested in our experiment is also included in the graph. Figure 14 presents the aggregate results for all the ten countries considered in our experiment for both 2017 and 2016 prediction.

For the 2017 prediction (top of Fig. 14), we can observe that the number of recommendations that are considered successful (Q1) is 90% (18 out of 20

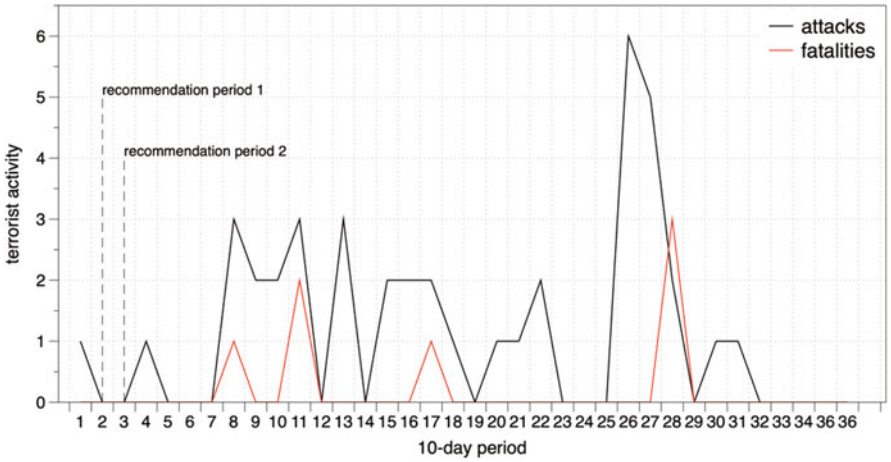


Fig. 12 Recommended periods for France (predictor: *fatalities*)

recommendations), while only the 10% of the predictions (2 out of 20) belongs to Q4 (meaning that the recommendation is considered very unsuccessful, actually recommending a period with high fatality numbers from attacks to the visitor). These cases will be part of our future work. On the same graph, the results from the randomly selected ten-day periods are worse, since less than 75% of the cases achieve a Q1 or Q2 recommendation. The reason for the relatively high accuracy for both the predicted and the randomly selected periods, when compared to the respective accuracy when the 'number of attacks' metric was used as the predictor (Sect. 5.1), is that several countries had registered minimal fatalities; therefore, the random or predicted choice had a higher chance to land on a zero-fatality period, increasing the number of Q1 predictions. In comparison, however, the algorithm prediction was much more accurate than the randomly selected period averages, proving algorithm effectiveness even in low data point situations.

For the 2016 prediction (bottom of Fig. 14), we can observe a similar dispersion. The number of very successful recommendations (Q1) is 90%, while the one from the randomly selected periods is less than 70%.

Similarly, for the 2015 (two years back) recommendation experiments, the percentage of the recommendations belonging to Q1 was 85%, to Q2 was 10% and to Q3 was 5%, proving the algorithm effectiveness when applied to past years.

6 Conclusion and Future Work

Potential visitors to destinations are discouraged by reports on terrorism or fear of terrorist attacks. Destination safety is quite hard to assess since past attacks may continue to be mentioned and still distil fear to local population, businesses and

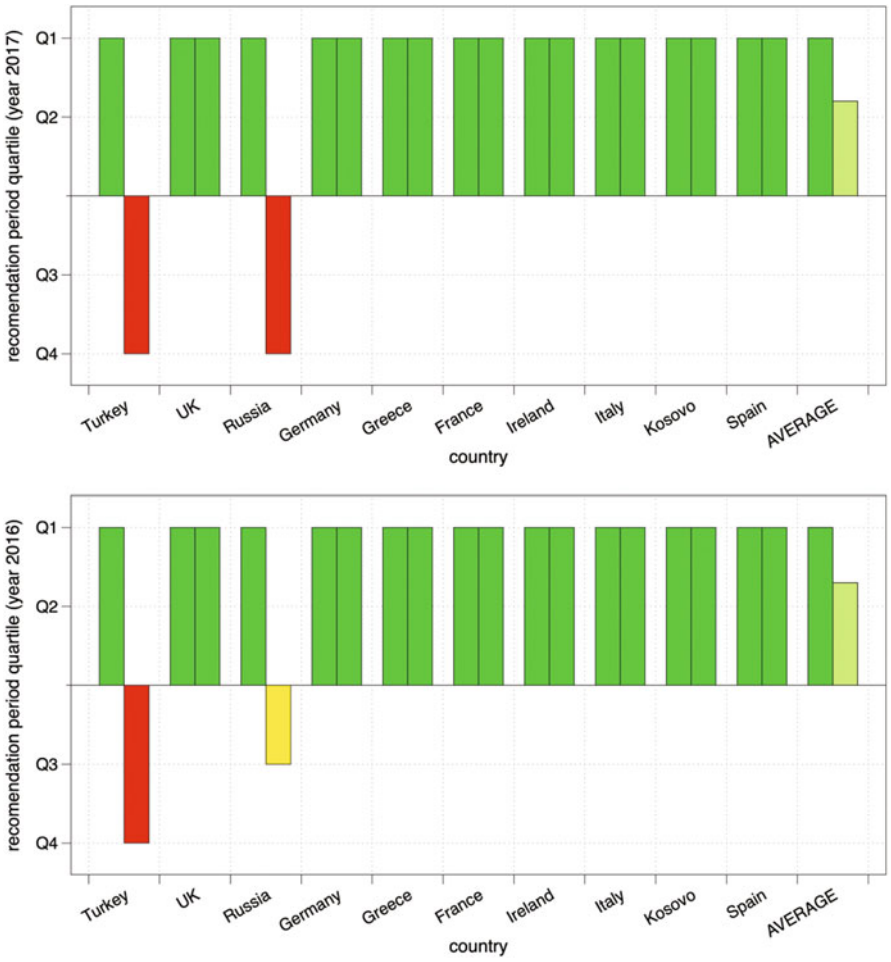


Fig. 13 Recommended period success for different countries when classifying into quartiles the number of fatalities. For each country, the left value is the first recommended period and the right value is the second recommended period

visitors alike, for several years after. A terrorist event results in a state of flux. On the one hand, tour operators and local businesses aim to restore normality and welcome new visitors. On the other hand, visitors, businesses and local governments are uncertain about safety and are looking for ways to assess the situation in the aftermath. The same is true for neighbouring locations, as well.

Perceived safety about a location or a country can be computed through user feedback on social media and is often reflected in the visitor count. Finding ways to measure and predict likeness of terrorist activity in specific time periods would enhance the perceived safety for those periods. Terrorist activity is not uniform over

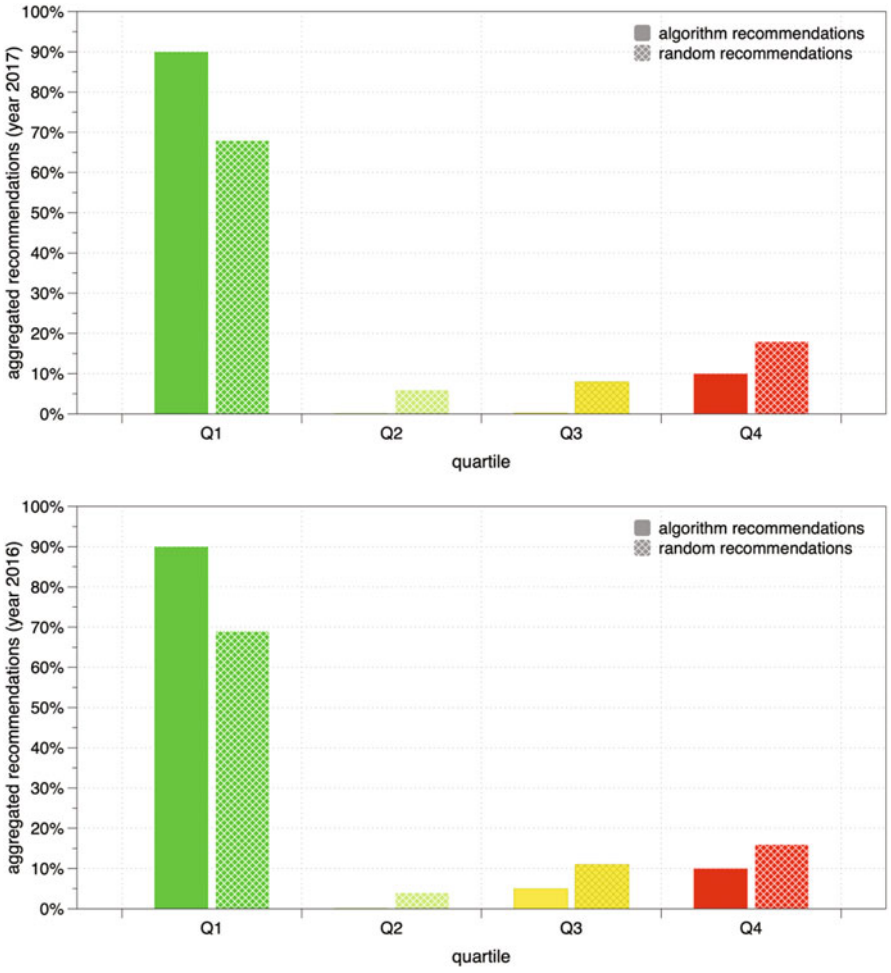


Fig. 14 Aggregated recommendation into quartiles for all countries (fatalities)

time in the span of year. Past data may be used to investigate patterns of activity and predict the safest periods to travel to locations that suffered recent terrorist attacks.

In this chapter, we have used the GTD data from past years and explored through the prism of *attacks* and *fatalities* as the means to understand and predict near-future terrorist activity for European countries. This activity can be approached through these two metrics for recommended periods of relative safety for travellers. Such prediction can help visitors and citizens get an up-to-date, real view of risk and safety, as well as help businesses and authorities plan on rebuilding safety and the perception of the people through resilience [10, 62].

Furthermore, an algorithm that recommends the relatively safe timeslots for travel over as ten-day periods, without season limitations, was proposed. The

proposed algorithm is capable of working with limited information, in our case the number of attacks and fatalities of the past three years, which has been proposed in Spiliotopoulos et al. [55], as a simple yet effective predictor, along with their timestamps. Hence, it can be easily applied to many domains, such as travellers taking pilgrimage to a holy place or a football team travelling to an away game.

The presented algorithm was experimentally validated using ten European countries that had suffered from a large number of attacks in the 2014–2016 timeframe. The evaluation results showed that, for the majority of the cases, the ten-day period recommendations that were produced were considered successful. More specifically, the algorithm was tested using two evaluation parameters, the number of attacks and the number of fatalities, as well as two evaluation metrics, the relative number of incidents and the quartile each recommendation belongs to. In all cases, at least 80% of the recommendations were considered successful, by both metrics, at the same time. Furthermore, in 60% of the cases where the number of attacks was used as evaluation parameters and 90% of the cases where the number of fatalities were used, the recommended ten-day visiting period was considered very successful, since the number of attacks/fatalities was extremely low, for that country, or even zero (as in the case of France).

Through the validation, the algorithm was found to satisfy the original hypotheses set for this work. In regard to H1, the algorithm was found to accurately predict safe periods for travel using both number of attacks and number of fatalities. Especially for the latter, based on the experimental data, the algorithm was able to recommend Q1 safety periods even for countries that had suffered less than a total of 150 number of fatalities for the 2014–2016 period. In regard to H2, the algorithm was evaluated for years 2017, 2016 and 2015, and was found to exhibit the same high accuracy for all three predictions. In regard to H3, the algorithm was evaluated for year–country combinations that included Summer Olympics as a special event and was found to be equally accurate and applicable as it was for the other tested years.

One limitation of this work is on the algorithm applicability to very large countries. Such destinations may suffer terrorist attacks in specific regions, while other areas may be relatively safe. City-level safety prediction requires data on cities and countries to determine relative safety. Another limitation is that opportunistic terrorism targets that do not periodically occur cannot be predicted based on past information. However, identifying similarities with past opportunistic events and clustering them as special events may provide enough information (albeit lacking periodicity) for prediction shifts.

Our future work will focus on utilising social network streams, location features, transport and geographical data for real-time predictions [1, 4, 30, 36]. Geolocation, multimedia and text content [37, 53] can also be analysed for the sentiment of local citizens and visitors for modelling their perception of safety, worry or fear, towards a prospective visit to a country.

References

1. Aivazoglou, M., Roussos, A., Margaris, D., Vassilakis, C., Ioannidis, S., Polakis, J., & Spilitopoulos, D. (2020). A fine-grained social network recommender system. *Social Network Analysis and Mining, 10*(1), 8.
2. Albu, C. E. (2016). Tourism and terrorism: A worldwide perspective. *CES Working Papers, 8*(1), 1–19.
3. Antonakaki, D., Spiliotopoulos, D., Samaras, C. V., Ioannidis, S., & Fragopoulou, P. (2016). Investigating the complete corpus of referendum and elections tweets. In *Advances in social networks analysis and mining 2016 IEEE/ACM conference* (pp. 100–105). IEEE/ACM. San Francisco, CA, USA.
4. Antonakaki, D., Spiliotopoulos, D., Samaras, C. V., Pratikakis, P., Ioannidis, S., & Fragopoulou, P. (2017). Social media analysis during political turbulence. *PLoS ONE, 12*(10), 1–23.
5. Araña, J., & León, C. (2008). The impact of terrorism on tourism demand. *Annals of Tourism Research, 35*(2), 299–315.
6. Asongu, S. A., Nnanna, J., Biekpe, N., & Acha-Anyi, P. (2019a). Contemporary drivers of global tourism: Evidence from terrorism and peace factors. *Journal of Travel & Tourism Marketing, 36*(3), 345–357.
7. Asongu, S. A., Uduji, J. I., & Okolo-Obasi, E. N. (2019b). Tourism and insecurity in the world. *International Review of Economics, 66*(4), 453–472.
8. Burns, P., Lester, J., & Bibbings, L. (2010). Tourism and visual culture. *Methods and Cases., 2*, 1–234.
9. Cavlek, N. (2002). Tour operators and destination safety. *Annals of Tourism Research, 29*(2), 478–496.
10. Cox, A., Prager, F., & Rose, A. (2011). Transportation security and the role of resilience: A foundation for operational metrics. *Transport Policy, 18*(2), 307–317.
11. Ding, F., Ge, Q., Jiang, D., Fu, J., & Hao, M. (2017). Understanding the dynamics of terrorism events with multiple-discipline datasets and machine learning approach. *PLoS ONE, 12*(6), e0179057.
12. Ekstrand, M., Riedl, R., & Konstan, J. (2011). Collaborative filtering recommender systems. *Foundations and Trends in Human-Computer Interaction, 4*(2), 81–173.
13. Eurostat. (2017). *Tourism statistics*. https://ec.europa.eu/eurostat/statistics-explained/index.php/Tourism_statistics. Accessed 21 June 2020.
14. Fuchs, G., Uriel, N., Reichel, A., & Maoz, D. (2012). Vacationing in a terror-stricken destination. Tourists' risk perceptions and rationalizations. *Journal of Travel Research, 52*(2), 182–191.
15. Global Terrorism Index. (2019). http://globalterrorismindex.org/. Accessed 12 Dec 2019.
16. Goldman, O. S., & Neubauer-Shani, M. (2016). Does international tourism affect transnational terrorism? *Journal of Travel Research, 56*(4), 451–467.
17. Guo, W. (2019). Common statistical patterns in urban terrorism. *Royal Society Open Science, 6*(9), 2–13.
18. Gupta, A. (2011). Terrorism and its impact on financial performance: A case of tourism industry. *International Journal of Financial Management, 1*(4), 46–52.
19. Institute for Economics and Peace. (2019). http://economicsandpeace.org/. Accessed 12 Dec 2019.
20. Herlocker, J., Konstan, J., Terveen, L., & Riedl, J. (2004). Evaluating collaborative filtering recommender systems. *ACM Transactions in Information Systems, 22*(1), 5–53.
21. Kalaiarasi, S., Mehta, A., Bordia, D., & Sanskar, D. (2019). Using Global Terrorism Database (GTD) and machine learning algorithms to predict terrorism and threat. *International Journal of Engineering and Advanced Technology (IJEAT), 9*(1), 1–6.
22. Kapuściński, G., & Richards, B. (2016). News framing effects on destination risk perception. *Tourism Management, 57*, 234–244.

23. Khan, S. (2011). Gendered leisure: Are women more constrained in travel for leisure? *Tourismos: An International Multidisciplinary Journal of Tourism, 6*(1), 105–121.
24. Kılıçlar, A., Uşaklı, A., & Tayfun, A. (2018). Terrorism prevention in tourism destinations: Security forces vs. civil authority perspectives. *Journal of Destination Marketing and Management, 8*, 232–246.
25. Korstanje, M. E. (2013). Preemption and terrorism. When the future governs. *Cultura, 10*(1), 167–184.
26. Liu, A., & Pratt, S. (2017). Tourism's vulnerability and resilience to terrorism. *Tourism Management, 60*, 404–417.
27. LaFree, G., & Dugan, L. (2007). Introducing the global terrorism database. *Political Violence and Terrorism, 19*, 181–204.
28. Larsen, S., Brun, W., & Ogaard, T. (2009). What tourists worry about. Construction of a scale measuring tourist worries. *Tourism Management, 30*(2), 260–265.
29. Margaris, D., & Vassilakis, C. (2016). Pruning and aging for user histories in collaborative filtering. In *7th Symposium Series on Computational Intelligence (SSCI) 2016 International Symposium* (pp. 1–8). IEEE. Athens, Greece.
30. Margaris, D., Vassilakis, C., & Georgiadis, P. (2016). Recommendation information diffusion in social networks considering user influence and semantics. *Social Network Analysis and Mining, 6*(1), 108.
31. Margaris, D., & Vassilakis, C. (2017a). Enhancing user rating database consistency through pruning. In *Transactions on large-scale data- and knowledge-centered systems* (Vol. XXXIV, pp. 33–64). Berlin: Springer.
32. Margaris, D., & Vassilakis, C. (2017b). Improving collaborative filtering's rating prediction quality in dense datasets, by pruning old ratings. In *Symposium on Computers and Communications (ISCC) 2017 IEEE* (pp. 1168–1174). IEEE. Heraklion, Greece.
33. Margaris, D., & Vassilakis, C. (2017c). Exploiting internet of things information to enhance venues' recommendation accuracy. *Service Oriented Computing & Applications, 11*(4), 393–409.
34. Margaris, D., & Vassilakis, C. (2018). Exploiting rating abstention intervals for addressing concept drift in social network recommender systems. *Informatics, 5*(2), 21.
35. Margaris, D., Vassilakis, C., & Georgiadis, P. (2018). Query personalization using social network information and collaborative filtering techniques. *Future Generation of Computer Systems, 78*, 440–450.
36. Margaris, D., Spiliotopoulos, D., & Vassilakis, C. (2019a). Social relations versus near neighbours: Reliable recommenders in limited information social network collaborative filtering for online advertising. In *2019 Advances in Social Networks Analysis and Mining (ASONAM 2019) IEEE/ACM International Conference* (pp. 1160–1167). IEEE/ACM. New York, NY, USA.
37. Margaris, D., Vassilakis, C., & Spiliotopoulos, D. (2019b). Handling uncertainty in social media textual information for improving venue recommendation formulation quality in social networks. *Social Network Analysis and Mining, 9*(1), 64.
38. Margaris, D., Kobusinska, A., Spiliotopoulos, D., & Vassilakis, C. (2020). An adaptive social network-aware collaborative filtering algorithm for improved rating prediction accuracy. *IEEE Access, 8*(1), 68301–68310.
39. Meierrieks, D., & Gries, T. (2013). Causality between terrorism and economic growth. *Journal of Peace Research, 50*(1), 91–104.
40. Meng, X., Nie, L., & Song, J. (2019). Big data-based prediction of terrorist attacks. *Computer Electric Engineering, 77*, 120–127.
41. Mo, H., Meng, X., Li, J., & Zhao, S. (2017). Terrorist event prediction based on revealing data. In *Big data 2017 international conference* (pp. 239–244). IEEE. Beijing, China.
42. Pain, R. (2014). Everyday terrorism: Connecting domestic violence and global terrorism. *Progress in Human Geography, 38*(4), 531–550.
43. Palak, A., Mahak, S., & Satish, C. (2019). Comparison of machine learning approaches in the prediction of terrorist attacks. In *Contemporary Computing (IC3) 2019 Twelfth International Conference* (pp. 1–7). IEEE. Noida, India.

44. Pizam, A., & Fleischer, A. (2002). Severity versus frequency of acts of terrorism: Which has a larger impact on tourism demand? *Journal of Travel Research, 40*(3), 337–339.
45. Rai, T. S. (2019). Predicting future terror attacks. *Science, 366*(6467), 834–835.
46. Ranga, M., & Pradhan, P. (2014). Terrorism terrorizes tourism: Indian Tourism effacing myths? *International Journal of Safety and Security in Tourism, 1*(5), 26–39.
47. Samitas, A., Asteriou, D., Polyzos, S., & Kenourgios, D. (2018). Terrorist incidents and tourism demand: Evidence from Greece. *Tourism Management Perspectives, 25*, 23–28.
48. Schefbeck, G., Spiliotopoulos, D., & Risse, T. (2012). The recent challenge in web archiving: Archiving the social web. In *Archives Congress 2012 International Council* (pp. 20–24).
49. Seabra, C., Dolnicar, S., Abrantes, J., & Kastenholz, E. (2013). Heterogeneity in risk and safety perceptions of international tourists. *Tourism Management, 36*, 502–510.
50. Seddighi, H., & Theocharous, A. (2002). A model of tourism destination choice: a theoretical and empirical analysis. *Tourism Management, 23*(5), 475–487.
51. Sönmez, S. (1998). Tourism, terrorism, and political instability. *Annals of Tourism Research, 25*(2), 416–456.
52. Sönmez, S., & Graefe, A. (1998). Influence of terrorism risk on foreign tourism decisions. *Annals of Tourism Research, 25*(1), 112–144.
53. Spiliotopoulos, D., Margaris, D., Vassilakis, C., Petukhova, V., & Kotis, K. (2019a). A methodology for generated text annotation for high quality speech synthesis. In *Information, Intelligence, Systems and Applications (IEEE IISA 2019) 10th IEEE International Conference* (pp. 1–8). IEEE. Patras, Greece.
54. Spiliotopoulos, D., Tzoannos, E., Stavropoulou, P., Kouroupetroglou, G., & Pino, A. (2012). Designing user interfaces for social media driven digital preservation and information retrieval. In: Miesenberger K., Karshmer A., Penaz P., Zagler W. (eds). Computers Helping People with Special Needs. ICCHP 2012. Lecture Notes in Computer Science, vol 7382. Springer, Berlin, Heidelberg. https://doi.org/10.1007/978-3-642-31522-0_87.
55. Spiliotopoulos, D., Vassilakis, C., & Margaris, D. (2019b). Data-driven country safety monitoring terrorist attack prediction. In *2019 Advances in Social Networks Analysis and Mining (ASONAM 2019) IEEE/ACM International Conference* (pp. 1128–1135). IEEE/ACM. New York, NY, USA.
56. The Organisation for Economic Co-operation and Development (2019). http://www.oecdbetterlifeindex.org/topics/safety/. Accessed 12 Dec 2019.
57. Feng, Y., Wang, D., & Yin, Y et al. (2020). An XGBoost-based casualty prediction method for terrorist attacks. Complex Intell. Syst. *6*, 721–740. https://doi.org/10.1007/s40747-020-00173-0.
58. Troian, J., Arciszewski, T., & Apostolidis, T. (2019). The dynamics of public opinion following terror attacks: Evidence for a decrease in equalitarian values from Internet Search Volume Indices. *Cyberpsychology: Journal of Psychosocial Research on Cyberspace, 13*(3), 4.
59. Wolff, K., & Larsen, S. (2014). Can terrorism make us feel safer? Risk perceptions and worries before and after the July 22nd attacks. *Annals of Tourism Research, 44*, 200–209.
60. Xia, T., & Gu, Y. (2019). Building terrorist knowledge graph from global terrorism database and Wikipedia. In *Intelligence and Security Informatics (ISI) 2019 IEEE International Conference* (pp. 194–196). IEEE. Shenzhen, China.
61. Yang, Y., Pah, A. R., & Uzzi, B. (2019). Quantifying the future lethality of terror organizations. *Proceedings of the National Academy of Sciences, 116*(43), 21463–21468.
62. Zemishlany, Z. (2012). Resilience and vulnerability in coping with stress and terrorism. *Israel Medical Association Journal, 14*(5), 307–309.

Analyzing Cyber Influence Campaigns on YouTube Using YouTubeTracker

Thomas Marcoux, Nitin Agarwal, Recep Erol, Adewale Obadimu, and Muhammad Nihal Hussain

Abstract YouTube is the second most popular website in the world. Over 500 hours of videos are uploaded every minute and 5 billion videos are watched every day – almost one video per person worldwide. Because videos can deliver a complex message in a way that captures the audience's attention more effectively than text-based platforms, it has become one of the most relevant platforms in the age of digital mass communication. This makes the analysis of YouTube content and user behavior invaluable not only to information scientists but also communication researchers, journalists, sociologists, and many more. There exists a number of YouTube analysis tools but none of them provide an in-depth qualitative and quantitative insights into user behavior or networks. Towards that direction, we introduce YouTubeTracker – a tool designed to gather YouTube data and gain insights on content and users. This tool can help identify leading actors, networks and spheres of influence, emerging popular trends, as well as user opinion. This analysis can also be used to understand user engagement and social networks. This can help reveal suspicious and inorganic behaviors (e.g., trolling, botting, commenter mobs) that may cause algorithmic

This research is funded in part by the U.S. National Science Foundation (OIA-1946391, OIA-1920920, IIS-1636933, ACI-1429160, and IIS-1110868), U.S. Office of Naval Research (N00014-10-1-0091, N00014-14-1-0489, N00014-15-P-1187, N00014-16-1-2016, N00014-16-1-2412, N00014-17-1-2675, N00014-17-1-2605, N68335-19-C-0359, N00014-19-1-2336, N68335-20-C-0540), U.S. Air Force Research Lab, U.S. Army Research Office (W911NF-17-S-0002, W911NF-16-1-0189), U.S. Defense Advanced Research Projects Agency (W31P4Q-17-C-0059), Arkansas Research Alliance, the Jerry L. Maulden/Entergy Endowment at the University of Arkansas at Little Rock, and the Australian Department of Defense Strategic Policy Grants Program (SPGP) (award number: 2020-106-094). Any opinions, findings, and conclusions or recommendations expressed in this material are those of the authors and do not necessarily reflect the views of the funding organizations. The researchers gratefully acknowledge the support.

T. Marcoux (✉) · N. Agarwal · R. Erol · A. Obadimu · M. N. Hussain
University of Arkansas at Little Rock, Little Rock, AR, USA
e-mail: txmarcoux@ualr.edu; nxagarwal@ualr.edu; rxerol@ualr.edu; amobadimu@ualr.edu; mnhussain@ualr.edu

© The Author(s), under exclusive license to Springer Nature Switzerland AG 2021

M. Çakırtaş, M. K. Ozdemir (eds.), *Big Data and Social Media Analytics*, Lecture Notes in Social Networks, https://doi.org/10.1007/978-3-030-67044-3_5

manipulations. Utility of the YouTubeTracker application is demonstrated via case studies on NATO's 2018 Trident Juncture Exercise and the 2019 Canadian elections.

Keywords YouTube · YouTubeTracker · Social media · Information operations · Bots · Disinformation · Misinformation

1 Introduction

YouTube provides a platform for professionals and amateurs to share their content to a previously unattainable audience. This idea allows the consumer of digital contents to interact with the content they resonate with not only through comments, but also rating and sharing through social media. This turned passive consumers into active actors of the content they enjoy – a major shift in online behavior. One would think such a phenomenon would elicit much curiosity from the data science community, but YouTube has failed to garner as much scrutiny as other social media giants such as Facebook and Twitter – largely due to being a video-based platform. However, there is much to be learned from YouTube data. Besides the longitudinal analysis of content engagement such as traffic patterns, number of likes and dislikes on a specific video over a period of time, comments are a deep source of insight on user behavior. Comments' data can be mined to shed light on user interests, networks and overlaps between communities, or content consumption behaviors.

More importantly, we introduce novel video analysis techniques to gain insight on the mechanisms through which influence is applied in the cyber-space. Efforts to influence opinions are omnipresent in this space and especially active during events which tend to polarize political opinions. YouTube provides a humongous amount of streaming data: over 500 hours of videos are uploaded every minute on average, this number was "only" 300 in 2013 [17]. Because of this sheer volume of data, there is a lack of systematic research that would help us analyze content engagement, user behavior, and more. In an attempt to provide analysts the tools they need to perform various research (behavioral, political analysis, sociology, etc.), we present YouTubeTracker. In the subsequent section, we briefly highlight some of the state of the art technologies in YouTube analysis. We then discuss some of the features and capabilities of our YouTubeTracker application, along with multiple use-cases and new features.

2 State of the Art in YouTube Analysis

YouTube is a video sharing platform that provides an unparalleled ability for hosting and sharing video content [1]. YouTube also provides a great deal of customization, and opportunities to solidify user's branding and content engagement across various platforms [1]. According to Alexa [2], an Internet traffic monitoring

service, YouTube is the second most popular website and accounts for 20% of web traffic. Research by [3, 4] suggests that around 300 hours of videos are uploaded every minute and 1 billion hours of videos are watched each day. Another study by Cha et al. [5] found that 60% of YouTube videos are watched at least 10 times on the day they are posted. The authors in [5] highlight that if a video does not attract viewership in the first few days after upload, it is unlikely to attract viewership later on. Although YouTube provides a means for users to track their content engagement, there is a lack of systematic research on YouTube due to the dearth of analytical tools that can analyze YouTube data. Some of the noteworthy analytical tools for YouTube include: channelmeter [6], vidooly [7], socialreport [8], quintly [9], ranktrackr [10], socialbakers [11], rivaliq [12], cyfe [13], and dasheroo [14]. However, none of these tools provide in-depth qualitative and quantitative insights into various behavioral patterns on YouTube. In a previous publication [15] we identified YouTube as a potential vehicle of misinformation. We proposed the use of YouTube metadata for understanding and visualizing these phenomena by observing data trends. We also analyzed commenter networks through third party tools in an attempt to reveal the main communities revolving around political events as well as their most active actors. In this publication, we expand on this effort by creating a new tool focused on metadata analysis and by proposing the novel use of a video barcode visualization technique to compare and identify wide-spread video clips and their authors.

Recognizing the need for creating useful tools for extracting actionable knowledge from YouTube, we developed YouTubeTracker. Next, we discuss the capabilities of this tool.

3 YouTubeTracker

YouTubeTracker is an application that provides valuable insights in a drilled down version from YouTube data. In this section, we describe some of the features and analytical capabilities of YouTubeTracker, accessible at https://vtracker.host.ualr.edu.

3.1 Tracker Feature

"Trackers" are a concept that help users curate a collection for analysis based on a topic of interest.[1] A tracker could comprise of a set of channels or videos grouped under one topic or theme chosen by the user. Users can feed content of interest to their tracker or dynamically add content they discover while browsing. This allows

[1]This is a common concept across our family of tools such as Blogtrackers (https://btracker.host.ualr.edu).

the user to analyze an ensemble of items and discover overarching patterns. On the tracker dashboard, the user is presented with a bird's eye view of their selected tracker. The total number of videos and channels are displayed, along with the sum of all likes, dislikes, views, subscribers and comments – along with a time frame of the channels' activity. The social media footprint study informs the user on the network of their trackers. The distribution of different social media sites used across the videos and channels of the tracker are displayed in a bar chart.

3.2 Posting Frequency

Posting Frequency reveals posting activity in all or some channels, as well as the top contributors and their location. This can reveal unusual trends, such as large user engagement for a fairly recent channel – which can be a strong indication of bots generating artificial content engagement.

3.3 Content Analysis

Content Analysis provides advanced features such as language and category distribution – as well as prominent comment analysis. This feature also allows users to see most active commenters of a community, their impact on the discourse, or the topics discussed by the community.

3.4 Content Engagement

Content Engagement shows an overview of the type of interaction users have with the selected channels and videos. The line charts display the number of views, likes, dislikes, comments and subscribers over time. This can track viewer's interest in a specific topic over time in an easily understandable manner, which lets analysts measure interest in current events. Trend analysis of content engagement also allows users to detect inorganic activity such as comment mobs, artificial views, likes, etc.

4 Case Study: 2018 Trident Juncture Exercise

In one of our preliminary analyses that leveraged this tool, we analyzed content relevant to NATO's 2018 Trident Juncture exercise that was held October 2018 to November 2018. The tracker consisted of official NATO channels and anti-NATO videos published on YouTube during that period. As shown in Table 1, a total of

Table 1 Data Statistics for NATO's TRIDENT JUNCTURE Exercise (2018) as reported from YouTubeTracker Database

YouTube metrics	Count
Videos	1,324
Views	7,947,124
Likes	169,988
Dislikes	10,624
Comments	28,127
Commenters	15,491
Likes on comments	77,324
Replies to comments	22,014

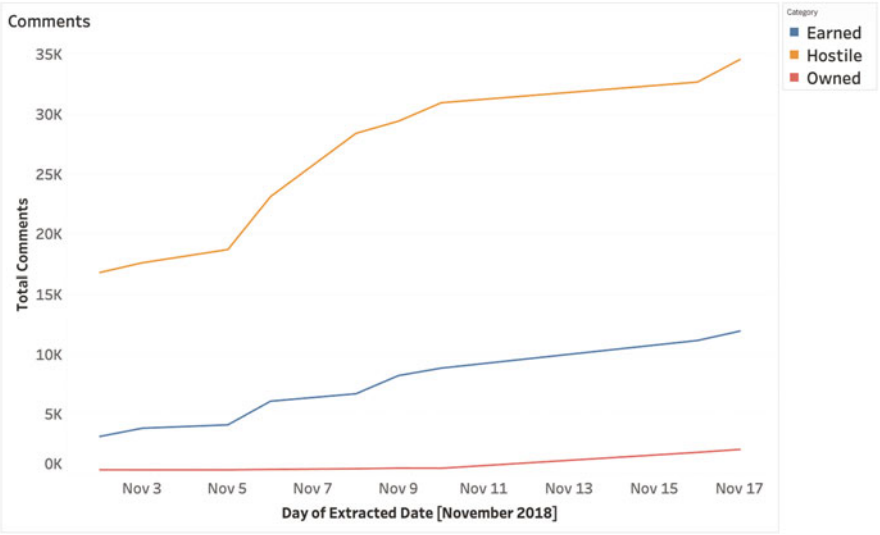


Fig. 1 Comments trends for hostile, earned and owned videos

1,324 videos were analyzed among which 96 videos were categorized as NATO-owned, 390 videos were hostile, and 838 videos were earned – that is, supportive of NATO but not NATO-owned. Table 1 below shows the exact numbers we found.

We sought signs of algorithmic manipulation – usually dense sets of activity like commenting and liking – seeking to promote specific pieces of content to reach organic audiences. We found that hostile videos received higher user engagement (views, comments, etc.) on average than NATO owned or earned videos. While NATO owned and earned videos had entirely organic engagement, hostile videos exhibited strong indications of robotic activities. For instance, not only were most liked and replied-to comments posted on hostile videos (as shown by the trend lines in Fig. 1), but translating high-engagement comments also revealed robotic speech patterns. Google translate was used for Russian, French, and German comments. Resulting in odd sentences that a human translator confirmed to be unusually worded. This could be a case of computer-generated comments.

NATO owned videos had mostly positive comments, whereas, comments on hostile videos had exceptionally high negative sentiment towards the exercise, NATO, and the US. Discussions in Russian language tend to be most liked and most replied. Several of these discussions exhibited strong signs of inorganic or robotic activity – a tactic typically used to drive the content up into YouTube's recommendation algorithms. While most videos were posted on channels located in the United States. Videos posted on channels from Russia were largely hostile.

5 Extended Work

YouTubeTracker is a living and breathing tool that sees constant interaction and evolution. New features and novel analytical tools are periodically added to it and it is used to collect data to conduct analysis of real-world events. One such extension is the use of Elasticsearch to make the YouTubeTracker application scalable.

5.1 Elasticsearch

As previously discussed, one of the major challenges of collecting and storing YouTube data is the sheer volume of data. At the current scale of the project, it is of course impossible for us to track and store metadata information for every existing video. Hence the selective use of trackers where the user directs the tool towards specific videos of interest and focuses on these videos and/or channels only. Despite this first layer of data management, the tool can still experience slowdowns when dealing with very large trackers. Performing diagnosis revealed that these slowdowns are mostly due to interaction with the stored comments, which are exceedingly large. In order to address this issue, we decided to use Elasticsearch indexing [16].

Elasticsearch is a scalable data management solution specializing in handling large volumes of data quickly, suitable for our purposes. To give a quick overview of Elasticsearch, its architecture is divided into nodes, clusters, and indexes, to name the main components. A cluster holds the entirety of our data (divided into N nodes) and can contain as many indexes as we need. For instance, one tracker could be an index, which would let us query the data (namely, the YouTube comments in this scenario) pertaining to our tracker and this data only. This allows YouTubeTracker to be more responsive by utterly ignoring data that is not relevant to our current tracker. We always strive to give more insights through YouTubeTracker but it is also important to make the tool as fast as possible to save time and maximize productivity for our users and analysts.

5.2 2019 Canadian Elections Use-Case

The 2019 Canadian elections study we performed is a good example of using YouTubeTracker for data collection. In some cases, for complex, multi-narratives event such as elections (and this is especially true for a bilingual country), the current tools are not yet refined enough to get the insight analysts are looking for. Or maybe a particular metric has not been included in the tool because it is too specific, and it has been deemed to not provide enough insight for most users. For these reasons, YouTubeTracker also provides a private admin console that lets users perform their own analysis.

Instead of manually looking for every video relevant to a particular event or narrative, we let users divide their desired datasets into trackers. They could be a direct upload of a set of known videos, one or more channel with content of interest, or even a simple keyword search which will periodically add content relevant to the search to the user's tracker. After creating this tracker or appending content to it, data is automatically collected.

For example, an analyst may be performing textual analysis and need text fields during the period of time relevant to the event of interest. The tool lets them select the titles and descriptions of videos over an arbitrary date range. This gives users a great deal of power over what type of data they choose to export and conduct analysis on. Canadian elections are precisely the type of case where this methodology would come in handy. Selecting a time range of videos crawled using political keywords would allow analysts to obtain a precise selection of videos relevant to the elections. From this selection, if analysts only concern themselves with textual data they may produce visualizations such as Fig. 2 and help provide context to their study.

5.3 Video Characterization Using T-SNE and Barcode Visualization

As far as analytical tools are concerned, we have detected one major area of interest to focus our efforts on. A very common scenario in the propagation of misinformation and/or disinformation is the systematic use of the same video clips, usually with a wide range of audio narration added to the video. One concept we need to introduce to address this problem is the video barcode (Fig. 3).

For each video, a barcode is produced using dominant colors in the frame (optionally normalized to adjust for video length). Each video has a unique barcode. This allows us to detect embedded videos as shown in Fig. 4. This is a classic example of the earlier case of different narratives being superimposed on videos through audio narration.

The score given to the videos in Fig. 4 (0.81) is given by the cosine similarity between the two movie barcodes represented as Red-Green-Blue (RGB) vectors. A

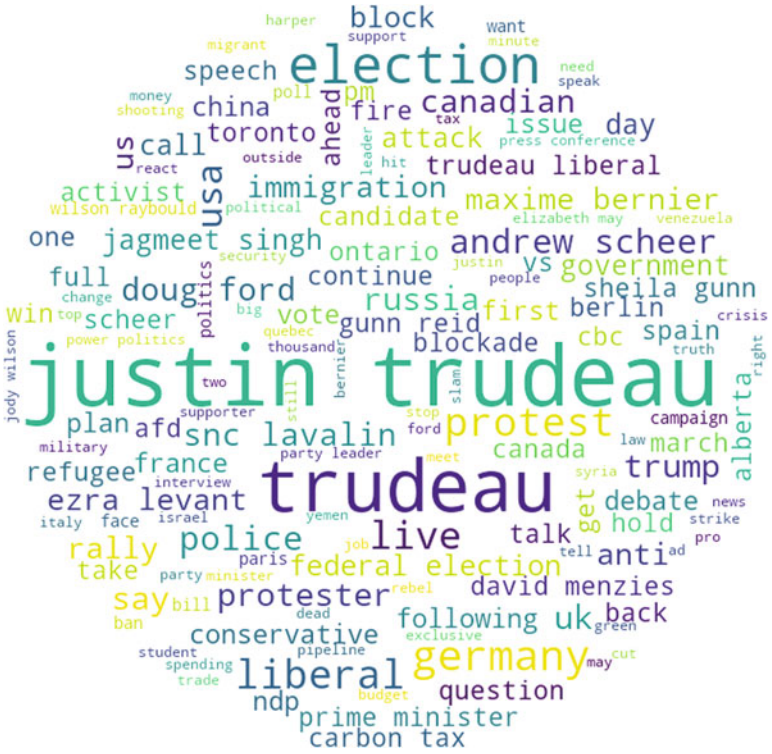


Fig. 2 Canadian Election Wordcloud Generated Using YouTube Data

Fig. 3 Example of a Video Barcode

high similarity score such as 0.81 can alert users that videos share one or more clips and could constitute videos of interest in a misinformation study.

By storing the video barcodes as RGB values and comparing these values using t-SNE, we can detect "neighbor" videos. Conversely, we can plot a graph of every video of interest to visualize their similarity (Fig. 5). In Fig. 5, each data point represents a movie barcode plotted using the T-distributed Stochastic Neighbor Embedding (t-SNE) machine learning algorithm. We select 6 classes with a focus on 2 of them (highlighted in red and blue). In this study, the red class represents barcodes from a series of videos relevant to Asia-Pacific events, or "APAC". The blue class is a series of barcodes made from videos relevant to the NATO's military exercise Baltic Operations ("BALTOPS") held in 2019 in the Baltic region. The remaining four classes are filler classes made from short videos commonly found on YouTube such as cartoons, etc.

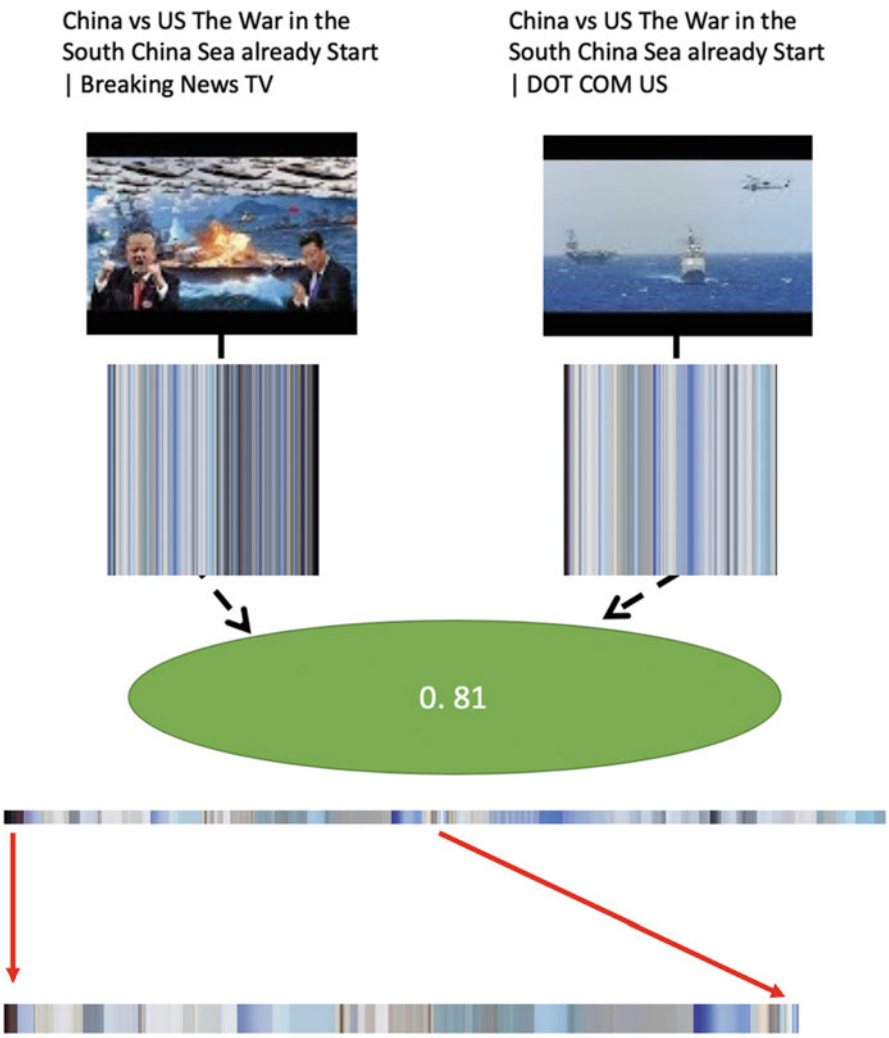


Fig. 4 Example of Political Videos Sharing Identical Clips – Detected by YouTubeTracker Barcode Feature

From this study, we observe that the APAC class (in red) tends to form close clusters. This means that these videos share similar barcodes and could be an indication of video clips being reused, potentially sharing different narratives. This methodology can be applied to trackers within the YouTubeTracker tool. Allowing users to instantly visualize videos that share similar barcodes as they will tend to form clusters.

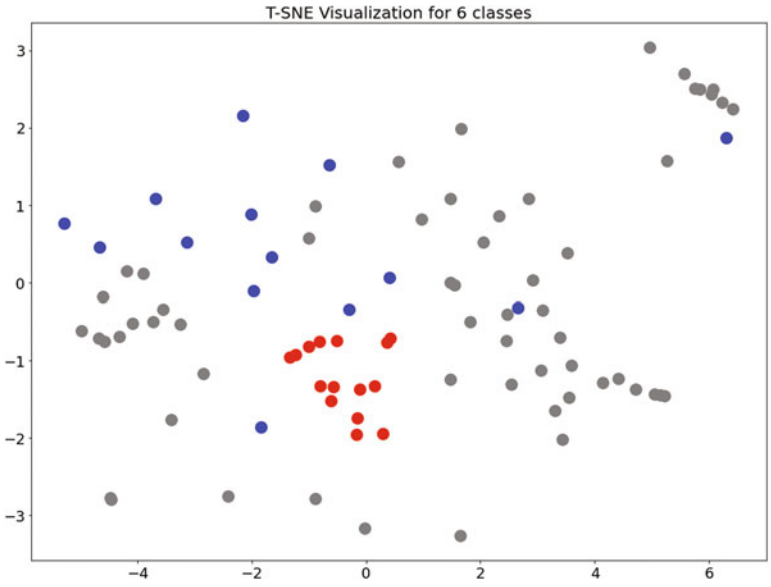


Fig. 5 T-SNE Visualization for 6 classes

6 Conclusion and Future Works

In this chapter, we went through the lack of, and need for, in-depth, quantitative behavior analysis tools for YouTube. Existing tools tend to focus on aggregations of metadata and focus on individual channels. Besides these features, YouTube-Tracker[2] also shows more advanced analytics. In YouTubeTracker, we leverage a variety of tools for performance and analysis. Elasticsearch allows us to perform operations on the extremely large volume of data YouTube generates, while barcode visualization lets us represent the entire length of a video in one image. This will allow for the quick detection of visually similar videos, a valuable tool when detecting mass propaganda campaigns.

To demonstrate the potential of the tool, we showed a real world use-case with the Canadian presidential elections. We also recognize the importance of user-friendly design, as our goal is to provide a valuable tool not only for an academic-oriented audience but also for journalists, political scientists, or even business owners who may wish to develop a deeper understanding of YouTube narratives, trends, and brands.

[2] https://vtracker.host.ualr.edu

References

1. YouTube, L. L. C. (2011). YouTube. Retrieved 27, 2011.
2. Youtube.com Traffic, Demographics and Competitors – Alexa. [Online]. Available: https://www.alexa.com/siteinfo/youtube.com. Accessed 03 June 2018.
3. Youtube Statistics – 2018. MerchDope, 26 Apr 2018.
4. Press, Y. YouTube for Press. YouTube. [Online]. Available: https://www.youtube.com/intl/en-GB/yt/about/press/. Accessed 13 Dec 2017.
5. Cha, M., Kwak, H., Rodriguez, P., Ahn, Y., & Moon, S. (2009). Analyzing the video popularity characteristics of large-scale user generated content systems. *IEEE/ACM Transactions on Networking, 17*(5), 1357–1370.
6. ChannelMeter – Software & Analytics for YouTube MCNs, Creator Networks, and Social Video. [Online]. Available: https://channelmeter.com/. Accessed 02 Jan 2020.
7. Online Video Analytics & Marketing Software. Vidooly. [Online]. Available: https://vidooly.com/. Accessed 02 Jan 2020.
8. Social Media Management Software & Reporting Tools. [Online]. Available: https://www.socialreport.com/. Accessed 02 Jan 2020.
9. quintly GmbH, Social media analytics & competitive benchmarking | quintly. [Online]. Available: https://www.quintly.com. Accessed 02 Jan 2020.
10. RankTrackr – Rank tracker for SEO professionals. RankTrackr.com. [Online]. Available: http://ranktrackr.com/. Accessed 02 Jan 2020.
11. AI-Powered Social Media Marketing Suite – Socialbakers. Socialbakers.com. [Online]. Available: https://www.socialbakers.com. Accessed 02 Jan 2020.
12. Rival IQ: Competitive Social Media Analytics for Digital Marketers. Rival IQ. [Online]. Available: https://www.rivaliq.com/. Accessed 02 Jan 2020.
13. All-In-One Business Dashboard. Cyfe. [Online]. Available: https://www.cyfe.com/. Accessed 02 Jan 2020.
14. Dasheroo – Free Business Dashboards Done Right. Dasheroo. [Online]. Available: https://www.dasheroo.com. Accessed 02 Jan 2020.
15. Hussain, M. N., et al. (2018). Analyzing Disinformation and Crowd Manipulation Tactics on YouTube. In *2018 IEEE/ACM International Conference on Advances in Social Networks Analysis and Mining (ASONAM)*. IEEE.
16. Open Source Search: The Creators of Elasticsearch, ELK Stack & Kibana | Elastic [Online]. Available: https://www.elastic.co/. Accessed 02 Jan 2020.
17. More Than 500 Hours Of Content Are Now Being Uploaded To YouTube Every Minute – Tubefilter. [Online]. Available: https://www.tubefilter.com/2019/05/07/number-hours-video-uploaded-to-youtube-per-minute/. Accessed 02 Jan 2020.

Blog Data Analytics Using Blogtrackers

Adewale Obadimu, Muhammad Nihal Hussain, and Nitin Agarwal

Abstract The use of social media has increased precipitously over the past few years. Despite the emergence of social networking services like Twitter and Facebook, blogging has, although slowly, continued to rise and provides an effective medium for spreading hoaxes and radicalizing content. Individuals also use blogs as a platform to mobilize, coordinate, and conduct cyber campaigns ranging from awareness for diseases or disorders to deviant acts threatening democratic principles and institutions. Today, blogs are increasingly being used to convey mis/disinformation and political propaganda. With no restriction on the number of characters, many use blogs to set narratives then use other social media channels like Twitter and Facebook to disseminate those narratives and steer the audience to their blogs. Hence, blog monitoring and analysis is of great value for public affairs, strategic communications, journalists, and political and social scientists to examine various cyber campaigns. There are a few challenges in blog data analysis. For instance, since blogs are not uniformly structured, the absence of a universal Application Programming Interface (API) makes it difficult to collect blog data for analysis. Similarly, identifying relevant blogs for analysis is also challenging due to the lack of an efficient blog search engines. To facilitate research in this direction, in this chapter, we present the Blogtrackers tool which is designed to explore the blogosphere and gain insights on various events. Blogtrackers can help in analyzing the networks of blogs and bloggers. This tool can also be used to identify influential bloggers, analyze emerging trends, assess tones, and extract key entities in blogs.

Keywords Blog analysis · Blog monitoring · Social media · Blogtrackers

A. Obadimu (✉) · M. N. Hussain · N. Agarwal
University of Arkansas at Little Rock, Little Rock, AR, USA
e-mail: amobadimu@ualr.edu; mnhussain@ualr.edu; nxagarwal@ualr.edu

© The Author(s), under exclusive license to Springer Nature Switzerland AG 2021

M. Çakırtaş, M. K. Ozdemir (eds.), *Big Data and Social Media Analytics*, Lecture Notes in Social Networks, https://doi.org/10.1007/978-3-030-67044-3_6

1 Introduction

Blogging has become a popular means for users to express their opinions on topics that concern or impact them. Typically, blogs are used as an interactive platform for information exchange in that they encourage readers to engage in the conversation via a commenting feature. Blogs also provide a rich medium for content analysis to gain situational awareness about various events. Other social media platforms like Twitter and Facebook can serve as vehicles that drive the audience to blogs and perpetuate discussions. While blogs and other social communication platforms have great democratizing potential, only certain narratives get amplified. In other words, although everyone has the opportunity to have a voice, only a few ultimately get heard. The blogosphere, defined as the network of blogs, is growing at an exponential rate. Users of WordPress, one of the most popular blogging platforms, produce about 70.5 million new posts and 52.1 million new comments each month [1]. According to the latest statistics for WordPress in 2020, over 409 million people view more than 21.1 billion pages each month on WordPress sites [1]. Analyzing blog data helps in understanding the pulse of a society, e.g., knowing what topics resonate within a community, recognizing grievances of particular groups, and assessing situational awareness of citizens during conflicts, crises, and disasters. The lack of character limitations and censorship among the blogosphere allows for blogs to provide an effective platform to set narratives and develop discourse.

Often blogs serve as a way for citizens to gain situational awareness of the socio-political landscape of their environment. In fact, propaganda usually originates on blogs and is strategically disseminated on Twitter, Facebook, WhatsApp, Telegram, Viber, and other social networking platforms. For instance, during the 2016 U.S. Presidential Elections, disinformation was originally posted on blogs published by Macedonian teenagers [2]. Similarly, anti-NATO propaganda has been propagated by pro-Kremlin blogs during NATO exercises [3]. Therefore, identifying the root sources, i.e., propaganda-riddled blogs, is a vital step towards understanding how to tackle disinformation. Using topical-relevance and cyber forensics-based approaches [4], we identify the relevant blogs using automated and semi-automated techniques [5, 6] and expand our search space for propaganda originators. Then the blogs are crawled and indexed in the Blogtrackers (http://btracker.host.ualr.edu/) software for further analysis.

The remainder of this chapter is outlined as follows. First, we introduce Blogtrackers and briefly describe its analytical capabilities. Next, we discuss the state of the art in blog monitoring tools (Sect. 2) and then discuss Blogtrackers' analytical capabilities in detail (Sect. 3). In Sect. 4, we demonstrate Blogtrackers' effectiveness in analyzing blogs targeting the Asia-Pacific region. Section 5 presents our conclusions and future directions for Blogtrackers.

2 State of the Art in Blog Monitoring and Analysis

Due to the immensity of publicly available data in the blogosphere [7], there is a need for creating useful tools for extracting actionable knowledge. Several blog tracking tools emerged in the past, but these attempts have been largely discontinued. Some of the noteworthy tools include – BlogPulse, Blogdex, BlogScope, and Technorati. BlogPulse was developed to provide search and analytic capabilities, automated web discovery for blogs, show information trends, track conversations, and monitor daily activity on blogs. This tool was discontinued in 2012 [8]. Blogdex, another service that has been discontinued, provided a resource for understanding hot-button issues in the blogosphere [9]. BlogScope was another blog tracking service developed as a research project in the department of computer science at the University of Toronto, which provided blog analytics and visualizations but was shut down in early 2012 [10]. Technorati was originally launched as a blog index and a blog search engine. It used a proprietary search and ranking algorithm to provide a directory of blogs sorted by content type and authority. However, it did not provide blog monitoring or analytical capabilities to the end-users. Furthermore, Technorati stopped its blog indexing service in May 2014 [11]. The service now offers an advertising platform. Recognizing the need for a blog-monitoring tool in the research and practitioner community, we developed Blogtrackers, which is discussed next.

3 Blogtrackers: Analytical Capabilities

Blogtrackers is designed to explore the blogosphere and gain insights about events and how these events are perceived in the blogging community. Blogtrackers provides an analyst with a means to develop situational awareness. Although this tool was principally developed using Java technologies, we also leveraged other open source technologies such as elastic search[12] to drastically improve the performance of the application. We utilized the Data-Driven Document [13] as the underlying method for our data visualization. We also utilized the Linguistic Inquiry Word Count (LIWC) [14] to compute the tonality of blogs and bloggers. Finally, we used a technique developed by [15] to obtain the influence scores of blogs and bloggers, respectively. Following are a few features and analytical capabilities of Blogtrackers:

1. A tracker is a collection of blogs selected by a user for analysis. The “Setup Tracker page” allows a user to search for a topic of interest (see Fig. 1) and to select blogs to create a tracker for analysis. A user can create and save any number of trackers
2. The Dashboard gives an overview of the selected tracker (see Fig. 2). It displays the number of blogs, bloggers, blog posts, and the total positive and negative sentiments. It also displays blog sites’ hosting location and language distribution.

Pertinent information such as the trend of keywords, the distribution of blogs and bloggers, the most influential blogs/bloggers, and leading topics of discussion can also be seen on this page.

3. The Sentiments and Tonality feature displays the trend of positive and negative sentiments of blogs for any selected time-period (see Fig. 4). This helps in understanding the effect an event has on the blogosphere. Additionally, the analyst can drill down by clicking on any point of interest and view radar charts displaying tonality attributes such as personal concerns, time orientation, core drives, and cognitive processes.
4. The Posting Frequency feature can be utilized to identify any unusual patterns in blog postings (see Fig. 5). This aids in detecting real-time events that appear to have interested in the blogging community. This feature also displays a list of active bloggers with a relative number of posts. A user can click on any data point on the graph to get a detailed list of the named-entities that were mentioned in blogs during that time period.
5. The Keyword Trends feature provides an overall trend of keywords of interest. It helps track changes in topics of interest in the blogging community. An analyst can correlate keyword trends with events to examine discussion topics and themes relating to that event. The analyst can select any data point on the trend line to view all the blogs. Figure 3 shows the keyword trends related to blogs related to the Asia-Pacific region.
6. The Influence feature helps identify the influence a blogger or blog post has on the blogosphere. Blogtrackers find the posts that are authoritative by assigning a score calculated using the iFinder model [10, 11]. This feature lists the top 5 influential bloggers and displays a trend line to show the variation in bloggers' influence. Clicking on a point on the trend line allows a deeper dive into the data. This feature also provides the capability to visually distinguish between influential and active bloggers (see Fig. 7). Further, a user can explore the content themes of active-influential, inactive-influential, active-noninfluential, and inactive-noninfluential bloggers.
7. The Blog Portfolio Analysis feature provides additional information about a blog. It gives a day-of-the-week average trend of a blog that helps in determining if the blog is a professional blog or a hobby blog (see Fig. 6). Also provided are monthly posting trends and sentiments for the past three years to determine the variation in activity and emotions. A list of URLs and domains mentioned in the blog is provided to allow the user to discover the source of the blogger's information.
8. The Blogger Portfolio Analysis feature shows additional information about a blogger's posting patterns. It gives a day-of-the-week average trend of a blogger that helps in determining if the blogger is a professional blogger or a hobby blogger. Also provided are monthly posting trends and sentiments for the past three years to determine the variation in activity and emotions. A list of URLs and domains mentioned in the blog is provided to show the source of the blogger's information.

9. The Topic Analysis feature provides a means of discovering abstract "topics" that occur in a collection of documents using statistical modeling. This feature allows the user to identify themes of discussions from the blogs. Dynamic analysis of these themes further reveals mainstream or dominant themes, emerging themes, and fading themes in a topic stream style visualization. The topic of stream visualization also allows the user to identify seasonality or periodicity of certain themes.
10. Browser Plugin allows users to add blogs into the Blogtrackers crawling pipeline for further processing. Currently, this feature is available on all leading web browsers such as Chrome, Mozilla, Safari, and Microsoft's Edge browser.

The next section will demonstrate Blogtrackers' analytical capabilities in analyzing blogs in the Asia-Pacific region. We will discuss how Blogtrackers can assist analysts in gaining insights from blog content.

4 Analysis of Asia-Pacific Blogs: A Case Study

In this section, we briefly describe how Blogtrackers can be used to analyze a specific topic of interest. Since blogs serve as platforms for framing narratives, we conducted a few discourse analyses to identify various themes and narratives about the Asia-Pacific region using the blogosphere. The blog tracking process usually starts when a user searches for a keyword. After which the user adds posts of interest to a tracker for further analysis. For instance, there are over 51,000 posts in the Blogtrackers database containing the keyword "Asia-Pacific" (see Fig. 1). Although it seems that the results come from two different blogs, however, this is not the case. The search result displayed in Fig. 1 has been sorted in order of recency. As such, posts from zerohedge.com and southfront are part of the most recent results containing the keyword "Asia-Pacific". Blogtrackers also provide a capability that allows users to sort posts by their influence.

The purpose of the dashboard is to provide high-level information about the currently selected tracker. Figure 2 shows a quick insight into blogs that are posting Asia-Pacific-related content. Surprisingly, most of the blogs in this tracker were hosted in the USA and written in the English language. This insight requires further investigation as it could imply a means of bypassing the strict internet censorship by the government in this region. Despite the initial flatline in the annual posting trend of blogs in this tracker, there was a steady rise in the frequency of posting from 2006 to 2016 suggesting an uptick in interest in Asia-Pacific related content around that period. This increase in blogging activity was due to various external events. For instance, some of the events that dominated Asia's headlines in 2016 include: Donald Trump wins U.S. presidency, Brexit, Rohingya crisis, Duterte's drug war, Aleppo Boy, Blasphemy in Indonesia and the Ahok protests, The Hague ruling on the South China Sea, and the Death of Thai King.

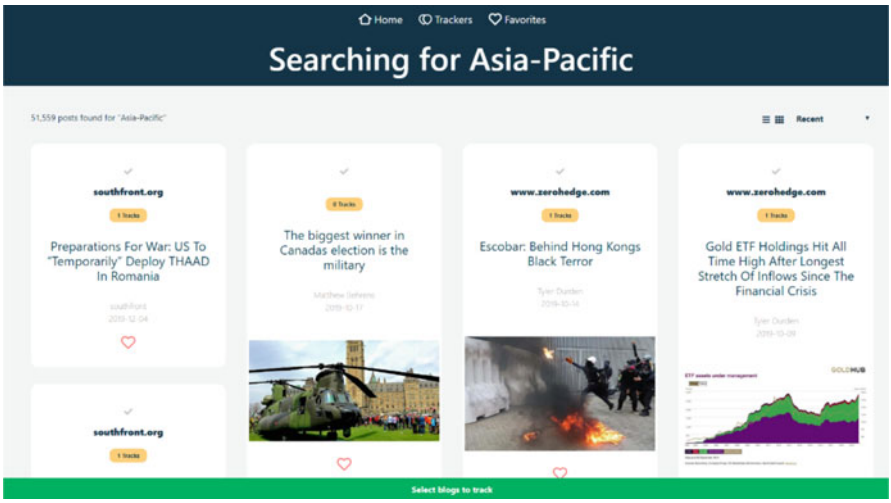


Fig. 1 Search result showing blogs containing Asia-Pacific keyword using Blogtrackers

The keyword trend shows the relative proportion of the frequently used keywords in the currently selected tracker. For instance, the top keywords for this tracker are related to "war", "israel", "trump", "asia" and "russia". Figure 3 shows the usage of one of the top keywords, "russia", relative to another keyword, "america", in the tracker over a period of time. There is a higher usage of the word "russia" suggesting that more Asia-Pacific bloggers are discussing issues that are related to Russia more than America. In essence, the keyword trend provides insight into how a specific keyword is being used contextually in the blogosphere. In most cases, the most active blog will not be the most influential blog. However, in this case, ZeroHedge is the most influential and most active blog. This implication of this finding is that ZeroHedge is very prolific in terms of blog posting and most of the content posted by ZeroHedge regarding the Asia-Pacific region resonates well in the blogosphere. ZeroHedge is described as a "market-focused" blog that presents both in-house analysis, and analysis from investment banks, hedge funds, and other investment writers and analysts [16]. Over time Zero Hedge expanded into the non-financial analysis, advocating what CNN Business called an anti-establishment and conspiratorial worldview, and which has been associated with alt-right views, and a pro-Russian bias.

Figure 4 shows the drilled-down analysis of the aggregate of the positive and negative sentiment of all the posts in this tracker. Starting in 2016, there are several negative reports in the posts. For instance, the first post in Fig. 4 has a title "This is for Allah - Seven Killed In London Terrorist Rampage: Three Assailants Shot". Another post has a title "John McCain Diagnosed With Brain Cancer". These negative reports, among others could be the reason for the higher negative sentiment. Blogtrackers can also be used to perform other detailed analysis. Figure 5 shows the trend of posting activities, top keywords, and posts associated with

Fig. 2 Dashboard for the "Asia-Pacific" tracker

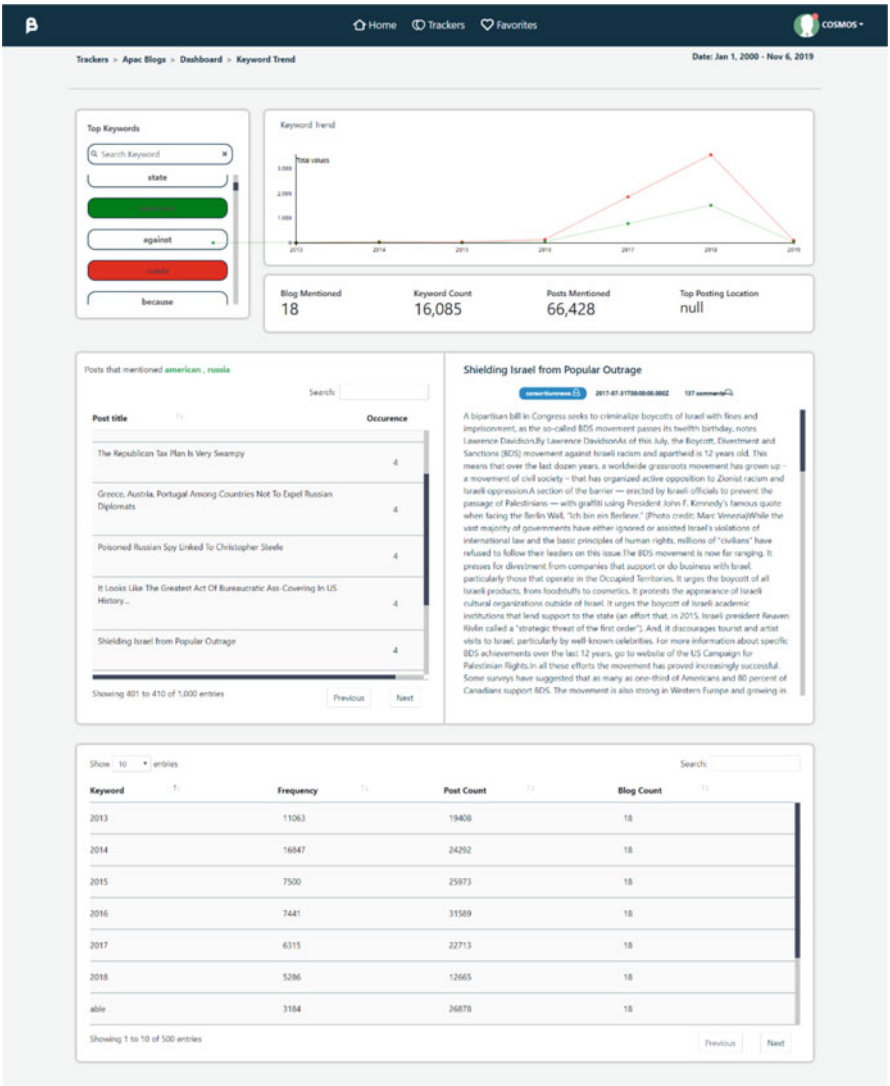


Fig. 3 Keyword Analysis for the "Asia-Pacific" tracker

top bloggers in the tracker. This figure helps users understand the posting pattern of a blogger in the tracker. This feature also allows a comparative analysis of posting trend for two or more bloggers. For instance, despite being a top blogger in the tracker, Tyler Durden has a lower posting frequency in 2015 relative to another blogger, admin, in this tracker. The high posting frequency from various bloggers in 2016 reinforced our earlier analysis suggesting a surge in interest about Asia-Pacific-related activities during this time. One way to understand the

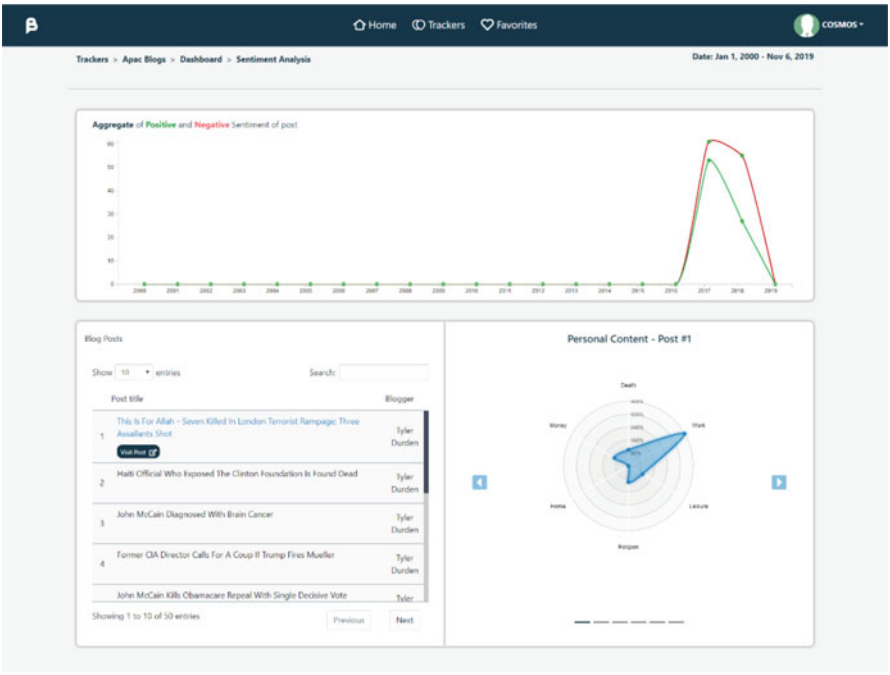


Fig. 4 Sentiment analysis for the "Asia-Pacific" tracker

development of the Blogosphere is to find influential blog sites (Fig. 6). The most influential blog in this tracker is ZeroHedge; whereas, the most influential blogger is consortiumnews. Figure 7 shows the influence trend of "consortiumnews" relative to another influential blogger, "Preston James, Ph.D", for a comparative study. This analysis suggests that posts from consortiumnews resonate more in the blogosphere than posts from Preston James, Ph.D. Fig 7 allows users to gain a deeper insight into the posting habit of blogs and bloggers. For instance, looking at the daily posting pattern of the ZeroHedge blog, we can deduce that this is a professional blog since most of the posts are published during the week (see Fig. 6). A similar analysis can be conducted for bloggers using the blogger portfolio feature. Another interesting feature of Blogtrackers is topic modeling which provides a deep dive into various aspects of topics of interest in a selected tracker. The topics in this tracker are related to "Asia-Pacific" affairs.

5 Conclusion and Future Works

In this chapter, we explained the lack of and need for blog analysis tools. We introduced the Blogtrackers tool, discussed its various analytical capabilities, and demonstrated its effectiveness by presenting an analysis of Asia-Pacific-related

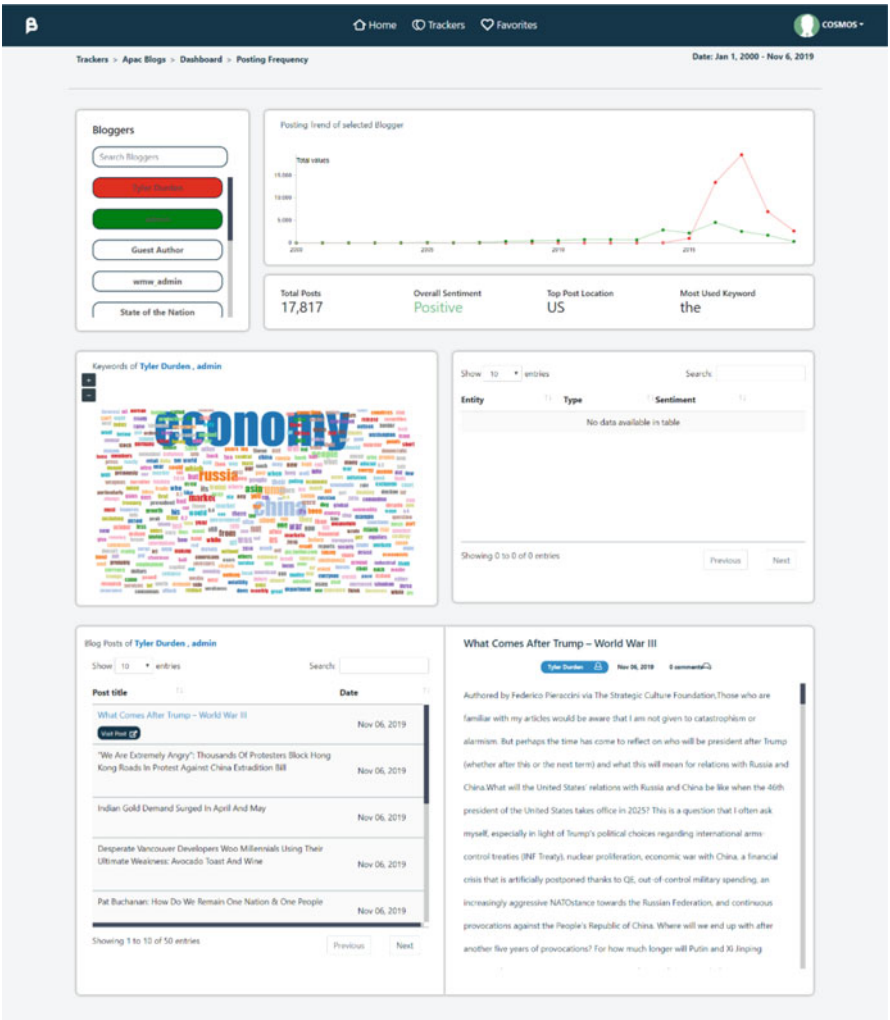


Fig. 5 Posting frequency analysis for the "Asia-Pacific" tracker

blogs. The case study highlighted how Blogtrackers can assist analysts in searching for blogs with related discourse, gaining a bird's-eye view using the Blogtrackers' dashboard feature, and also using the dashboard to dive deep into trends of interest, such as sentiment and opinion trends, posting trends of bloggers, keyword trends, bloggers' influence trends, and discussion theme (or topic) trends. We are constantly improving Blogtrackers and adding new capabilities such as discourse analysis feature to study how narratives evolve over the course of an event. We are also planning on refining our keyword search algorithms to improve the validity of our results. We understand that conversation travels from one channel to another and

Fig. 6 Blog Portfolio Analysis for the “Asia-Pacific” tracker

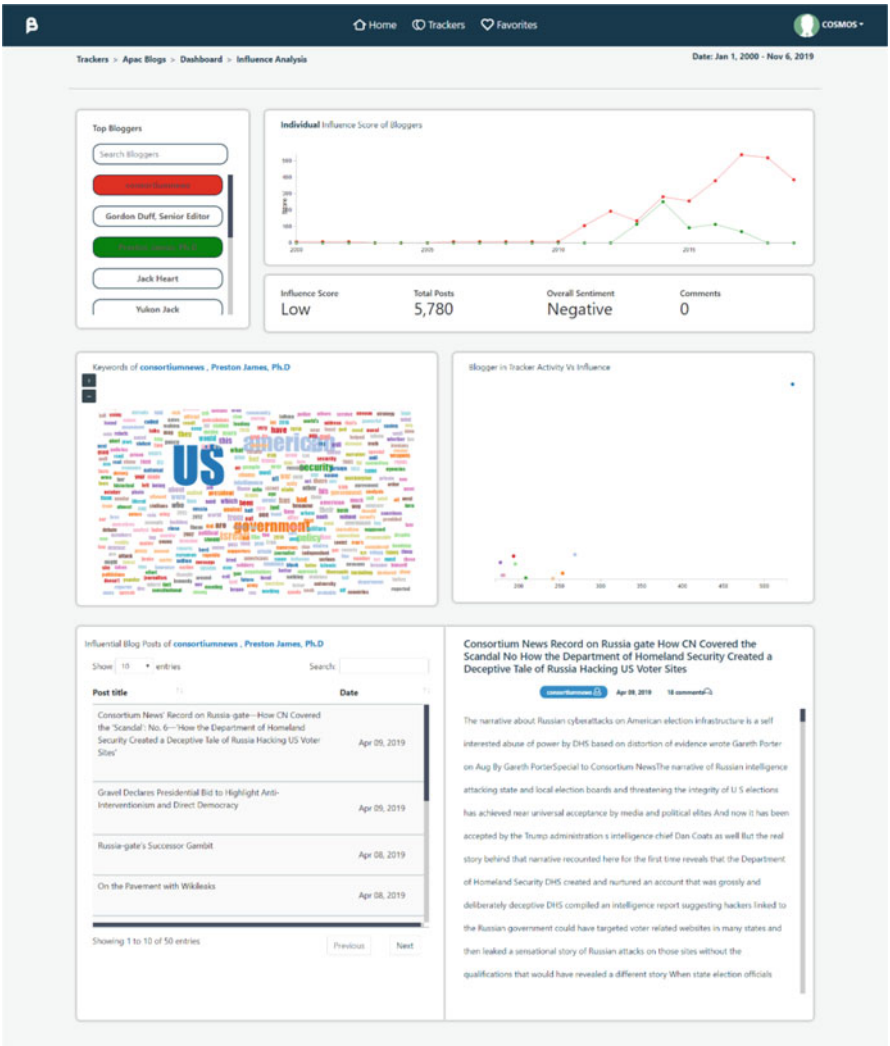


Fig. 7 Influence Analysis for the "Asia-Pacific" tracker

having a feature in Blogtrackers that enables cross-media analysis by building a network of blogs, websites, media sharing sites like YouTube, and other social media websites like Facebook and Twitter would provide richer insights. All these interesting avenues form our immediate research directions.

Acknowledgments This research is funded in part by the U.S. National Science Fsoundation (OIA-1946391, OIA-1920920, IIS-1636933, ACI-1429160, and IIS-1110868), U.S. Office of Naval Research (N00014-10-1-0091, N00014-14-1-0489, N00014-15-P-1187, N00014-16-1-2016, N00014-16-1-2412, N00014-17-1-2675, N00014-17-1-2605, N68335-19-C-0359, N00014-

19-1-2336, N68335-20-C-0540), U.S. Air Force Research Lab, U.S. Army Research Office (W911NF-17-S-0002, W911NF-16-1-0189), U.S. Defense Advanced Research Projects Agency (W31P4Q-17-C-0059), Arkansas Research Alliance, the Jerry L. Maulden/Entergy Endowment at the University of Arkansas at Little Rock, and the Australian Department of Defense Strategic Policy Grants Program (SPGP) (award number: 2020-106-094). Any opinions, findings, and conclusions or recommendations expressed in this material are those of the authors and do not necessarily reflect the views of the funding organizations. The researchers gratefully acknowledge the support.

References

1. https://hostingtribunal.com/blog/wordpress-statistics/gref. Accessed 01 Jan 2020.
2. Subramanian, S. (2017). Inside the Macedonian Fake News Complex. Wired. https://www.wired.com/2017/02/veles-macedonia-fake-news/. Last checked: 24 Dec 2019.
3. Agarwal, N., Kumar Bandeli, K. (2018). Examining strategic integration of social media platforms in disinformation campaign coordination. *Journal of NATO Defence Strategic Communications, 4*, 173–206. https://doi.org/10.30966/2018.RIGA.4.6.
4. Al-khateeb, S., & Agarwal, N. (2019). Deviance in Social Media and Social Cyber Forensics: Uncovering Hidden Relations Using Open Source Information (OSINF). SpringerBriefs in Cybersecurity. Springer. ISBN: 978-3-030-13689-5.
5. Khaund, T., Bandeli, K. K., Walter, O., & Agarwal, N. (2019). A Novel Methodology to Identify and Collect Data from Relevant Blogs Leveraging Multiple Social Media Platforms and Cyber Forensics. ALLDATA 2019, p. 49.
6. Hussain, M. N., Obadimu, A., Bandeli, K. K., Nooman, M., Al-khateeb, S., & Agarwal, N. (2017). A Framework for Blog Data Collection: Challenges and Opportunities. June, 2017.
7. Furukawa, T., Ishizuka, M., Matsuo, Y., Ohmukai, I., Uchiyama, K., & others. (2007). Analyzing reading behavior by blog mining, in *Proceedings of the National Conference on Artificial Intelligence*, vol. 22, p. 1353.
8. BlogPulse. (2017). Wikipedia. 08 Mar 2017.
9. Blogdex. (2016). Wikipedia. 04 Nov 2016.
10. Bansal, N., & Koudas, N. (2007). Blogscope: a system for online analysis of high-volume text streams, in *Proceedings of the 33rd International Conference on Very Large Data Bases*, pp. 1410–1413.
11. "Technorati'the World's Largest Blog Directoryis Gone," Business 2 Community. [Online]. Available: http://www.business2community.com/social-media/technorati-wrlds-largest-blog-directory-gone-0915716. Accessed 19 Dec 2019.
12. Elastic Search system. URL: https://www.elastic.co/. Last accessed on: April 1, 2021.
13. Data Driven Document system. URL: https://d3js.org/. Last accessed on: April 1, 2021.
14. Tausczik, Y. R., & Pennebaker, J. W. (2010). The psychological meaning of words: LIWC and computerized text analysis methods. *Journal of Language and Social Psychology, 29*(1), 24–54.
15. Agarwal, N., Liu, H., Tang, L., & Yu, P. S. (2008). Identifying the influential bloggers in a community, in *Proceedings of the 2008 International Conference on Web Search and Data Mining*, pp. 207–218.
16. ZeroHedge blog website. URL: https://www.zerohedge.com/. Last accessed on: April 1, 2021.

Using Social Media Surveillance in Order to Enhance the Effectiveness of Crew Members in Search and Rescue Missions

Dimitrios Lappas, Panagiotis Karampelas, and Georgios Fessakis

Abstract Social media nowadays are linked almost with every aspect of our lives. They can and have been used to explain social relations, human behaviors, political affections, product preference, just to mention a few applications of social network analysis. Moreover, there are cases in which social media surveillance can be proved valuable for saving human lives as the case which is studied in this book chapter. More specifically, we attempt to test how information collected from social media can improve the ability of a Search and Rescue (SAR) crew to detect people in need using visual search. For the purposes of the study, we simulated a SAR mission in a 3D virtual environment and we asked the study participants to locate refuges needing assistance in different areas of an island. The initial information provided to the volunteers was differentiated and the experiments showed that volunteers who were searching for clues based on input from local social media posts were able to track all the people in need and thus social media surveil-lance was proved to be very promising if it is applied in such cases.

Keywords Social media · Surveillance · Visual search · Search & rescue · Social network analysis

1 Introduction

The continuous regional conflicts, financial crises, climate change, religious persecutions and many other factors the recent years has led several people to abandon the place of their birth and flee in other areas of the world seeking a better life.

D. Lappas · G. Fessakis
Learning Technology and Education Engineering Lab, University of the Aegean, Rhodes, Greece
e-mail: gfesakis@rhodes.aegean.gr

P. Karampelas (✉)
Department of Informatics & Computers, Hellenic Air Force Academy, Dekelia Air Force Base, Attica, Greece
e-mail: panagiotis.karampelas@hafa.haf.gr

© The Author(s), under exclusive license to Springer Nature Switzerland AG 2021

M. Çakırtaş, M. K. Ozdemir (eds.), *Big Data and Social Media Analytics*, Lecture Notes in Social Networks, https://doi.org/10.1007/978-3-030-67044-3_7

Table 1 Arrivals in Europe the last six years [6, 7]

Year	Arrivals	Dead and missing
2019	125,317	1319
2018	141,472	2265
2017	185,139	3139
2016	373,652	5096
2015	1,032,408	3771
2014	225,455	3538

Such a journey turns out to be usually dangerous since it may require travelling through war zones or unfriendly areas and quite often even deaths of displaced people are reported [1, 2]. According to United Nations [3], North America, Europe and Oceania are, most frequently, the destination of those people who seek a better life or try to escape from conflict areas. In this attempt, there are a lot of incidents in which migrants are transported in shipping containers under horrendous conditions resulting in most of the times in their death [4, 5]. In addition, there are several other instances, especially in the Mediterranean Sea, of migrants having drown trying to reach Europe either by crossing the Strait of Gibraltar at the west of Europe, the Libyan Sea at the south or the Aegean Sea at the east. As it can be seen in Table 1, the preferable route to enter Europe is through the sea and as a result every year there are several casualties reported.

In these dangerous routes, more than 19,100 people have died or went missing from 2014 until the end of 2019 while attempting to cross the borders of Europe through the sea. The European Nations and other International Organizations recognizing the problem aim at alleviating the operative events at the countries of origin but also organize Search and Rescue (SAR) missions at the dangerous crossings to help people in danger [8]. The European Union on the other hand being one of the top destinations of migrants and experiencing a large number of migrants fatalities during the borders crossing launched European Union Naval Force – Mediterranean (EUNAVFOR MED) operation Sophia in 2015 [9]. The operation runs until March 2020 to disrupt smugglers' networks which are responsible for the trafficking of people in the Mediterranean Sea and at the same time prevent loss of refugees' lives saving them from the overcrowded and dangerous vessels in which they usually travel. In addition, individual missions by the affected countries are also organized since such incidents occur along the Mediterranean coast-line. These missions are mainly organized by the respective nation authorities most of the times with the participation of non-governmental organizations (NGOs) [10].

In the southern coast of the Mediterranean Sea, there are, at least, nine NGOs which are active helping vessels in need at the maritime borders [10–12]. There are several other smaller NGOs active in the Aegean Sea which track and help refugees in several ways, either by ensuring that they reach the shores safely or by administering first aid or by collaborating with the Greek government and providing humanitarian aid at the refugee settlements [13].

Volunteerism in emergencies [14] organized in NGOs comes as a relief not only to those in need but also to the authorities since quite frequently the authorities'

personnel and the corresponding resources are allocated to multiple tasks and not only to SAR missions. However, if the volunteers are not trained or well-coordinated with the authorities, several problems may arise. For example, in fatal incidents volunteer organizations may get sued either by the family of the untrained volunteers or by the surviving members of the victims' families for not receiving effective guidance and/or assistance [14]. To improve the collaboration between authorities and NGOs, volunteer training becomes a priority especially in countries where the help of such organizations is essential to handle the migration flows. In this line, our work focuses on exploring different techniques in volunteers' training for SAR missions taking advantage of social network surveillance technologies.

Part of the work presented in this book chapter was published in the 2019 IEEE/ACM International Conference on Advances in Social Networks Analysis and Mining in the context of the 4th Social Network Analysis Surveillance Techniques 2019 Workshop [15]. However, the specific chapter extends the initial work by combining three different training approaches that were tested during the experiments and correspondingly studies the new results. The experiment presented was conducted with student volunteers from a Greek Island which is affected by the migration flows. The students were asked to locate refugees in critical medical condition by assuming the part of a rescuer in a SAR mission organized by the Greek authorities. The students depending on the group they were assigned in, received no information, received raw information acquired by local social media or specific instructions based on the information collected by the local social media compiled by intelligent analysts. The results of the experiment showed that social network surveillance can improve the rate of success of a SAR mission when volunteers are involved who are trained and instructed based on intelligent information.

To the best of our knowledge, there is no similar study done which attempts to explore the significance of the information collected from social media in training volunteers that participate in Search and Rescue maritime missions in order to understand how that information is used to visually locate refugees in need in a diverse and hostile environment such as the steep coastline of an island.

The rest of the chapter is organized as follows: Sect. 2 presents how a Search and Rescue mission can be organized and what are the main approaches used when human beings are in danger. Following that, the related work in the area of social media usage in crisis situations is presented and how social media can help in emergency situations is analyzed with the section to conclude with the relevant work in the area of Visual Search and how it can be used in SAR missions. Section 3 describes the problem that the proposed methodology deals with. Section 4 presents the methodology of the work. Section 5 presents and discusses the experimental results and finally Sect. 6 concludes the chapter with the respective out-comes of the work done and highlights future work.

2 Related Work

2.1 Search and Rescue Missions (SARs)

According to United Nations (UN) General Assembly Resolution 57/150 of 16th of December 2002, each country is responsible to assist people in need either victims of natural disasters or of any other emergency occurring on its territory [16]. For that purpose, emergency response capacity should be available in every member of the United Nations organization according to the specific mandate. To achieve this, International Search and Rescue Advisory Group (INSARAG) of United Nations Organization has devised a list of capabilities that each nation should achieve in order to be able to assist people in need. These include search, rescue, medical, management and logistics [16]. Thus, specialized administrative structures have been setup in different countries in order to respond to the needs of people in emergency situations and coordinate and administer the different entities, governmental and not, that operate in natural disaster or when people are in danger. Authorities in this context form first responder teams with the aim to be deployed first in the site of a disaster and help the affected people. Most of the times, the number of this personnel is not adequate depending also on the magnitude of the disaster and thus volunteers are called to play a vital role in this kind of missions.

In general, there are two categories of personnel that participate in SAR missions. The first category includes all professionals, such as helicopter pilots, doctors and specially trained rescue teams. The second category comprises all the volunteers, such as local people who want to participate and contribute to the SAR missions. Those people having no special training most of the times can contribute in a SAR mission mainly in the search domain. On one hand, personnel who has not been trained in SAR missions nor have experience, it is better to remain clear of the scene. On the other hand, it is a benefit if there are many people to act as lookouts. For example, when searching for a man overboard, or when isolating survivors from debris at an aircraft crash site, the more eyes the better. However, the number of people that are needed to watch the area of the disaster depends mainly on the prevailing conditions [17]. If, for example, the area where a passenger has fallen off of a ship is known and in sight then it is not necessary to recruit many volunteers to look out for the person in the sea, whereas if a kid is missing in a wood, the greater number of volunteers seeking out in the woods the better.

Depending on the available equipment and the current conditions in an emergency, lookouts can search with naked eye, binoculars, or Night Vision Goggles (NVGs). Binoculars can be used either for looking near, e.g., when looking through or underneath a bush, or in distance. Similarly, night vision goggles can be used in low vision situations e.g., when light is low either due to the night or smoke and fog depending on their specification. In any case, lookouts should be fully briefed and relieved every 60 min to avoid eye fatigue [17].

The personnel which participate in a search and rescue mission does not mean necessary that they are looking only to find a lost person or a survivor. It is also

useful to find clues related to the lost person or persons in need. For example, footsteps on the sand mean that someone walked in the area. If a team is in the field and discovers a solid clue, then the team has assisted in the search by advancing the last known position. However, it is not so easy to identify clues.

Effective searching requires far more training and practice than most of the people commonly assume [18]. By connecting different clues and positions, it often draws a line directly to where the person in need can eventually be found. Therefore, it is very important for the personnel who operates in the field, either a first responder or a volunteer, regardless of what duty they are assigned, to always remember to watch for clues and report anything they find. In order to better prepare staff and volunteers to build such capacities, training is essential and thus the specific book chapter attempts to better prepare SAR team members to operate more efficiently during their mission by taking advantage of social media surveillance techniques and also immersing them in simulated training in which they will be trained to improve their visual search techniques.

2.2 Social Media in Crisis Situations

The proliferation of social web has changed our social life providing us with the opportunity to instantly participate in events or incidents all over the world using social media at the exact time they are taking place. People tend to share in social media personal experience, opinion or interest, tell jokes, promote their services or products, share their emotions, get support or ask questions [19]. It has been observed that even in emergencies and disasters, several people turn to social media not only to share their condition, post news about the event, but also to list emergency requirements, express their compassion, and discuss the situation [20].

In a recent study [21], the researchers reviewed how social media were used in emergencies and they found that there are mainly two types of users who post in such situations. The people who are affected by the disaster and are on the scene but also remote users who attempt to assist being away from the disaster site and act as a "Virtual Operation Support Team". It is noteworthy, that the use of social media in emergencies started by the general public and the authorities followed [21, 22] posting real time information, recruiting volunteers, asking relief supplies, etc.

A significant example of the usefulness of social media in emergency situations is the one that was recorded by Simon et al. in [21]. In this case, various social media participants immediately after the 2011 Haiti earthquake collected data remotely and created and shared with the local authorities and government, a map with the areas affected and in need. In another similar incident of an earthquake in 2012 in Emilia-Romagna and Lombardy regions of northern Italy, in less than an hour the extend of the disaster was evident through the posts in social media and the precise number of casualties was available to anyone from all over the world to see [23].

There are many other cases, such as hurricane Sandy in the East Coast of the United States (2011), the Oakland and Victoria flooding in Australia (2011), the

earthquake and tsunami in Tōhoku, Japan (2011), the outbreak of dengue fever in Indonesia (2010), where social media were used to inform people about several aspects of the disasters, to facilitate communication between the authorities and people in need and to provide a quick overview of the current situation in the affected areas [23]. The impact of social media in the above incidents attracted re-searchers' attention to this emerging phenomenon and, thus, a more systematic re-view of the role of social media in emergency situations was initiated.

A specialized study was also carried out in Israel [41], trying to directly measure the role of social networks in emergency situations by organizing a mock search and rescue mission inside a University. There were two teams of first responders assigned to find as many people in need as they could. The first team had received information from social media while the second team had no information. The results of the study showed that the first team located "more rapidly and effectively" their targets proving the importance of social media information in such a situation. Our approach is based on a similar setting, however, working with three teams which received no information, information directly from social media and information collected from social media but analyzed by an intelligence specialist. The environment of our study used a computer simulation platform that resembled a real setting and not a protected environment.

Other studies such as in [40] have also studied the significance of social network analysis in SAR maritime missions however again in a different context than our approach. The researchers in this study have proved that social network analysis can assist in the development of a common operational picture in a multi-agency operation. In our case, the maritime SAR operation was the setting that was used to explore how social media surveillance can assist volunteers training.

Furthermore, according to Alexander [23], social media can be used in emergency situations in different fashions. They can be used to listen to the affected people express their sentiments and their opinion; this information can be extremely useful in order to understand the situation in the area of the disaster. Secondly, social media can be used to monitoring the prevalent topics of discussion and reacting upon difficult situations or preventing the spreading of fake news or rumors.

In addition, social media can become part of the emergency planning and disaster response by creating virtual social communities with discrete roles and predefined plans of action in case of emergency [24] such as coordinating the distribution of relief supplies or providing usable information to those in need. However, this should be done beforehand and not during the disaster since at that time, it will not be possible to inform all the community members for their assumed roles and actions. Our proposed approach for surveilling social media takes advantage of both the listening and monitoring functional use of social media to collect information and clues about the refugees reaching the shores in order to assist first responders to locate them and offer them the necessary medical care.

Nevertheless, the information shared in social media does not necessarily convey the actual situation in the area of disaster or in emergency situations. There are examples recorded immediately after major events, such as in the 2010 Chile earthquake, where fake rumors about an upcoming earthquake were posted in social

media intensifying the sense of chaos and insecurity of local people. Another example is a train accident in a densely populated city in Belgium that was carrying toxic gas. Just after the accident inaccurate posts on social media resulted in chaos in the area of the accident since it was not possible for the authorities to verify the real needs of the local population, according to the report of the chief of the fire brigade.

In other cases, such as the Sandy hurricane in 2012 in United States there were a lot of fake photos manipulated by photo editing tools showing catastrophes in other areas which were reposted by hundreds of social media users resulting in being also broadcasted in several television channels creating fake alarms to authorities and causing distress to several people in these areas [23]. In general, even though social media in disasters provide real time data that are freely accessible by everyone and offer a factual representation of the ongoing human activities, usually, there are several negative aspects that impose restrictions on the use of the information posted. For example, according to [21–26], social media posts may have quality and reliability problems, since false or inaccurate information can be posted.

Usually, each social media participant posts events or news usually without any structure or information that may help the first responders to understand the value of the post and react upon it. Several areas, especially those in rural environments, are lacking the necessary infrastructure to support the use of Internet and social media in an emergency and thus the reports in social media may lack information from those areas, and thus misrepresent the actual magnitude of the disaster. Another reported issue is that, most times, social media posts lack spatial information and as a result do not help first responders to plan a targeted mission resulting in losing time trying to locate people in different locations than they actually are.

Researchers, understanding the importance of social media in emergency response, have proposed a number of tools that collect information published in social media with the purpose to assist first responders to react timely in critical situations [27]. Landwehr & Carley [25] have reviewed three existing tools, able to clean and analyze social media data and can potentially be used in emergency situations. These tools are Social Radar by MITRE [28], which mainly identifies social sentiment using social media posts, CRAFT by General Dynamic, which has currently been enhanced and renamed to NetOwl, an intelligent framework for sentimental analysis, geotagging, event extraction and analysis and several other functions [29] and SORACS by Carnegie Mellon University, which currently has evolved to a collection tool which can analyze various dimensions of social networks depending on the interests of the researcher [30] and thus in emergency situations. Since all these tools can be used to collect and analyze social media data and extract useful information from social media, they can be used in emergency situations to collect relevant with the incident information and make them available to first responder teams.

In our case, social media monitoring and surveillance is a key characteristic of the proposed training, since in a local environment such as that of an island, where several refugee arrivals take place almost every day, the detection of people in need is of paramount importance in order to receive them and give them first aid.

Feedback from social media can provide valuable input to the volunteers who are responsible for locating and saving refugees and, thus, our proposal is to combine social media surveillance, intelligence analysis with visual search in SAR mission training in order to improve the search capabilities of SAR team members in order to increase discovery rates of people in need.

2.3 Visual Search Principles & Patterns

Visual search is a fundamental human activity. Several times a day everyone is forced to search. We are searching for the keys of our car, for products in a supermarket or a name in a list. Sometimes visual search is easy, when someone is searching for a red pen among a lot of blue pens on a desk and sometimes search is difficult, when someone is searching for his/her favorite brand store in a mall. The red color of the pen among other blue pens can be easily distinguished but a brand store in a mall in which there are many other stores is not easy to be found. Having though specific information about the location of a target can facilitate visual search even if the target has no distinguishing characteristics [31]. Information that the brand store we are looking for is on the first floor of the mall next to the main lift can narrow down the area of search effectively. In a visual search, a target is the item that must be found. In the course of this study, the task of locating a specific target is considered as visual search.

During visual search process, a large number of optical stimuli are observed. Some of them are the targets and other are distracters, items that you are not looking for and which distracts you from finding the target. In order to find out if any of the observed optical stimulus are the targets of your visual search, a recognition process is followed [32, 33]. It is easy to understand how the recognition process works by the following example. A man walks on a road and he talks to his mobile trying to orient himself for an appointment. He looks around him and he sees a playground on his left and on his right a shop in which there is someone who has a cook's hat on his head and holds yeast. The man describes his position by saying that he is between a playground and a pizza restaurant. He understood that there is a playground next to him because he had seen many playgrounds in the past and he compared the objects in the environment to all the representations found in his memory, until he found the one that matched the stimulus. On the other hand, he understood that there is a pizza restaurant because he analyzed the characteristics of the man he saw and he recognized it by the cognitive processing of these characteristic details which define it. The first recognition method of the above example is the template matching theory and the other one is the feature comparison model [34–37].

Information regarding a specific target can facilitate visual search using any of the above theories. The particular piece of information that the red color pen has a red cap, also helps everyone to recognize it, according to feature comparison model. Information that the brand store we are looking for is on the first floor of the mall next to the main lift also helps us to find exactly the area in which it is located, even

if we do not know where exactly the main lift is located in the specific level of the mall. This happens because we have seen a lift in the past and we can recognize it using our previous knowledge, according to template matching theory.

3 Problem Statement

As presented in the relevant literature, social media monitoring and surveillance play an important role in emergency situations by facilitating information exchange, communication between citizens and the authorities, coordinating different stakeholders, etc. [21–26]. Our idea is to use social network surveillance to enhance volunteers' ability to locate missing people in SAR missions. We set as our target environment the islands in Aegean Sea, where the refugee routes cross and several people have drowned or got lost. We examine whether information collected from social media regarding the position or the behavior of refugees can assist SAR team members to locate them more efficiently than searching in the dark, without any hint. It is expected that in accordance with visual search principles, it will be possible for the crew members to locate easier their target in a complex environment like that of an SAR mission. To confirm that information collected from social media improves the performance of the SAR crew, we ran a simulation experiment with student volunteers who were called to participate in a virtual SAR mission and locate refugees that had recently arrived by boat in an island and needed immediate medical care. More specifically, based on visual search theory and search and rescue mission training we created three groups of students who shared different level of information before the mission. Then, we studied their performance and statistically drew some very interesting conclusions regarding the role of information collected by social media surveillance in the success of the mission. The experimental case study is presented and discussed in the rest of the book chapter.

4 Methodology

4.1 Description

The research methodology that was chosen to confirm our hypothesis is the case study approach. This is considered more suitable for an exploratory research design with the aim of gaining a better understanding of a complex situation under specific conditions [38]. The case study methodology resides in the qualitative/subjective educational research approach which seeks to understand and interpret complex social phenomena in terms of their participants' views, in contrast to the scientific paradigm. In this experiment, students (19–21 years old) attending Aegean University/School of Humanities were invited to participate in a simulation game that

Fig. 1 ARMA III [39]: The view of the island

investigated how information collected from social networks may facilitate visual search in SAR missions. The objective of the study for the participants was to locate all refugees who needed immediate medical care and had landed in different coasts of the island.

4.2 Simulation Platform

In an attempt to explore the influence of the information collected from social network posts to the efficiency of SAR missions, a case study based on a video game, called "Arma III", developed by Bohemia Interactive (https://arma3.com/) [39], has been designed and conducted. The specific game was chosen because it provides an open world editor in which new elements can be added, it is very realistic, and the story play takes place on the Aegean-based islands of Altis (which has been constructed based on the terrain of Lemnos island) and Stratis (which is based on the terrain of Saint Efstratios island which is close to Lemnos island). The missions of the game are not always preplanned and any user can create a new scenario to play and share with other players. Figures 1 and 2 display some representative photos from the game.

Fig. 2 ARMA III [39]: Realistic environment

4.3 Scenario

In an attempt to make visual search more meaningful and to motivate and engage participants in SAR missions, a story-based scenario was designed. The story is evolving on the Aegean island of Saint Efstratios, where Hellenic Coast Guard forces inspected a cargo ship in which crew members revealed symptoms of an unknown disease. During the inspection, the ship's master confessed that among the crew of the ship, there were also refugees who disembarked to the island using boats. Following a medical examination, it was found that the crew members were sick by malaria and apparently the refugees who had disembarked were also sick. Since malaria is a highly dangerous and contagious disease, refugees who had escaped to the island needed to be found quickly for medical care, in order to contain the infectious disease and also prevent the contamination of the local population of the Aegean island. Hellenic Coast Guard announced to the public that everyone who could help as a volunteer in order to locate the refugees in need is welcome. It was also announced that all the residents of the island should remain at their homes until further notice.

Along with the Coast Guard investigations, the Cyber Crime Division of Hellenic Police searched for helpful clues on the internet and found the following social media posts:

- a fisherman who lives on the island posted on an amateur fishing BlogSpot, a photograph of a boat that was abandoned at a beach of the island. Just above the photo he wrote the comment "what a crap! who has left it here?"

Table 2 Description of refugee's cases

Cases	Dead and missing
Case 1	Group of refugees sitting around a glowing fire
Case 2	Group of refugees standing near a boat
Case 3	Group of refugees running on a road
Case 4	Group of refugees moving around a shanty
Case 5	Group of refugees standing near a different shanty
Case 6	Group of refugees standing away from a boat
Case 7	Group of refugees sitting around a smoking fire

- a resident of the island broadcasted live video on Facebook, recorded from his home, showing that somewhere on the island there was smoke. He wanted to warn other people of a fire risk.
- a resident of the island posted on Instagram a photograph taken by a drone, displaying his shanty broken into and in front of it there were scattered items of personal care (clothes, water bottles, packed food, rubbish, etc.). There was also the comment "what happened to my little shanty? what the authorities are doing to protect us from burglars?"

The Aegean University/School of Humanities, aware of the needs of the island society, has decided to contribute to the work of the local authorities on the island by encouraging students to volunteer their services to find the refugees, thus actively contributing to preventing the transmission of an infectious disease. The students' request for voluntary work was approved by the authorities and a student was placed as a crew member on the Coast Guard helicopter, which would search the coasts of the island for refugees.

In our scenario the helicopter was due to fly along the coast of the island for 15 min. The flight path was predetermined and the game player could not change its course. She could only change the view angle (turn right or left, up or down) by using the mouse of the PC. She could also zoom on a specific area, as if she used binoculars, by using the right button of the mouse.

Overall, seven groups of refugees were placed in different places on the island. Two of them were near a boat in the coast, another two were near a fire, two of them were near a shanty and a group was running on a road to hide from the helicopter (Table 2).

5 Experimental Analysis

5.1 Experimental Description

The experiment took place on April 15th, 2019 and November 27th, 2019 in the Laboratory of Learning Technology and Educational Engineering (LTEE) at the School of Humanities, Department of Sciences of pre-school Education and of

Educational Design, University of the Aegean, located in the island of Rhodes in Greece. The participants of the experiment were 42 students of the corresponding department who were attending the third or fourth year of their studies. In the laboratory there was a PC with the "Arma III" software [39] installed and the custom scenario of the SAR mission ready to be executed. The protocol of the experiment presupposed that each student would come into the laboratory individually. The participants then were briefed about the mission they had to accomplish and then they were asked to play the scenario and try to locate the refugees by doing visual search. During the game, an observer, member of the research team, recorded the number of refugees each participant located during the SAR mission and all the comments made. After the experiment a short debriefing took place in which the researcher informed the participants about their performance and thanked them for their valuable help.

5.2 Results

Students that participated in our research were classified in three classes. The first class included those who had no information given about their mission and were asked to find and locate the refugees (class 0), those who had been given raw information from social media surveillance and were asked to find and locate the refugees (class 1) and those who had been given the information from social media surveillance compiled by an intelligent analyst and were asked to find either the refugees or clues related to them (class 2). In the last class the researcher provided specific information for what they are looking i.e., we must find a boat or other clues that could also be provided to land forces in order to approach the area where the refugees may be located. As it was previously mentioned, there were seven cases of refugees scattered in different places of the island. The students had two possible answers for each case during the visual search with the helicopter; either they would find them (Score – 1) or they did not (Score – 0). The performance of all the students participated in the experiment is presented in Table 3 in which the cases found are designated with Score - 1 and those they did not find with Score – 0. In cases 4 and 5 almost all the observers found the refugees and thus we have placed them as the last columns since these two cases show the least significant difference among the classes.

In order to find out if there is any important link between the information given to the participants, the mission of their visual search and the results of their attempt, the Fisher's exact test ($p < 0.05$) was used. The results of the test showed an important link among the information, the mission and the performance of the volunteers in four cases of refugees that were placed on the island (Case 1, Case 2, Case 6 and Case 7).

For example, in case 1, the refugees were placed around a fire in the slope of a mountain. Refugees were placed behind the light of the fire and thus they were not easily detected (Figs. 3 and 4).

Table 3 Volunteers' performance

Observers	Class	Case 1	Case 2	Case 3	Case 6	Case 7	Case 4	Case 5	Results
Observer 1	0	1	1	0	0	1	1	1	5/7
Observer 3	0	1	1	1	1	1	1	1	7/7
Observer 5	0	1	0	1	1	0	1	1	5/7
Observer 7	0	0	1	1	0	1	1	1	5/7
Observer 9	0	1	1	1	1	1	1	1	7/7
Observer 11	0	1	1	0	1	0	1	1	5/7
Observer 13	0	1	0	1	1	0	1	1	5/7
Observer 15	0	1	1	1	1	1	1	1	7/7
Observer 17	0	1	1	1	1	0	1	1	6/7
Observer 19	0	1	0	1	1	0	1	1	5/7
Observer 21	0	1	0	1	1	1	1	1	6/7
Observer 34	0	0	1	0	1	1	1	1	5/7
Observer 35	0	0	0	0	1	0	0	1	2/7
Observer 36	0	1	1	1	1	1	1	1	7/7
Observer 2	1	1	1	0	1	1	1	1	6/7
Observer 4	1	1	1	0	1	1	1	1	6/7
Observer 6	1	1	0	1	1	1	1	1	6/7
Observer 8	1	1	1	1	0	1	1	1	6/7
Observer 10	1	1	1	1	1	0	1	1	6/7
Observer 12	1	1	1	0	1	1	1	1	6/7
Observer 14	1	0	1	1	1	1	1	1	6/7
Observer 16	1	1	1	1	1	1	1	1	7/7
Observer 18	1	1	1	1	1	1	1	1	7/7
Observer 20	1	1	1	1	1	1	1	1	7/7
Observer 22	1	1	1	1	1	1	1	1	7/7
Observer 37	1	1	1	1	1	1	1	1	7/7
Observer 38	1	0	1	1	1	1	1	1	6/7
Observer 39	1	1	0	1	1	1	1	1	6/7
Observer 23	2	1	1	0	1	1	1	1	6/7
Observer 24	2	1	1	1	1	0	1	1	6/7
Observer 25	2	1	1	1	1	1	1	1	7/7
Observer 26	2	1	1	1	1	1	1	1	7/7
Observer 27	2	1	1	1	1	1	1	1	7/7
Observer 28	2	1	1	1	1	1	1	1	7/7
Observer 29	2	1	1	0	1	1	1	1	6/7
Observer 30	2	1	1	1	1	1	1	1	7/7
Observer 31	2	1	0	1	1	1	1	1	6/7
Observer 32	2	1	1	1	1	0	1	1	6/7
Observer 33	2	1	1	1	1	1	1	1	7/7
Observer 40	2	1	1	1	1	1	1	1	7/7
Observer 41	2	1	1	1	1	1	0	1	6/7
Observer 42	2	1	1	1	1	1	1	1	7/7

Fig. 3 The fire on the mountain as it appears from the helicopter

Fig. 4 The same fire using the binoculars

Table 4 Observed and theoretical frequencies (Information/case 1)

	Observed Frequencies			Theoretical frequencies		
Class/Performance	0	1	Total	0	1	Total
0	3	11	14	1.6667	12.3333	14
1	2	12	14	1.6667	12.3333	14
2	0	14	14	1.6667	12.3333	14
Total	5	37	42	5	37	42

Table 5 Significance by cell – Fisher's exact test (Information/case 1)

Class/Performance	0	1
0	>	<
1	>	<
2	<[a]	>

[a]Values for the case displayed in bold are significant at the level a = 0.05

The observed and the theoretical frequencies for case 1 are shown in Table 4.

The comparison between the observed and the theoretical frequencies of the three classes of students for case 1, showed a statistically significant difference in a class (Table 5).

The observed frequency of those who did not find the refugees in case 1 for the participants who had the information about the fire and were looking not only for refugees but also for clues is 0/14 (Table 4), which is significantly lower than the theoretical expected frequency which is 1.6667/14 (Table 4).

During the helicopter's flight it was easy for someone to recognize the light of the fire. The class of the participants, who had the information for the fire somewhere on the island and looked for clues, had better performance on their search. All of them were looking for the light of a fire and used the binoculars to better investigate a cloud or smoke in the island (in other words, they zoomed in areas where there was suspicion of smoke). When they spotted the light of the fire, they said that something suspicious went on in the mountain and asked for land forces to investigate the area. Finally, when they zoomed in the fire, they located the refugees.

In case 2, the refugees were placed near a boat but they could not easily be seen unless someone zoomed in the area (Figs. 5 and 6).

The observed and the theoretical frequencies for case 2 are shown in Table 6.

The comparison between the observed and the theoretical frequencies of the three classes of students for case 2, showed a statistically significant difference in a class (Table 7).

The observed frequency of finding the refugees in case 2 for the participants who had no information about the abandoned boat is 9/14 (Table 6), which is significantly lower than the theoretical expected frequency 11.3333/14 (Table 6).

It was a little hard for the first class of participants (those with no information) to recognize the boat, as it seemed like another rock in the sea. The two other classes of participants who were given the information that a fisherman had posted a

Fig. 5 An abandoned boat as it appears from the helicopter

Fig. 6 The same boat using the binoculars

photograph of an abandoned boat on the coast of the island, had better results in their search. When they identified something that looked like a boat, they immediately zoomed in the area and found the refugees.

Table 6 Observed frequencies and theoretical (Information/case 2)

	Observed frequencies			Theoretical frequencies		
Class/Performance	0	1		0	1	Total
0	5	9	14	2.6667	11.3333	14
1	2	12	14	2.6667	11.3333	14
2	1	13	14	2.6667	11.3333	14
Total	8	34	42	8	34	42

Table 7 Significance by cell – Fisher's exact test (Information/case 2)

Class/Performance	0	1
0	>	**<**[a]
1	<	>
2	<	>

[a]Values for the cases displayed in bold are significant at the level a = 0.05

Fig. 7 The boat as it appears from the helicopter

In case 6, the refugees were also placed near a boat similarly to case 2. However, the refugees were not close to the boat as in case 2, but they had been placed in the hill over the boat and thus even if the boat was located, the participants had to search for the refugees in the area around the boat. Figures 7 and 8 present the whereabouts of the boat as it was seen in the simulation.

The observed and the theoretical frequencies for case 6 are shown in Table 8.

Fig. 8 The same boat using the binoculars. No refugee can be seen close to the boat

Table 8 Observed frequencies and theoretical (Information/case 6)

	Observed frequencies			Theoretical frequencies		
Class/Performance	0	1	Total	0	1	Total
0	2	12	14	1	13	14
1	1	13	14	1	13	14
2	0	14	14	1	13	14
Total	3	39	42	3	39	42

Table 9 Significance by cell – Fisher's exact test (Information/case 6)

Class/Performance	0	1
0	>	<
1	=	=
2	<[a]	>

[a]Values for the cases displayed in bold are significant at the level a = 0.05

The comparison between the observed and the theoretical frequencies of the three classes of students for case 6, showed a statistically significant difference in a class (Table 9).

The observed frequency of not finding the refugees in case 6 for the participants who had the information about the boat and were looking for clues is 0/14 (Table 8), which is significantly lower than the theoretical expected frequency 1/14 (Table 8).

Fig. 9 The smoke when flying with the helicopter

The class of the participants who had information and were looking for clues, found the refugees faster than the others. Once they recognized the boat as a clue, they scanned the area around for more clues, in order to give more accurate details to land forces. As a result, they quickly found the refugees. On the other hand, the participants of the other groups even if they recognized the boat, they spent a few seconds looking for the refugees close to the boat. Then, most of them searched the area in distance from the boat and finally found them. A participant from the first class (those who had no information) could not found the refugees even though she recognized the boat.

In case 7, the refugees were placed near a fire again. In contrast to case 1 there was no obvious light of the fire but there was only a column of smoke in the sky. Figures 9 and 10 presents how the specific scene was setup and viewed by the helicopter and when using the binoculars.

The observed and the theoretical frequencies for case 7 are presented in Table 10.

The observed frequency of finding the refugees in case 7 for the participants who had no information about the fire is 8/14 (Table 10), which is significantly lower than the theoretical expected frequency 11/14 (Table 10). The observed frequency of not finding the refugees in case 7 for the participants who had information about the fire and looked for refugees is 1/14 (Table 10), which is significantly lower than the theoretical expected frequency 3/14 (Table 10).

Fig. 10 The smoke through the binoculars

Table 10 Observed frequencies and theoretical frequencies (Information/case 7)

	Observed frequencies			Theoretical frequencies		
Class/Performance	0	1	Total	0	1	Total
0	6	8	14	3	11	14
1	1	13	14	3	11	14
2	2	12	14	3	11	14
Total	9	33	42	9	33	42

Table 11 Significance by cell – Fisher's exact test (Information/case 7)

Class/Performance	0	1
0	>	**<**[a]
1	**<**[a]	>
2	<	>

[a]Values for the cases displayed in bold are significant at the level a = 0.05

The comparison between the observed and the theoretical frequencies of the three classes of students for case 7, showed a statistically significant difference in two classes (Table 11).

Table 12 Summary for all participants

Parameters	Case 1	Case 2	Case 3	Case 6	Case 7	Case 4
R^2	0.0757	0.0956	0.0202	0.0513	0.1414	0.0250
F	1.5965	2.0610	0.4021	1.0541	3.2118	0.5000
Pr > F[a]	0.2156	0.1410	0.6717	0.3582	0.0511	0.6104

[a]If p(Pr > F) < 0.05 then there is significant effect of information in the performance of the participants to the corresponding case

Table 13 Summary (Means) – Information

Class	Case 1	Case 2	Case 3	Case 6	Case 7	Case 4
2	1.0000	0.9286	0.8571	1.0000	0.8571	0.9286
1	0.8571	0.8571	0.7857	0.9286	0.9286	1.0000
0	0.7857	0.6429	0.7143	0.8571	0.5714	0.9286

The comments of the participants for the specific case were also interesting since the students who had the information from Facebook immediately said "*there is smoke . . . so there is a fire*", "*there is smoke over there . . . it is time to investigate it*", "*there is someone on the mountain . . . maybe there are refugees there*".

It is also interesting to investigate the results of our research with the method of Analysis of Variance (ANOVA). ANOVA is a statistical technique that assesses potential differences in a scale-level dependent variable by a nominal-level variable having 2 or more categories. In this study, there are three classes of participants and seven cases of refugees in a game that they must find. ANOVA test was used to find out if there is a statistically significant relationship between the different classes of the students and the results of their visual search in every case (Tables 12, 13 and Fig. 11). Case 5 has been omitted in the tables and diagrams since all the participants found the refugees and thus there is no difference whether additional information have been used or not.

There is a marginal statistically significant relationship in case 7 (P(Pr > F) = 0,0511). The refugees were placed near a fire and there was a column of smoke in the sky. The information that was given to the two classes of participants was that a resident of the island broadcasted live video on Facebook, recorded from his home, showing that somewhere on the island there was smoke. He wanted to warn other people of a fire risk. This information helped helicopter's crew to have better performance in finding the refugees opposite to the others.

6 Conclusions

Our work has focused on how social media surveillance can facilitate a helicopter crew in Search and Rescue missions in coastal areas, where refugees arrive. Based on a simulation game, we ran a SAR scenario in which one third of the participants

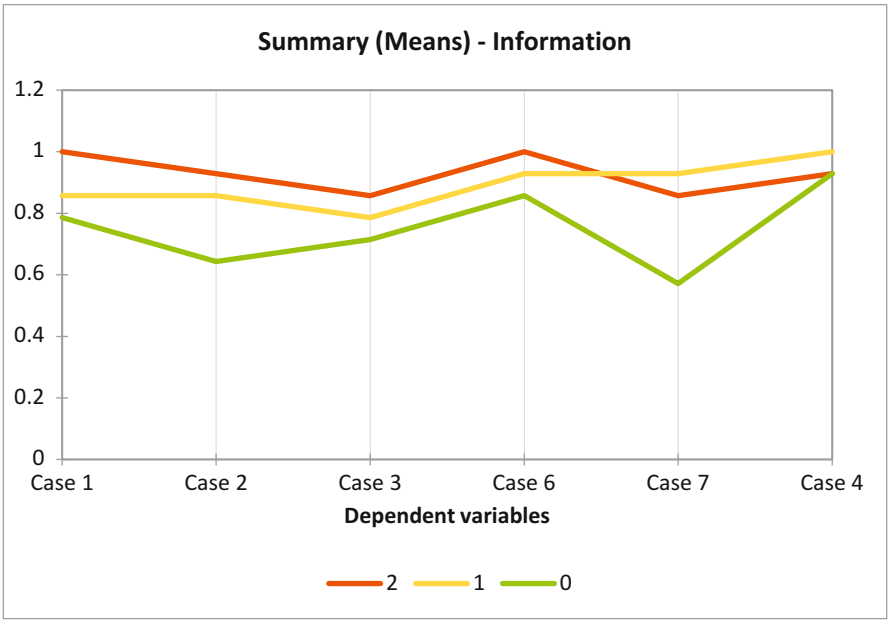


Fig. 11 ANOVA: summary (Means) – Information

had been briefed using information from social media about possible indications of refugee activity, such as smoke or clothes left to dry. The participants had to locate refugees who needed immediate medical care. The other third was provided with the same information and was asked to locate any clues relevant to the information provided and not necessarily the refugees themselves since land forces would later be sent to locate them. The last third were participants who had no information about possible activities of the refugees.

The experimental results showed that the first two groups of participants, those who had been briefed in advance about related posts in social media, had better performance in their visual search than the group who had no information. In other words, the experimental results showed that information collected from social media and related to clues suggesting the position of the targets was useful to the SAR helicopter crew during their visual search and had a positive effect on their performance results. Thus, using social media to collect information regarding a SAR mission has been statistically proven to be more effective than undertaking it without any information and in general can facilitate the rescue of people in need provided that such information is promptly collected, analyzed and communicated to the SAR crew.

The next step to the study is to automate the social media surveillance process using social network analysis techniques such as sentiment analysis, text analytics, social media mining with geotagging, etc. In that way, an early warning system

could be developed that would provide timely information to the rescuers and thus improve survival chances for people in need.

References

1. The Guardian. (2019, May). *New migrant caravan receives cooler welcome in Mexico*. https://www.theguardian.com/world/2019/mar/26/new-migrant-caravan-mexico-cooler-welcome. Accessed 15 May 2019.
2. BBC. (2019, May). *Dozens drown as migrant boat capsizes off Tunisia*. https://www.bbc.com/news/world-africa-48224793. Accessed 15 May 2019.
3. United Nations. (2019). *Trends in international migrant stock 2019*. https://www.un.org/en/development/desa/population/migration/data/index.shtml. Accessed 4 Jan 2020.
4. The Guardian. (2019, November). *Dozens of migrants found in refrigerated container on ferry*. https://www.theguardian.com/uk-news/2019/nov/19/dozens-of-stowaways-found-on-uk-bound-ferry-reports-say. Accessed 4 Jan 2020.
5. Adam, K., & Booth, W. (2019, October). *U.K. police say 39 bodies found in truck container believed to be Chinese*. https://www.washingtonpost.com/world/british-police-39-bodies-found-inside-truck-believed-to-be-chinese-nationals/2019/10/24/15de357e-f644-11e9-ad8b-85e2aa00b5ce_story.html. Accessed 4 Jan 2020.
6. UNHCR. (2020). *EUROPE – Dead and missing at sea: Number of Dead and Missing by Route*. https://data2.unhcr.org/en/documents/download/73199. Accessed 4 Jan 2020.
7. UNHCR. (2019). *Mediterranean Situation: Greece*. https://data2.unhcr.org/en/situations/Mediterranean. Accessed 15 May 2019.
8. Panebianco, S. (2016). The Mediterranean migration crisis: Border control versus humanitarian approaches. *Global Affairs, 2*(4), 441–445.
9. EUNAVFOR MED operation Sophia. (2020). https://www.operationsophia.eu/about-us/. Accessed 04 Jan 2020.
10. Cuttitta, P. (2018). Repoliticization through search and rescue? Humanitarian NGOs and migration management in the Central Mediterranean. *Geopolitics, 23*(3), 632–660.
11. Cusumano, E. (2019). Humanitarians at sea: Selective emulation across migrant rescue NGOs in the Mediterranean Sea. *Contemporary Security Policy, 40*(2), 239–262.
12. Cusumano, E. (2018). The sea as humanitarian space: Non-governmental Search and Rescue dilemmas on the Central Mediterranean migratory route. *Mediterranean Politics, 23*(3), 387–394.
13. Chtouris, S., & Miller, D. S. (2017). Refugee flows and volunteers in the current humanitarian crisis in Greece. *Journal of Applied Security Research, 12*(1), 61–77.
14. Whittaker, J., McLennan, B., & Handmer, J. (2015). A review of informal volunteerism in emergencies and disasters: Definition, opportunities and challenges. *International Journal of Disaster Risk Reduction, 13*, 358–368.
15. Lappas, D., Karampelas, P., & Fessakis, G. (2019). The role of social media surveillance in search and rescue missions. In *Proceedings of the 2015 IEEE/ACM international conference on advances in social networks analysis and mining 2019* (pp. 1105–1111). ACM.
16. INSARAG. (2015). INSARAG Guidelines 2015, Volume II: Preparedness and Response. http://portal.undac.org/pssuportal/portalrest/filesharing/download/public/7FBS4Bt4kuozXvN. Accessed 04 Jan 2020.
17. NTTP 3-50.1. (2009, April). *Navy Search and Resque (SAR) Manual*. Department of the Navy Office of the Chief of the Naval Operations, USA.
18. Study Guide: Search and Rescue Field Certification. (2016). *New Mexico Department of Public Safety Search and Rescue*. Policy Advisory Committee on Education (PACE).

19. Chew, C., & Eysenbach, G. (2010). Pandemics in the age of twitter: Content analysis of tweets during the 2009 H1N1 outbreak. *PLoS One, 5*(11), p.e 14118.
20. Houston, J. B., Hawthorne, J., Perreault, M. F., Park, E. H., Goldstein Hode, M., Halliwell, M. R., Turner McGowen, S. E., Davis, R., Vaid, S., McElderry, J. A., & Griffith, S. A. (2015). Social media and disasters: A functional framework for social media use in disaster planning, response, and research. *Disasters, 39*(1), 1–22.
21. Simon, T., Goldberg, A., & Adini, B. (2015). Socializing in emergencies—A review of the use of social media in emergency situations. *International Journal of Information Management, 35*(5), 609–619.
22. Huang, C. M., Chan, E., & Hyder, A. A. (2010). Web 2.0 and internet social networking: A new tool for disaster management?-lessons from Taiwan. *BMC Medical Informatics and Decision Making, 10*(1), 57.
23. Alexander, D. E. (2014). Social media in disaster risk reduction and crisis management. *Science and Engineering Ethics, 20*(3), 717–733.
24. Rodrigueza, R. C., & Estuar, M. R. J. E. (2018, August). Social network analysis of a disaster behavior network: An agent-based modeling approach. In *2018 IEEE/ACM international conference on advances in social networks analysis and mining (ASONAM)* (pp. 1100–1107). IEEE.
25. Landwehr, P. M., & Carley, K. M. (2014). Social media in disaster relief. In *Data mining and knowledge discovery for big data* (pp. 225–257). Berlin/Heidelberg: Springer.
26. Wang, Z., & Ye, X. (2018). Social media analytics for natural disaster management. *International Journal of Geographical Information Science, 32*(1), 49–72.
27. Teodorescu, H. N. (2015). Using analytics and social media for monitoring and mitigation of social disasters. *Procedia Engineering, 107*, 325–334.
28. MITRE. (2019). *Social radar technologies*. https://www.mitre.org/research/technology-transfer/technology-licensing/social-radar-technologies. Accessed 20 May 2019.
29. General Dynamics. (2019). *NetOwl*. https://www.netowl.com. Accessed 20 May 2019.
30. Carnegie Mellon University, (2019). *Center for Computational Analysis of Social and Organizational Systems (CASOS) Tools*. http://www.casos.cs.cmu.edu/tools/index.php. Accessed 20 May 2019.
31. Posner, M. I., Snyder, C. R. R., & Davidson, B. J. (1980). Attention and the detection of signals. *Journal of Experimental Psychology: General, 109*(2), 160–174.
32. Pashler, H. (1998). *The philosophy of attention*. Cambridge, MA: MIT Press.
33. Posner, M. I., & DiGirolamo, G. J. (1998). Conflict, target detection and cognitive control. In R. Parasuraman (Ed.), *The attentive brain*. Cambridge, MA: MIT Press.
34. Barber, P., & Legge, D. (1976). *Perception and information*. London: Methuen.
35. Lindsay, P., & Norman, D. (1972). *Human information processing*. London: Academic.
36. Neiser, U. (1967). *Cognitive psychology*. New York: Appleton-Century-Crofts.
37. Reynolds, A. G., & Flagg, P. W. (1997). *Cognitive psychology*. Cambridge, MA: Winthrop Publ.
38. Merriam, S. B. (1998). *Qualitative research and case study applications in education* (2nd ed.). San Francisco: Jossey-Bass Publishers.
39. Bohemia Interactive a.s., (2019). ARMA III.
40. Baber, C., Stanton, N. A., Atkinson, J., McMaster, R., & Houghton, R. J. (2013). Using social network analysis and agent-based modelling to explore information flow using common operational pictures for maritime search and rescue operations. *Ergonomics, 56*(6), 889–905.
41. Simon, T., Adini, B., El-Hadid, M., Goldberg, A., & Aharonson-Daniel, L. (2014). The race to save lives: Demonstrating the use of social media for search and rescue operations. *PLoS Curr*. https://doi.org/10.1371/currents.dis.806848c38f18c6b7b0037fae3cd4edc5. PMID: 25685618; PMCID: PMC4322004.

Visual Exploration and Debugging of Machine Learning Classification over Social Media Data

Mayank Kejriwal and Peilin Zhou

Abstract Humanitarian and geopolitical crises (such as COVID-19) are frequently extra-national in scope. Technology, including applications of natural language processing and machine learning, can play a vital role in mitigating this burden, especially with availability and real-time analyses of social media. One such application is *situation labeling*, intuitively defined as the semi-automatic assignment of one or more *actionable* labels (such as food, medicine or water) from a controlled vocabulary to tweets or documents that become available in the aftermath of a crisis, such as an earthquake. Despite multiple advances, users of current situation labeling systems are often unwilling to trust these (and other machine learning) outputs without some provenance and visualization of results. This article describes an interactive visualization approach called SAVIZ that allows non-technical users to intuitively and interactively explore outputs of situation labeling systems. We illustrate the potential of SAVIZ with two real-world crisis datasets from Twitter. Our platform is completely built using open-source tools, can be rendered on a web browser and is backward-compatible with several pre-existing crisis intelligence platforms.

Keywords Crisis informatics · Situation labeling · Visualization · Machine learning · Text classification · Situational awareness · Natural language processing · Embeddings · Representation learning

M. Kejriwal (✉) · P. Zhou
University of Southern California, Los Angeles, CA, USA
e-mail: kejriwal@isi.edu

© The Author(s), under exclusive license to Springer Nature Switzerland AG 2021

M. Çakırtaş, M. K. Ozdemir (eds.), *Big Data and Social Media Analytics*, Lecture Notes in Social Networks, https://doi.org/10.1007/978-3-030-67044-3_8

1 Introduction

In its global humanitarian overview in 2020, the United Nations Office for the Coordination of Human Affairs (OCHA) reported[1] forecast that in 2020, more than 167 million people will need humanitarian assistance. In 2019, many more needed humanitarian assistance than was originally forecast, primarily because of the effects of conflicts and 'extreme climate' events. According to the overview, not only is global climate change increasing people's vulnerability to humanitarian disasters, with the top eight worst food crises all linked to conflict and climate change effects, but infectious diseases are becoming harder to control and are becoming more widespread. This is in addition to persistent problems like malnutrition. At the start of 2019, some 821 million people were under-nourished (according to the same overview), with more than a 113 million suffering from acute hunger.[2]

While donors do step up to the challenge, the massive scale of these problems typically leads to a funding shortfall, and less effectiveness in tackling these issues. One solution that has been touted is *crisis informatics*, which has emerged as an important interdisciplinary area [32], with contributions from both social and computational sciences, including machine learning, information retrieval, natural language processing, social networks and visualization [19, 28, 29, 34, 38]. The key idea behind crisis informatics is to use technological solutions, particularly those powered by information science or even 'Big Data', to help predict disasters and mobilize resources more effectively [9, 26].

To realize this vision in more specific ways, multiple government and private programs have been instituted, some with direct, and others with indirect, salience to crisis informatics. An example of the latter is the DARPA LORELEI program[3] that was established with the explicit agenda of providing *situational awareness* for *emergent incidents*, under the assumption that the emergent incident occurs in a region of the world where the predominant language is computationally *low-resource* [16]. Emergent incidents do not have to be limited to natural disasters, though that they were considered a critical use case. An example of a computationally low-resource language is Uighyur, a Turkic language spoken by about 10–25 million people in Western China, for which few automated technology capabilities currently exist [2]. LORELEI situational awareness systems like THOR [20] must first translate tweets and messages into English, using automated machine translation algorithms, and to provide further analytical capabilities, must execute additional Natural Language Processing (NLP) and Artificial Intelligence (AI)

[1] https://www.unocha.org/sites/unocha/files/GHO-2020_v9.1.pdf

[2] This chapter is an extended version of [23], which was a 4-page demonstration submission at the 2019 IEEE/ACM ASONAM conference in Vancouver, Canada. In this article, we describe the system in depth, including a description of its workflow, full description of its use-cases on real data, links to the system (which has never been released prior) and an updated version that includes time as a control variable.

[3] http://www.darpa.mil/program/low-resource-languages-for-emergent-incidents

algorithms like named entity recognition, automatic detection of *need types* (e.g., does the tweet express a *food* need or a *medical* need?) and sentiment analysis.

Despite advances in NLP and AI, such algorithms continue to be imperfect. For example, we executed a state-of-the-art crisis informatics NLP system called ELISA [4] on an Ebola[4] dataset collected over Twitter. Among other things, ELISA ingests a tweet as input and uses a pre-trained machine learning module developed over the course of the ongoing DARPA LORELEI program to output categorical *situation labels* such as food, medicine, water and infrastructure that allow humanitarian responders to quickly decide where to focus their attention and resources (as opposed to reading every single tweet in the corpus). While for some (pre-processed) tweets such as 'ebola in sierra leone', ELISA correctly outputs the label 'med', it also erroneously outputs labels like 'med' for other tweets like 'vivian dou yemm moh', which have become meaningless due to mangled machine translation or heavy dependence on emoticons and symbols (that get removed during pre-processing). While the modules get better over time, performance is still well below 70% F-measure due to data sparsity and noise. Performance is even worse when the modules are trained on one type of disaster or locale, but have to be applied in another. This is a pervasive problem in crisis informatics, since every crisis is different, making generalization difficult. It also makes humanitarian responders question the veracity of such a system, making transition and uptake of advanced AI technology a social challenge.

In this chapter, we present a highly lightweight, interactive visualization platform called SAVIZ that can be deployed on a web browser in less than 30 seconds for thousands of tweets, and is designed for short, crisis-specific messages collected over social media like Twitter, and processed by systems like ELISA. SAVIZ relies on established, pre-existing and open-source technologies from the representation learning, visualization and data processing communities. SAVIZ is backward-compatible with crisis informatics sub-systems recently released under the LORELEI program, and has been applied on real-world datasets collected from the Twitter API.

2 Related Work

Visualization is an important part of any human-centric system that is attempting to make sense of a large amount of information. Several good crisis informatics platforms that provide visualizations include Twitris [17], CrisisTracker [35], Twitcident [1], TweetTracker [24], AIDR [14], and several others. Some research efforts focus on improving accuracy on a narrow but difficult and important problem, such as extracting information from micro-blogs or determining if a particular

[4]Ebola is a rare and deadly disease that currently has no approved vaccine or cure: https://www.cdc.gov/vhf/ebola/index.html

message is relevant to the disaster in question (from a much bigger stream of messages). Specific examples include work on extracting parcels of information from disaster-related social media messages [15], work on semi-automatic detection of informative tweets during emerging disasters [44], the Twitter-specific case study by Thom et al. [40], among several others [5, 12, 36, 37]. In the last few years, novel advances in AI, including deep learning, have also been explored for crisis response applications. The preprint by Nguyen et al. [30] is a good reference. Finally, work such as [21] address the critical issue of how to acquire Twitter data efficiently in the aftermath of a crisis, especially without paid subscriptions or unlimited API calls.

Another line of work is algorithmic, rather than applied, but is relevant because improvements in some of these algorithms has a direct effect on the functioning and performance of downstream interactive tools and applications. For example, work on discovering geographical topics in the Twitter stream [13] has a direct effect on information extraction and relevance detection. Algorithmic innovations in AI areas such as entity linking [6, 27], event detection [3, 11], crowdsourcing [10, 41], representation learning [18, 25], and sentiment analysis [8] also have consequential effects.

A more sophisticated interactive system, THOR (Text-enabled Humanitarian Operations in Real-time) [20], also provides sophisticated situational awareness, but is designed for computationally low-resource languages like Uighyur and Bengali, and consequently has a stronger focus on NLP tasks like machine translation.

Several aspects of SAVIZ distinguish it from the systems referenced above. The most important difference is that, unlike the systems above, SAVIZ ingests not just the raw social media data stream itself, but the categorical outputs of NLP and machine learning systems like situation labeling and sentiment analysis [33]. Thus, SAVIZ allows the user to jointly explore both the social media data and the labels, which serves two purposes: to understand the noise in the classification system, and to understand the social media stream in aggregate. For example, consider again Fig. 3, which expresses the (initially non-intuitive) finding that 'water' (green) is as big an issue in the context of the collected data, as 'med' (pink), which is what one would expect in a dataset collected from the Twitter API specifically using Ebola-related keywords. Other differences between SAVIZ and systems like CrisisTracker [35] is its use of embeddings and 2D projection (using t-SNE [25]) as an interactive visual aid. As more advanced embeddings (including network and knowledge graph embeddings [39, 42, 43]) continue to be developed and released as open-source, SAVIZ will be well-positioned to use these to provide an alternate 'view' of the data. The current version of SAVIZ is already capable of treating embeddings as a black-box, by directly ingesting the high-dimensional vectors as its input. This allows the system to be lightweight, simple and customizable.

3 SAVIZ: Brief Overview

SAVIZ has a simple processing workflow that is illustrated schematically in Fig. 1. As a first step, the system ingests an input Twitter corpus that has been collected in the aftermath of a crisis. A good system that is capable of such focused data collection is CrisisLex [31]; also, the recent pipeline approach by Gu and Kejriwal is an alternative way of collecting relevant tweets using methods like active learning [21].

Once the corpus has been acquired, an NLP-based situational awareness system like ELISA [4] is typically executed over it. In addition to machine translation and named entity recognition, ELISA also does *situation labeling* on each message.

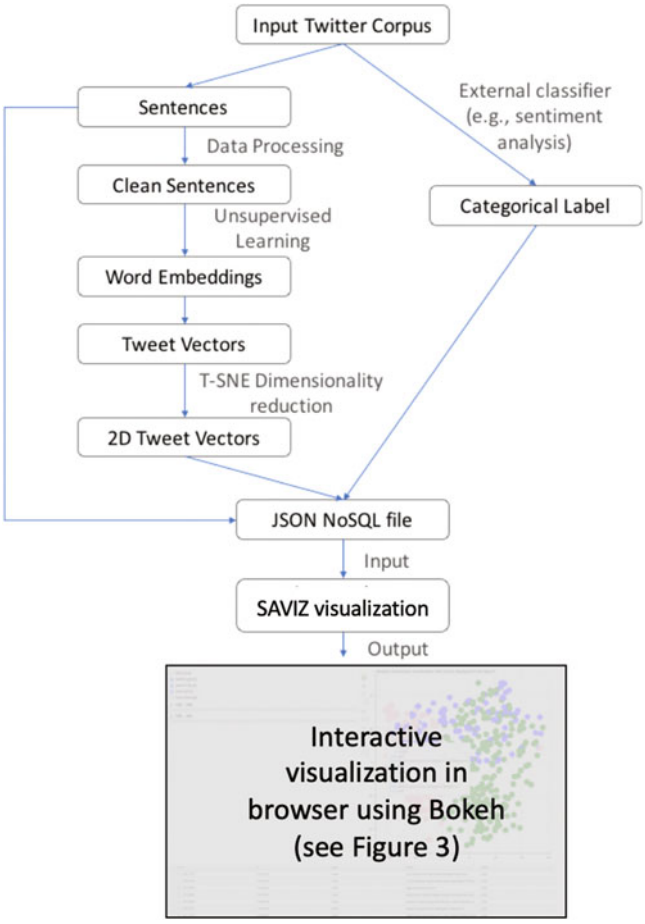


Fig. 1 The SAVIZ workflow over a corpus of Twitter data collected in the aftermath of a crisis. A detailed version of the interactive visualization at the bottom is demonstrated in Fig. 3

Situation labeling is a multi-label problem wherein one or more *situation types*, such as food, medicine, water, evacuation etc. from a small set of pre-defined types, are assigned to each message. However, generic black-box algorithms from the sentiment analysis and classification literature could also be used in this phase. The result of such analysis is one or more *categorical* labels per tweet. For example, for a given message 'Tragic loss of life when dam collapsed. Many more trapped, awaiting rescue', the situation labeling may output 'evacuation' and 'infrastructure' (depending on how the system was trained, and how broad its outputs should be), while sentiment analysis may output 'negative'. Other kinds of classification have also been explored in the literature including *urgency detection* [22]. For example, the message above is arguably high on an urgency scale, since many people could lose lives if the response is not expedient enough. Other messages that may be discussing a temporary power outage or a minor flood warning may not be deemed as urgent. By using these systems, first responders can plan efficient resource and personnel deployment.

The next few steps are unsupervised. The tweets are first preprocessed by converting them to lower-case and removing special symbols and characters (like #, @ etc.), along with tabs and newlines. For example, the tweet 'massive earthquake in NEPAL ——————————— Bhimsen Tower aka Dharahara In Nepal.... http://t.co/4tUDQDWvC4' would, after preprocessing, become 'massive earthquake in nepal bhimsen tower aka dharahara in nepal http://t.co/4tUDQDWvC4' (Last accessed January 2021).

Next, we use the 'bag-of-tricks' *word embedding* package (called fastText [18]) released by Facebook Research to embed the words and sentences into a dense, continuous and low-dimensional (specifically, 100-dimensional) vector space. Word embedding algorithms, which are a more specific sub-field of the more general area of *representation learning*, have been extremely influential in NLP for the last ten years, leading to improvements in performance across multiple NLP problems without necessitating domain-specific feature engineering [7]. More recently, the impact of word embeddings and other kinds of embeddings have percolated into multiple communities that rely on text and image analytics. Social media analytics and crisis informatics are good examples, including systems such as SAVIZ. In previous work, for instance, we used these word embeddings to derive vectors for hashtags, and found that simply exploring hashtags in a 2D space (described further below) can yield important insights about disasters (Fig. 2), such as the tragic mass shooting in Las Vegas towards the end of 2017.

To enable visualization, we use t-SNE[5] [25] to project the sentence vectors into a 2D space. The t-SNE algorithm has achieved tremendous impact as a standard embedding-visualization tool in the machine learning community (and also beyond). In principle, other dimensionality reduction tools existed before t-SNE was proposed, but by using a neural network for its optimization, t-SNE is able to achieve experimentally superior visualization wherein points that are close in the

[5] Stands for *t-Distributed Stochastic Neighbor Embedding*.

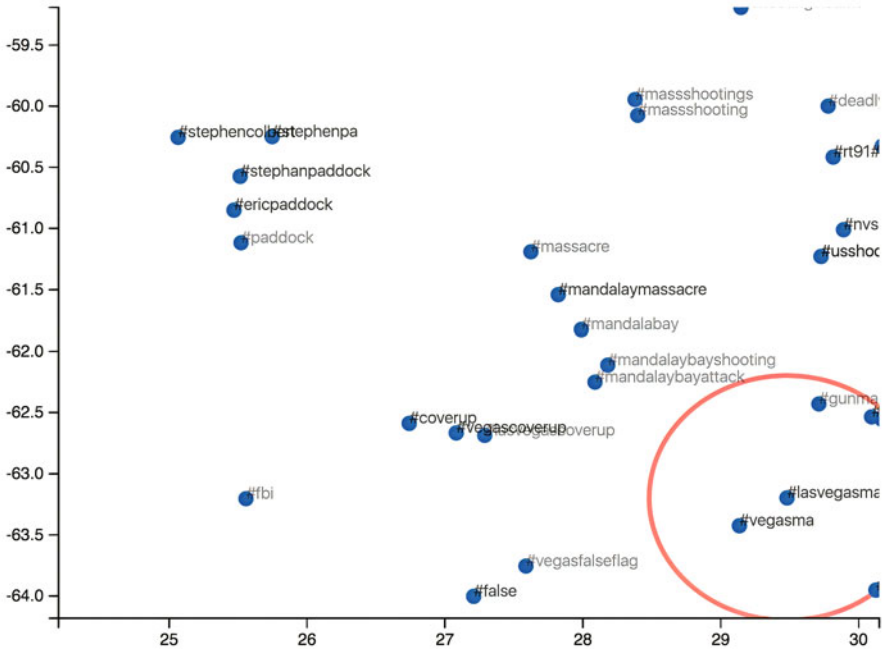


Fig. 2 Embedding of hashtags and visualization in 2D space using the t-SNE algorithm. The dimensions do not have any intrinsic meaning and are only useful as a coordinate system for visualization purposes

original vector space tend to be close in the reduced space also. This means that the visualization more accurately reflects what is happening in the higher dimensional space. In the context of this chapter, using t-SNE allows SAVIZ to present more compact and accurate visualizations (subsequently described) to a user.

SAVIZ uses all the information sets mentioned above, including the categorical labels output by systems like ELISA and the 2D points output by t-SNE, to compile a NoSQL file that is input to the SAVIZ visualization module. This module is based on Bokeh,[6] an interactive, well-documented visualization library that targets modern web browsers for presentation. Bokeh aims to provide elegant, concise construction of versatile graphics, and to extend this capability with high-performance interactivity over very large or streaming datasets. Because it uses Bokeh, SAVIZ requires no extensive set up, since the visualization itself is rendered on a web browser, making the system portable.

[6] https://bokeh.pydata.org/en/latest/docs/dev_guide.html

3.1 User Experience

We demonstrate the simplicity of using, and the key features of, the system for a complex disaster (the Ebola crisis) that continues to unfold in Africa. Figure 3 illustrates the SAVIZ interface for an Ebola dataset that was collected from Twitter. The full corpus that we collected, using Ebola-specific keywords, comprises 18,224 tweets, with timestamps ranging from 2014-Aug-01 00:03 to 2014-Sep-24 23:16. ELISA [4] was executed on this corpus, yielding zero or more situation labels per tweet from a vocabulary of eleven types: *food, infrastructure, water, utilities, regime-change, terrorism, medicine, evacuation, shelter, search*, and *crime/violence*. These labels could be noisy; no ground truth was available against which the accuracy of ELISA on the situation labeling task could be ascertained.[7] For visualization purposes (Fig. 3), we sampled 720 points from this corpus with timestamps ranging from 2014-Aug-01 00:03 to 2014-Sep-24 23:16, and over five common types (infrastructure, water, search, medicine, and food).

To validate user experience, we recently demonstrated SAVIZ at the ACM/IE-EE ASONAM conference in 2019 and allowed the user to play with the Ebola dataset and interface, including facet selection and de-selection, and interaction with points (including drawing of bounding boxes around points). Furthermore, as evidence that the system works for arbitrary disasters, we also considered a second disaster, namely the devastating Gorkha earthquake in Nepal in 2015. This corpus was also collected over Twitter and consists of 29,946 points, with timestamps ranging from 2015-Apr-25 01:00 to 2015-May-06 09:42. Once again, ELISA was executed over this corpus to yield zero or more situation labels from a vocabulary of seven[8] labels (utilities, water, food, medicine, shelter, search, and infrastructure). We present a visualization of the system for the Nepal disaster in Fig. 4. For visualization purposes, we considered a sample of 1,810 tweets, with timestamps ranging from 2015-Apr-25 01:00 to 2015-May-02 06:56, and over five common types (food, medicine, shelter, infrastructure and search).

Availability and Future Development. More recently, we have made SAVIZ available as both a Docker container[9] and a GitHub project to enable open download and experimentation. These projects are respectively available at the following links.[10]

Finally, SAVIZ continues to undergo development and feature additions that will allow users to gain more situational insights in crisis data. An important recent addition has been the ability for users to facet and filter based on *time*. Consider, for example, Fig. 5, which now includes a slider for time to enable a user to limit

[7] The LORELEI program conducts regular evaluations to compute such numbers on specific datasets gathered and curated by the Linguistic Data Consortium.

[8] Slightly different versions of ELISA were available at the times of data collection; hence the vocabularies are different between Nepal and Ebola.

[9] To access and download the Docker package, users may have to first create a free Docker account.

[10] https://hub.docker.com/repository/docker/ppplinday/situation-awareness-visualization?ref=login; https://github.com/ppplinday/Situation-Awareness-Visualization

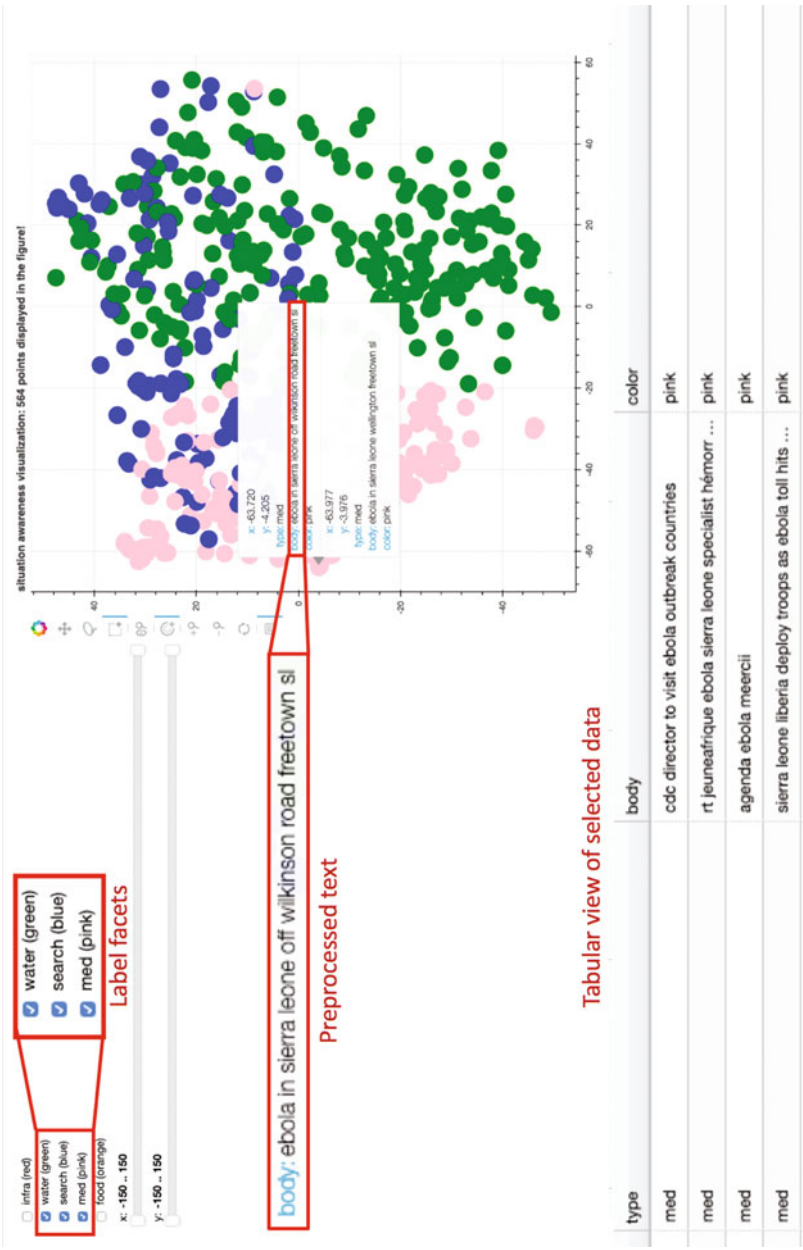


Fig. 3 The SAVIZ interface over an Ebola dataset collected over Twitter. The tabular view can be re-generated, and made more focused, by drawing a bounding box around any portion of the screen

type	time	body	color
search	2015-4-30 9:22	breaking boy rescued in kathmandu rubbl…	orange
search	2015-4-30 10:45	boy rescued in kathmandu rubble many vi…	orange
search	2015-4-30 9:15	boy rescued in kathmandu rubble many…	orange
search	2015-4-30 12:3	news update boy rescued in kathmandu r…	orange
search	2015-4-30 11:33	boy rescued in kathmandu rubble many vi…	orange
search	2015-4-30 9:10	boy rescued in kathmandu rubble many vi…	orange

Fig. 4 The SAVIZ interface over the Nepal earthquake dataset collected over Twitter

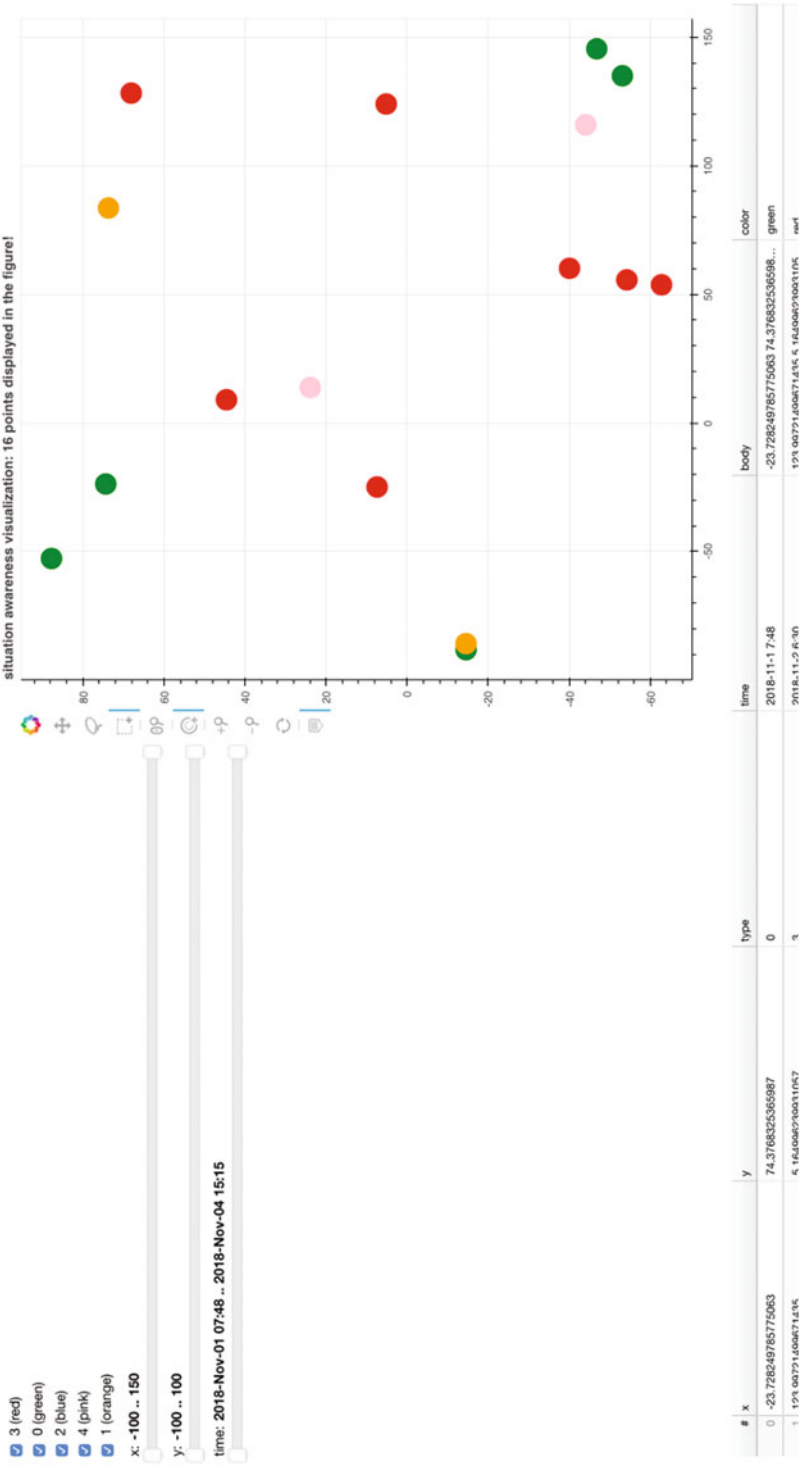


Fig. 5 SAVIZ with a time-filtering slider (below the *x*-*y* sliders) that allows the user more control over the visualization

what they see on the screen to a particular time frame. This facility is expected to be particularly useful for large datasets that have hundreds of thousands of points over a long (or dense) time period.

4 Conclusion

Despite many advances in the Artificial Intelligence and NLP communities, outputs from NLP and text classification algorithms still tend to be imperfect. In humanitarian domains, such as crisis response, first responders and other stakeholders who have to make decisions in the aftermath of crises, are not likely to trust such systems blindly. In this chapter, we presented a highly lightweight, interactive visualization platform called SAVIZ that can be deployed on a web browser in less than 30 s for thousands of tweets, and is designed for short, crisis-specific messages collected over social media like Twitter, and processed by NLP systems like ELISA. SAVIZ relies on established, pre-existing and open-source technologies from the representation learning, visualization and data processing communities. SAVIZ is backward-compatible with crisis informatics sub-systems recently released under the DARPA LORELEI program, and has been applied on real-world datasets collected from the Twitter API.

SAVIZ is intended to provide non-technical first responders with interactive situational awareness capabilities in support of crisis informatics. In the future, we are looking to extend its capabilities to help users correct existing, and provide new, annotations directly using the interface. Currently, data annotation services are expensive and require sharing of data, in addition to not being real time. SAVIZ places control firmly in the hands of the humanitarian and field users, who are best equipped to be both labeling and exploring the data. This is especially true for global crises such as COVID-19, which affect different communities and countries in different ways, and during which local decision-makers have a key role to play in mitigating the damaging impacts of the crisis.

Acknowledgments The authors were supported under the DARPA LORE-LEI program.

References

1. Abel, F., Hauff, C., Houben, G. J., Stronkman, R., & Tao, K. (2012). Twitcident: fighting fire with information from social web streams. In *Proceedings of the 21st International Conference on World Wide Web* (pp. 305–308). ACM.
2. Aibaidulla, Y., & Lua, K. T. (2003). The development of tagged uyghur corpus. In *Proceedings of PACLIC17* (pp. 1–3).
3. Atefeh, F., & Khreich, W. (2015). A survey of techniques for event detection in Twitter. *Computational Intelligence, 31*(1), 132–164.
4. Cheung, L., Gowda, T., Hermjakob, U., Liu, N., May, J., Mayn, A., Pourdamghani, N., Pust, M., Knight, K., Malandrakis, N., et al. Elisa system description for lorehlt (2017).

5. Choi, S., & Bae, B. (2015). The real-time monitoring system of social big data for disaster management. In *Computer science and its applications* (pp. 809–815). Springer.
6. Chong, W. H., Lim, E. P., & Cohen, W. (2017). Collective entity linking in tweets over space and time. In *European Conference on Information Retrieval* (pp. 82–94). Springer.
7. Collobert, R., Weston, J., Bottou, L., Karlen, M., Kavukcuoglu, K., & Kuksa, P. (2011). Natural language processing (almost) from scratch. *Journal of Machine Learning Research, 12*, 2493–2537.
8. Dos Santos, C. N., & Gatti, M. (2014). Deep convolutional neural networks for sentiment analysis of short texts. In *Proceedings of International Conference on Computational Linguistics (COLING)*. Dublin, Ireland.
9. Ford, J. D., Tilleard, S. E., Berrang-Ford, L., Araos, M., Biesbroek, R., Lesnikowski, A. C., MacDonald, G. K., Hsu, A., Chen, C., & Bizikova, L. (2016). Opinion: Big data has big potential for applications to climate change adaptation. *Proceedings of the National Academy of Sciences, 113*(39), 10729–10732.
10. Gao, H., Barbier, G., & Goolsby, R. (2011). Harnessing the crowdsourcing power of social media for disaster relief. *IEEE Intelligent Systems, 26*(3), 10–14.
11. Ghaeini, R., Fern, X. Z., Huang, L., & Tadepalli, P. (2016). Event nugget detection with forward-backward recurrent neural networks. In *The 54th Annual Meeting of the Association for Computational Linguistics* (p. 369).
12. He, X., Lu, D., Margolin, D., Wang, M., Idrissi, S. E., & Lin, Y. R. (2017). The signals and noise: Actionable information in improvised social media channels during a disaster. In *Proceedings of the 2017 ACM on Web Science Conference* (pp. 33–42). ACM.
13. Hong, L., Ahmed, A., Gurumurthy, S., Smola, A. J., & Tsioutsiouliklis, K. (2012). Discovering geographical topics in the twitter stream. In *Proceedings of the 21st International Conference on World Wide Web* (pp. 769–778). ACM.
14. Imran, M., Castillo, C., Lucas, J., Meier, P., & Vieweg, S. (2014). Aidr: Artificial intelligence for disaster response. In *Proceedings of the 23rd International Conference on World Wide Web* (pp. 159–162). ACM.
15. Imran, M., Elbassuoni, S., Castillo, C., Diaz, F., & Meier, P. (2013). Extracting information nuggets from disaster-related messages in social media. In *ISCRAM*.
16. Irvine, A., & Klementiev, A. (2010). Using mechanical turk to annotate lexicons for less commonly used languages. In *Proceedings of the NAACL HLT 2010 Workshop on Creating Speech and Language Data with Amazon's Mechanical Turk* (pp. 108–113). Association for Computational Linguistics.
17. Jadhav, A. S., Purohit, H., Kapanipathi, P., Anantharam, P., Ranabahu, A. H., Nguyen, V., Mendes, P. N., Smith, A. G., Cooney, M., & Sheth, A. P. (2010). Twitris 2.0: Semantically empowered system for understanding perceptions from social data.
18. Joulin, A., Grave, E., Bojanowski, P., & Mikolov, T. (2016). Bag of tricks for efficient text classification. arXiv preprint arXiv:1607.01759.
19. Kaufhold, M. A., Rupp, N., Reuter, C., & Habdank, M. (2019). Mitigating information overload in social media during conflicts and crises: design and evaluation of a cross-platform alerting system. *Behaviour & Information Technology, 39*(3) 1–24.
20. Kejriwal, M., Gilley, D., Szekely, P., & Crisman, J. (2018). Thor: Text-enabled analytics for humanitarian operations. In *Companion of the The Web Conference 2018 on The Web Conference 2018* (pp. 147–150). International World Wide Web Conferences Steering Committee.
21. Kejriwal, M., & Gu, Y. (2018). A pipeline for post-crisis twitter data acquisition. arXiv preprint arXiv:1801.05881.
22. Kejriwal, M., & Zhou, P. (2019). Low-supervision urgency detection and transfer in short crisis messages. arXiv preprint arXiv:1907.06745.
23. Kejriwal, M., & Zhou, P. (2019). Saviz: Interactive exploration and visualization of situation labeling classifiers over crisis social media data.
24. Kumar, S., Barbier, G., Abbasi, M. A., & Liu, H.: Tweettracker: An analysis tool for humanitarian and disaster relief. In *ICWSM* (2011).

25. Maaten, L. V. D., & Hinton, G. (2008). Visualizing data using t-sne. *Journal of Machine Learning Research 9*, 2579–2605.
26. Meier, P. (2015). *Digital humanitarians: how big data is changing the face of humanitarian response*. Taylor & Francis Press.
27. Moro, A., Raganato, A., & Navigli, R.: Entity linking meets word sense disambiguation: a unified approach. *Transactions of the Association for Computational Linguistics, 2*, 231–244 (2014).
28. Morss, R. E., Demuth, J. L., Lazrus, H., Palen, L., Barton, C. M., Davis, C. A., Snyder, C., Wilhelmi, O. V., Anderson, K. M., Ahijevych, D. A., et al. (2017). Hazardous weather prediction and communication in the modern information environment. *Bulletin of the American Meteorological Society, 98*(12), 2653–2674.
29. Nazer, T. H., Xue, G., Ji, Y., & Liu, H. (2017). Intelligent disaster response via social media analysis a survey. *ACM SIGKDD Explorations Newsletter, 19*(1), 46–59.
30. Nguyen, D. T., Joty, S., Imran, M., Sajjad, H., & Mitra, P. (2016). Applications of online deep learning for crisis response using social media information. arXiv preprint arXiv:1610.01030.
31. Olteanu, A., Castillo, C., Diaz, F., & Vieweg, S. (2014). CrisisLex: A lexicon for collecting and filtering microblogged communications in crises. In *Proceedings of the International Conference on Weblogs and Social Media (ICWSM)*. Oxford.
32. Palen, L., & Anderson, K. M. (2016). Crisis informaticsnew data for extraordinary times. *Science, 353*(6296), 224–225.
33. Pang, B., Lee, L., et al. (2008). Opinion mining and sentiment analysis. *Foundations and Trends® in Information Retrieval, 2*(1–2), 1–135.
34. Reuter, C., & Kaufhold, M. A. (2018). Fifteen years of social media in emergencies: a retrospective review and future directions for crisis informatics. *Journal of Contingencies and Crisis Management, 26*(1), 41–57.
35. Rogstadius, J., Vukovic, M., Teixeira, C., Kostakos, V., Karapanos, E., & Laredo, J. A. (2013). Crisistracker: Crowdsourced social media curation for disaster awareness. *IBM Journal of Research and Development, 57*(5), 4–1.
36. Schulz, A., Ristoski, P., & Paulheim, H. (2013). I see a car crash: Real-time detection of small scale incidents in microblogs. In *Extended Semantic Web Conference* (pp. 22–33). Springer.
37. Simon, T., Goldberg, A., & Adini, B. (2015). Socializing in emergenciesa review of the use of social media in emergency situations. *International Journal of Information Management, 35*(5), 609–619.
38. Stowe, K., Palmer, M., Anderson, J., Kogan, M., Palen, L., Anderson, K. M., Morss, R., Demuth, J., & Lazrus, H. (2018). Developing and evaluating annotation procedures for twitter data during hazard events. In *Proceedings of the Joint Workshop on Linguistic Annotation, Multiword Expressions and Constructions (LAW-MWE-CxG-2018)* (pp. 133–143).
39. Tang, J., Qu, M., Wang, M., Zhang, M., Yan, J., & Mei, Q. (2015). Line: Large-scale information network embedding. In *Proceedings of the 24th International Conference on World Wide Web* (pp. 1067–1077). International World Wide Web Conferences Steering Committee.
40. Thom, D., Krüger, R., Ertl, T., Bechstedt, U., Platz, A., Zisgen, J., & Volland, B. (2015). Can twitter really save your life? A case study of visual social media analytics for situation awareness. In *Visualization Symposium (PacificVis), 2015 IEEE Pacific* (pp. 183–190). IEEE.
41. Tierney, T. F. (2014). Crowdsourcing disaster response: Mobilizing social media for urban resilience. *The European Business Review, 80*(9), 1854–1867.
42. Wang, D., Cui, P., & Zhu, W. (2016). Structural deep network embedding. In *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining* (pp. 1225–1234). ACM.
43. Wang, Q., Mao, Z., Wang, B., & Guo, L. (2017). Knowledge graph embedding: A survey of approaches and applications. *IEEE Transactions on Knowledge and Data Engineering, 29*(12), 2724–2743.
44. Zhang, S., & Vucetic, S. (2016). Semi-supervised discovery of informative tweets during the emerging disasters. arXiv preprint arXiv:1610.03750.

Efficient and Flexible Compression of Very Sparse Networks of Big Data

Carson K. Leung, Fan Jiang, and Yibin Zhang

Abstract In the current era of big data, huge amounts of valuable data and information have been generated and collected at a very rapid rate from a wide variety of rich data sources. Social networks are examples of these rich data sources. Embedded in these big data are implicit, previously unknown and useful knowledge that can be mined and discovered by data science techniques such as data mining and social network analysis. Hence, these techniques have drawn attention of researchers. In general, a social network consists of many users (or social entities), who are often connected by "following" relationships. Finding those famous users who are frequently followed by a large number of common followers can be useful. These frequently followed groups of famous users can be of interest to many researchers (or businesses) due to their influential roles in the social networks. However, it can be challenging to find these frequently followed groups because most users are likely to follow only a small number of famous users. In this chapter, we present an efficient and flexible compression model for supporting the analysis and mining of very sparse networks of big data, from which the frequently followed groups of users can be discovered.

Keywords Big data · Social media analytics · Followship · Data compression · Data mining · Frequent patterns

C. K. Leung (✉)
Department of Computer Science, University of Manitoba, Winnipeg, MB, Canada
e-mail: kleung@cs.umanitoba.ca

F. Jiang
Department of Computer Science, University of Northern British Columbia (UNBC), Prince George, BC, Canada

Y. Zhang
Department of Computer Science, University of Manitoba, Winnipeg, MB, Canada

Department of Computer Science, University of Toronto, Toronto, ON, Canada

© The Author(s), under exclusive license to Springer Nature Switzerland AG 2021

M. Çakırtaş, M. K. Ozdemir (eds.), *Big Data and Social Media Analytics*, Lecture Notes in Social Networks, https://doi.org/10.1007/978-3-030-67044-3_9

1 Introduction

Nowadays, big data are everywhere [1]. These big data can be characterized by the well-known 3 V's, 4 V's, 5 V's, etc. (e.g., value, velocity, veracity [2–5], visualization [6, 7], volume). It is because huge volumes of valuable data have been generated and collected at a very high velocity from a wide variety of rich data sources for various real-life applications and services. Examples of these rich data sources include financial services [8], healthcare sectors [9–14], public services [15, 16], and social networks [17–26]. Due to their popularity, social networks can be considered as indispensable products in people's life. In these social networks, people express their personal opinions, browse interesting contents, and follow other favorite individuals (or organizations) on social networks. Consequently, plenty of valuable information and useful knowledge is embedded in social networks. Many researchers, who care about public behavior and psychology (e.g., business, economic, social science), may be interested in mining and analyzing social networks [27, 28] through data science [8, 29, 30], data mining [31–33] and/or machine learning [34, 35] techniques.

In general, *social networks* consist of users or social entities (e.g., individuals, corporations, organizations). Due to the popularity of social networking, the number of users in social networking sites keeps growing. As of July 2020[1], there were:

- 2.60 billion monthly active users (MAU) in Facebook,
- 2.00 billion MAU in WhatsApp and YouTube,
- 1.30 billion MAU in Facebook Messenger,
- 1.20 billion MAU in WeChat,
- 1.08 billion MAU in Instagram (aka Insta or IG),
- 800 million MAU in TikTok,
- 694 million MAU in Tencent QQ (aka QQ),
- 675 million LinkedIn members[2],
- 517–550 million MAU in Sina Weibo and Tencent Qzone,
- 430 million MAU in Reddit,
- 400 million MAU in Telegram[3],
- 397 million MAU in Snapchat,
- 367 million MAU in Pinterest, and
- 326 million MAU in Twitter.

The users use these, as well as other similar, social networking sites for a wide variety of purposes. For instance:

[1] https://www.statista.com/statistics/272014/global-social-networks-ranked-by-number-of-users/

[2] https://www.linkedin.com/company/linkedin/about/

[3] https://www.statista.com/statistics/234038/telegram-messenger-mau-users/

- Discord, Facebook Messenger, iMessage, KakaoTalk, Line, Skype, Snapchat, Telegram, Tencent QQ, Viber, WeChat, and WhatsApp are mostly used for instant messaging;
- Facebook, LinkedIn, Sina Weibo, Tencent Qzone, Tencent Weibo, Twitter, and Tumblr mostly used for micro-blogging;
- Flickr, Google Photos, Instagram, and Pinterest are mostly used for image sharing; whereas
- TikTok and YouTube are mostly used for video sharing.

Regardless of the purposes why the users use these social networking sites, a common observation on these social network users is that they are linked by some relationships (e.g., friendship, common interest) on the social networks. Examples of these linkages include:

- mutual friendships, and
- "following" patterns or relationships.

For example, in Facebook (as one of the aforementioned social networking sites), users can create a personal profile and add other Facebook users as friends. To elaborate, a user X can add another user Y as a friend by sending Y a friend request. Upon Y's acceptance of X's friend request, X and Y can become mutual friends. In addition to exchanging messages among *mutual friends*, Facebook users can also join common-interest user groups and categorize their friends into different customized lists (e.g., classmates, co-workers). The number of (mutual) friends may vary from one Facebook user to another. It is not uncommon for a user A to have hundreds or thousands of friends.

Besides mutual friendships, another common linkage between users in social networks is a *"following" pattern* or *"following" relationship*, which captures the linkage that a social network user X follows another user Y. Let us elaborate by continuing with the aforementioned example on Facebook users. Although many of the Facebook users are linked to some other Facebook users via the mutual friendship (i.e., if a user X is a friend of another user Y, then user Y is also a friend of user X), there are also situations in which such a relationship is no longer mutual. To handle these situations, Facebook added the functionality of 'subscribe' in 2011, which was relabelled as 'follow' in 2012. Specifically, a user can subscribe or follow public postings of some other Facebook users—usually, famous celebrities, public institutions, product and services, news media, and well-known bloggers—without the need of adding them as friends. A user X may follow other users who do not know user X. In this situation, the link between these social entities is no longer mutual but a directional "following" pattern.

With these "following" patterns, many decisions and recommendations can be made. Consider the following scenario. When many friends of a Facebook user X follow some individual users or groups of famous users, it is likely that user X may also be interested in following these individual users or groups of famous users. This

leads to a collection of most-followed pages[4], which include Facebook accounts of some sports players (e.g., soccer player Ronaldo), popular performers (e.g., musicians, actors and actresses), public figures, and politicians. One can discover groups of famous users (or social entities), which are followed by a significant number of common users. We call these groups of famous users the *frequently followed groups*. Upon the discovery of these frequently followed groups, if any social network user X follows some members of these groups, we could recommend other members of these groups to user X.

To find these frequently followed groups, we need an efficient and flexible way to represent the big social network. However, members in these frequently followed groups are only some tiny percentages of users in the entire social network. Hence, a major challenge is how to capture and represent the "following" relationships and mine frequently followed groups from the captured and represented relationships. Consequently, some logical questions include:

- How to capture and represent "following" relationships among users in a big social network?
- How to efficiently and flexibly compress such representation of "following" relationships among users who are sparsely distributed in a big social network?
- How to effectively mine and analyze the compressed representation to discover those frequently followed groups of users?

To answer these questions, we presented an efficient and flexible compression model for supporting the analysis and mining of very sparse networks of big data, from which the frequently followed groups of users can be discovered. As the current chapter is an extension of our paper titled "flexible compression of big data" [36] published in the *Proceedings of 2019 IEEE/ACM Conference on Advances in Social Networks Analysis and Mining* (*ASONAM'19*), our *key contributions* of the current chapter include:

- multi-line position list word-aligned hybrid compressed bitwise representations—called MPLWAH(k)—of a very sparse but big social network embedded with "following" patterns; and
- an efficient and flexible data science solution for supporting big data mining and analysis on the discovery of frequent "following" groups from the compressed data structure.

The remainder of the current chapter is organized as follows. We first provide some background and discuss some related works. Then, we describe our efficient and flexible compression model for supporting the analysis and mining of very sparse networks of big data, from which the frequently followed groups of users can be discovered. Afterwards, we show our evaluation results on three real-life datasets. Finally, we draw our conclusions and share some ideas for future work.

[4] https://socialblade.com/facebook/top/50/likes

2 Background and Related Work

Compression techniques have been applied to various areas, include compression of data [37, 38], image and video compression [39–42], as well as sequence compressions (e.g., DNA sequences) [43, 44]. Compressed data in these application areas help speed up the information retrieval of data in the areas. However, as they were not designed for social network analysis and mining, most of them cannot be easily adapted to compressing social networking sites.

For related works that focus on compressing networks or graphs [45, 46], most of them aim to compressed the networks for community detection. As such, the compressed networks may not be easily analyzed or mined for frequent patterns or "following" patterns (e.g., frequent "following" groups of famous users).

Regarding related works that focus on compressing social networks for frequent pattern mining and analysis, we [47] presented a social network mining strategy in the IEEE/ACM ASONAM 2016. The strategy applies the word-aligned hybrid (WAH) compression model to reduce the sparsity of "following" data. The idea behind this compression model is to divide the long bitmap into groups of 31 bits, then encode long-run consecutive zero groups (a group without 1 bit) into a compressed word. If a "1"-bit appears in a group, then the group is stored without compression.

In the IEEE/ACM ASONAM 2017, we [48] presented an improved social network "following" pattern solution, which applies an improved position list word-aligned hybrid (IPLWAH) compression model. This compression model encodes both

- long run consecutive zero bits, as well as
- their succeeding group if there are at most k single-bits in the succeeding group.

The solution further reduces the sparsity.

As a logical but non-trivial extension of these works, our MPLWAH(k) compression solution for supporting social network analysis and mining—presented in the current chapter—is based on the above two strategies. We encode long run consecutive zero groups and their succeeding four groups if there are at most four "1"-bits in them.

3 Our Efficient and Flexible Compression Model

To illustrate the key idea behind our efficient and flexible compression model, let us consider a sample social network with many users. Among them, the following six followees (i.e., famous users who are frequently followed other users) are of interest:

- Alice (user #84650),
- Bob (user #169237),

- Carol (user #169248),
- Don (user #253661),
- Eva (user #253667), and
- Fred (user #253673).

Let us consider the following seven followers who follow these six famous users in the social network:

- Gigi (user #7) follows Bob and Eva;
- Henry (user #12308) follows Alice, Carol and Eva;
- Iris (user #90009) follows Alice and Eva;
- John (user #101010) follows Bob, Carol and Eva;
- Kat (user #7600011) follows Alice, Bob, Carol, Don and Fred;
- Leo (user #112012012) follows Alice only; and
- Monica (user #123456789) follows Alice, Bob, Carol, Don, Eva and Fred.

Here, the relationship between Gigi and Bob is unidirectional such that Gigi follows Bob but Bob does not follow Gigi. If Alice happens to follow Leo, then the relationship between Alice and Leo would be (bidirectional) mutual friends such that they would follow each other.

3.1 Graph Representation of a Social Network

A logical representation of a social network is a *graph* $G_1 = (V, E)$ with a set V of vertices and a set E of edges. Each vertex represents an individual user (i.e., a social entity) in the social network, and each edge represents a "following" relationship between a pair of vertices. In theory, there can be at most $(|V|^2 - |V|)$ directional edges with |V| vertices in a social network graph. So, with millions of MAU in many real-life social networking sites, there can be trillions of directional edges. For instance, if $|V| \sim 10^6$, then $|E| \sim 10^{12}$. In reality, most social network users would not follow millions of other users. Hence, the number of edges is usually much smaller.

Reconsider the aforementioned sample social networks. The corresponding graph G_1 contains of millions of vertices representing user #1 to user #123456789 (Monica), including at least $6 + 7 = 13$ vertices representing the six followees and seven followers (i.e., Alice, Bob, ..., Leo, and Monica). The graph G_1 also contains $|E| = 22$ directional edges representing the 22 "following" relationships (e.g., Gigi is following Bob).

Another alternative representation for capturing the "following" relationships would be a *bipartite graph* $G_2 = (U, V, E)$. With it, U represents a set of followers, V represents a set of followees, and E represents a set of "following" relationships. Each edge $e \in E$ captures the information that a follower $u \in U$ follows a followee $v \in V$. The maximum number of edges is bounded above by $|U| \times |V|$.

3.2 Matrix Representation of a Social Network

A third logical representation of a social network is an *adjacency matrix*. When putting followers in the row and followees in the column, each entry in (row r, column c) represents a "following" relationship that r follows c. More precisely, the social network is represented by a Boolean matrix such as a "1" entry in in (row r, column c) represents the presence of a "following" relationship that r follows c. A "0" entry in (row r, column c) represents the absence of any "following" relationship between r and c, meaning that r does not follow c. This Boolean matrix is of size $|V| \times |V|$, where V is the set of users (i.e., social entities) in the social network.

Reconsider the aforementioned sample social networks. With millions of users (e.g., user #1 to at least user #123456789 named Monica), the adjacency matrix can be large. When $|V| \sim 10^6$, $|E| \sim 10^{12}$. It is relieved that, in terms of memory consumption, this matrix can be stored as a Boolean matrix, i.e., a bit matrix or a collection of bit vectors. Usually, the adjacency matrix representing the social networks would be sparse. For instance, reconsider the aforementioned sample social networks. Among all $|V| \times |V|$ bits in the matrix, only 22 of them are "1"-bits representing the 22 "following" relationships in the network.

3.3 Bit Vector Representation of a Follower in a Social Network

A fourth logical representation of a social network is a *collection of bit vectors*. As observed from the previous section, a social network can be represented as a Boolean matrix (or bit matrix), which in turn can be considered as a collection of rows. Each row can then be represented as a bit vector capturing the followees followed by the user for that row. With this representation, each vector is of length $|V|$. Given that the Boolean matrix representing the social network is usually sparse, the corresponding bit vector for each row of the matrix is also expected to be sparse (i.e., sparse bit vector). In terms of memory consumption, the "following" relationships can be captured by $|U|$ bit vectors, each of which of length $|V|$. Here, U represents a set of followers, and V represents a set of users to be followed by followers.

Reconsider the aforementioned sample social networks. The 22 "following" relationships can be captured by 6 bit vectors representing 6 followers (Alice, Bob, . . . , Fred). With millions of users (e.g., user #1 to at least user #123456789 named Monica), each bit victor can be long (e.g., $|V| \sim 10^6$ bits). It is relieved that, in terms of memory consumption, the total number "1"-bits among the 6 bit vectors is only 22, which represents the 22 "following" relationships in the social network. For instance, the bit vector for Monica is of length at least 123,456,789 bits. Among these bits, there are only six "1"-bits, which are located at the 84,650th, 169,237th, 169,248th, 253,661st, 253,667th and 253,673rd positions representing the 84,650th, 169,237th, 169,248th, 253,661st, 253,667th and 253,673rd users—namely, Alice,

Bob, Carol, Don, Eva and Fred. The remaining bits (at least 123,456,783 bits) on this bit vector for Monica are "0"-bits. This bit vector can be stored in 123,456,789 $\div$ 32 $\approx$ 3,858,025 words (32-bit words). Similar memory requirements for the other five followers, for a total of 6 $\times$ 3,858,025 words = 23,148,150 words (32-bit words).

3.4 Word-Aligned Hybrid (WAH) Compressed Bitmap Representation of a Follower in a Social Network

The (uncompressed) bit vector representing followers in a social network is observed to be usually sparse. In order to alleviate the data sparsity, the bit vector can be compressed by the word-aligned hybrid (WAH) compression so as to save memory usage, and thus enhance mining performance. With this compression model, the bit vector can be divided into groups of 31 bits. Depending on the contents (i.e., composition of "0"- and "1"-bits), these groups can be classified into literal and zero-fill ones. Groups of consecutive zero-fill 31 bits can form some zero-fill words, whereas remaining groups of 31 bits with a mixture of "0"- and "1"-bits form some literal words. More precisely,

- A literal word is represented as a 32-bit word, in which (i) the 1st bit is "0" indicating that it is a literal word and (ii) the remaining 31 bits contain a mixture of "0"- and "1"-bits.
- A zero-fill word is also represented as a 32-bit word, in which (i) the 1st bit is "1" indicating that it is a fill word, (ii) the 2nd bit is "0" indicating that the fill word is a zero-fill word, and (iii) the remaining 30 bits indicate the number of consecutive groups of 31 zeros.

The resulting WAH compressed bitmap consists of a sequence of these literal words and zero-fill words.

3.4.1 An Example of WAH Compressed Bitmap

Reconsider the (uncompressed) bit vector for Monica, in there are six "1"-bits located at the 84,650th, 169,237th, 169,248th, 253,661st, 253,667th and 253,673rd positions representing the 84,650th, 169,237th, 169,248th, 253,661st, 253,667th and 253,673rd users—namely, Alice, Bob, Carol, Don, Eva and Fred. The remaining bits are all "0"-bits. By dividing this bit vector into groups of 31 bits, we observed that there are:

- 2730 groups of consecutive zero-fill 31 bits; succeeded by
- a mixture of "0"- and "1"-bits in the next group of 31 bits (or precisely, a "1"-bit in the 20th position and "0" for the remaining 30 bits);
- 2728 groups of consecutive zero-fill 31 bits;

- another mixture of "0"- and "1"-bits in the next group of 31 bits (or precisely, a "1"-bits in the 8th and 19th positions and "0" for the remaining 29 bits);
- 2722 groups of consecutive zero-fill 31 bits;
- a third mixture of "0"- and "1"-bits in the next group of 31 bits (or precisely, a "1"-bit in the 19th position and "0" for the remaining 30 bits); succeeded by
- a fourth mixture of "0"- and "1"-bits in the next group of 31 bits (or precisely, a "1"-bit in the 25th position and "0" for the remaining 30 bits); and finally succeeded by
- a fifth mixture of "0"- and "1"-bits in the next group of 31 bits (or precisely, a "1"-bit in the 31st position and "0" for the remaining 30 bits).

With the observations, this uncompressed bit vector for Monica can be compressed via the WAH compression into a compressed bitmap consisting of a sequence of 8 literal and zero-fill 32-bit words:

- The first 32-bit word in the sequence is a zero-fill word 10 00 0000 0000 0000 0000 1010 1010 1010, where $1010\ 1010\ 1010_{(2)} = 2730_{(10)}$ indicates 2730 groups of consecutive zero-fill 31 bits.
- The second 32-bit word in the sequence is a literal word 0 00000 00000 00000 00001 00000 00000 0, where (i) the prefix "0" indicates that it is a literal word and (ii) the "1"-bit is in the 20th position within the suffix 31-bits.
- The third 32-bit word in the sequence is a zero-fill word 10 00 0000 0000 0000 0000 1010 1010 1000, for $1010\ 1010\ 1000_{(2)} = 2728_{(10)}$ groups of consecutive zero-fill 31 bits.
- The fourth 32-bit word in the sequence is a literal word 0 00000 00100 00000 00010 00000 00000 0, where the two "1"-bits are in the 8th and 19th positions within the suffix 31-bits.
- The fifth 32-bit word in the sequence is a zero-fill word 10 00 0000 0000 0000 0000 1010 1010 0010, for $1010\ 1010\ 0010_{(2)} = 2722_{(10)}$ groups of consecutive zero-fill 31 bits.
- The sixth 32-bit word in the sequence is a literal word 0 00000 00000 00000 00010 00000 00000 0, where the "1"-bit is in the 19th position within the suffix 31-bits.
- The seventh 32-bit word in the sequence is a literal word 0 00000 00000 00000 00000 00001 00000 0, where the "1"-bit is in the 25th position within the suffix 31-bits.
- The last 32-bit word in the sequence is a literal word 0 00000 00000 00000 00000 00000 00000 1, where the "1"-bit is in the 31st (i.e., last) position within the suffix 31-bits.

This example illustrates that the uncompressed bit vector for Monica (which requires 3,858,025 words) can be compressed into a WAH compressed bitmap consisting of only 8 words.

3.5 Improved Position List Word-Aligned Hybrid (IPLWAH) Compressed Bitmap Representation of a Follower in a Social Network

Compressing a bit vector representing followers in a social network into a WAH bitmap saves memory usage, and thus reduces runtime for social network analysis and social network data mining. Space required can further be reduced by the improved position list word-aligned hybrid (IPLWAH) compression. The key idea is to utilize some portions of the suffix 31-bits within those zero-fill words by encoding a few (i.e., $k \leq 5$) "1"-bits in a literal word succeeding a zero-fill word. There are variants of this compression model, depending on the number k of "1"-bits allowed to be utilized into the zero-fill word. So, with the IPLWAH(k) model, the uncompressed bit vector can be divided into groups of 31 bits, which can be classified into literal and zero-fill ones. Groups of consecutive zero-fill 31 bits can form some zero-fill words, whereas remaining groups of 31 bits with a mixture of "0"- and "1"-bits form some literal words. More precisely,

- A literal word is represented as a 32-bit word, in which (i) the 1st bit is "0" indicating that it is a literal word and (ii) the remaining 31 bits contain a mixture of "0"- and "1"-bits.
- A zero-fill word is also represented as a 32-bit word, in which (i) the 1st bit is "1" indicating that it is a fill word, (ii) the 2nd bit is "0" indicating that the fill word is a zero-fill word, (iii) the next 5 k bits capture the positions of the k "1"-bits in the literal word succeeding the consecutive groups of 31 zeros, and (iv) the last (30 – 5 k) bits indicate the number of these consecutive groups of 31 zeros.

The resulting IPWAH(k) compressed bitmap consists of a sequence of these literal words and zero-fill words.

3.5.1 An Example of IPLWAH(1) Compressed Bitmap

Reconsider the WAH compressed bitmap for Monica. We observed that it is represented as a sequence of 8 words encoding that:

- 2730 groups of consecutive zero-fill 31 bits are succeeded by a literal word with a "1"-bit in the 20th position within the suffix 31-bits; then
- 2728 groups of consecutive zero-fill 31 bits;
- a literal word with two "1"-bits in the 8th and 19th positions;
- 2722 groups of consecutive zero-fill 31 bits are succeeded by a literal word with a "1"-bit in the 19th position within the suffix 31-bits;
- a literal word with a "1"-bit in the 25th position within the suffix 31-bits; and finally succeeded by
- another literal word with a "1"-bit in the 31st (i.e., last) position within the suffix 31-bits.

With the observations, followees of Monica can be compressed via the IPLWAH(1) compression into a compressed bitmap consisting of a sequence of 6 literal and zero-fill 32-bit words:

- The first 32-bit word in the sequence is a zero-fill word 10 10100 0 0000 0000 0000 1010 1010 1010, where (i) prefix "10" indicates that it is a zero-fill word with (ii) 1010 1010 1010 $_{(2)} = 2730_{(10)}$ groups of consecutive zero-fill 31 bits succeeded by (iii) a literal word with a "1" in the bit position $10{,}100_{(2)} = 20_{(10)}$.
- The second and third 32-bit word in the sequence are the same as the third and fourth words in the WAH compressed bitmap for Monica:
  10 00 0000 0000 0000 0000 1010 1010 1000
  0 00000 00100 00000 00010 00000 00000 0
- The fourth 32-bit word in the sequence is a zero-fill word 10 10011 0 0000 0000 0000 1010 1010 0010, where (i) 1010 1010 0010 $_{(2)} = 2722_{(10)}$ groups of consecutive zero-fill 31 bits are succeeded by (ii) a literal word with a "1" in the bit position $10011_{(2)} = 19_{(10)}$.
- The fifth and sixth 32-bit words in the sequence are the same as the seventh and eighth (i.e., last) words in the WAH compressed bitmap for Monica:
  0 00000 00000 00000 00000 00001 00000 0
  0 00000 00000 00000 00000 00000 00000 1

In this example, IPLWAH(1) compressed bitmap combines the first and second words of the corresponding WAH bitmap into a single word as the first word of the IPLWAH(1) bitmap. Similarly, the fifth and sixth words of the corresponding WAH bitmap into another single word as the fourth word of the IPLWAH(1) bitmap. The example illustrates that the uncompressed bit vector for Monica (which requires 3,858,025 words) can be compressed into a WAH compressed bitmap consisting of 8 words, which can further reduced into an IPLWAH(1) compressed bitmap consisting of only 6 words.

3.5.2 An Example of IPLWAH(2) Compressed Bitmap

Note that followees of Monica can also be compressed via the IPLWAH(2) compression into a compressed bitmap consisting of a sequence of 5 literal and zero-fill 32-bit words:

- The first 32-bit word in the sequence is the same as the first word in the IPLWAH(1) compressed bitmap for Monica:
  10 10100 00000 0000 0000 1010 1010 1010
- The second 32-bit word in the sequence becomes a zero-fill word 10 01000 10011 0000 0000 1010 1010 1010, where (i) prefix "10" indicates that it is a zero-fill word with (ii) 1010 1010 1000 $_{(2)} = 2728_{(10)}$ groups of consecutive zero-fill 31 bits succeeded by (iii) a literal word with two "1"s in bit positions $01000_{(2)} = 8_{(10)}$ and $10{,}011_{(2)} = 19_{(10)}$.

- The third, fourth and fifth 32-bit words in the sequence are the same as the fourth, fifth and sixth (i.e., last) words in the IPLWAH(1) compressed bitmap for Monica:
  10 10011 00000 0000 0000 1010 1010 0010
  0 00000 00000 00000 00000 00001 00000 0
  0 00000 00000 00000 00000 00000 00000 1

In this example, IPLWAH(2) compressed bitmap combines the second and third words of the corresponding IPLWAH(1) bitmap into a single word as the second word of the IPLWAH(2) bitmap. The example illustrates that the uncompressed bit vector for Monica (which requires 3,858,025 words) can be compressed into a WAH compressed bitmap consisting of 8 words, which can further reduced into IPLWAH(1) and IPLWAH(2) compressed bitmaps consisting of only 6 words and 5 words, respectively.

3.6 Multi-group Position List Word-Aligned Hybrid (MPLWAH) Compressed Bitmap Representation of a Follower in a Social Network

Compressing a bit vector representing followers in a social network into a WAH or IPLWAH(k) bitmap—for $1 \leq k \leq 5$—saves memory usage, and thus reduces runtime for social network analysis and social network data mining. We observed that IPLWAH(k) utilizes the space of some portions of the suffix 31-bits within those zero-fill words by encoding a few (i.e., $k \leq 5$) "1"-bits in a single literal word succeeding a zero-fill word. A logical question is that what if those few "1"-bits are not in the same single literal word but distributed among consecutive literal words succeeding a zero-fill word?

In response to this question, we present the multi-group position list word-aligned hybrid (MPLWAH) compression, which further reduces the space required by IPLWAH(k). It does so by utilizing some portions of the suffix 31-bits within those zero-fill words to encode a few (i.e., $k \leq 5$) "1"-bits in consecutive literal words succeeding a zero-fill word. Again, there are variants of this compression model, depending on the number k of "1"-bits allowed to be utilized into the zero-fill word. So, with the MPLWAH(k) model, the uncompressed bit vector can be divided into groups of 31 bits, which can be classified into literal and zero-fill ones. Groups of consecutive zero-fill 31 bits can form some zero-fill words, whereas remaining groups of 31 bits with a mixture of "0"- and "1"-bits form some literal words. More precisely,

- A literal word is represented as a 32-bit word, in which (i) the 1st bit is "0" indicating that it is a literal word and (ii) the remaining 31 bits contain a mixture of "0"- and "1"-bits.

- A zero-fill word is also represented as a 32-bit word, in which (i) the 1st bit is "1" indicating that it is a fill word, (ii) the 2nd bit is "0" indicating that the fill word is a zero-fill word, (iii) the next (k–1) bits indicate whether the corresponding "1"-bits are in the succeeding group or not, (iv) the next 5 k bits capture the position of the k "1"-bits in consecutive literal words succeeding the consecutive groups of 31 zeros, and (v) the last (31 – 6 k) bits indicate the number of these consecutive groups of 31 zeros.

The resulting MPWAH(k) compressed bitmap consists of a sequence of these literal words and zero-fill words.

Note that, for a specific variant when $k = 1$, then a zero-fill word for MPWAH(1) is represented as a 32-bit word, in which (i) the first two bits are "10" indicating that it is a zero-fill word, (ii) the next 5 bits capture the position of a "1"-bit in consecutive literal words succeeding the consecutive groups of 31 zeros, and (iii) the last 25 bits indicate the number of these consecutive groups of 31 zeros. Observant readers may notice that this representation of zero-fill word for MPWAH(1) is identical to that for IPWAH(1). However, representations of the zero-fill words for MPWAH(k) and IPWAH(k) are different for other variants when $k > 1$ (but bounded above by 5), i.e., $1 < k \leq 5$. For MPWAH(k), while k is bounded by 5 in theory, but k is bounded above by 4 in practice.

3.6.1 An Example of MPLWAH(2) Compressed Bitmap

Reconsider the IPLWAH(2) compressed bitmap for Monica. We observed that it is represented as a sequence of 5 words encoding that:

- 2730 groups of consecutive zero-fill 31 bits are succeeded by a literal word with a "1"-bit in the 20th position within the suffix 31-bits; then
- 2728 groups of consecutive zero-fill 31 bits are succeeded by a literal word with two "1"-bits in the 8th and 19th positions within the suffix 31-bits;
- 2722 groups of consecutive zero-fill 31 bits are succeeded by a literal word with a "1"-bit in the 19th position within the suffix 31-bits, and another succeeding literal word with a "1"-bit in the 25th position within the suffix 31-bits; and finally succeeded by
- a third literal word with a "1"-bit in the 31st (i.e., last) position within the suffix 31-bits.

With the observations, followees of Monica can be compressed via the MPLWAH(2) compression into a compressed bitmap consisting of a sequence of 4 literal and zero-fill 32-bit words:

- The first 32-bit word in the sequence is a zero-fill word 10 0 10100 0000 0000 0000 1010 1010 1010, where (i) prefix "10" indicates that it is a zero-fill word with (ii) $1010\ 1010\ 1010_{(2)} = 2730_{(10)}$ groups of consecutive zero-fill 31 bits succeeded by (iii) a literal word with a "1" in the bit position $10{,}100_{(2)} = 20_{(10)}$.

- The second 32-bit word in the sequence is a zero-fill word 10 0 01000 10011 000 0000 1010 1010 1000, where (i) prefix "10" indicates that it is a zero-fill word with (ii) 1010 1010 1000 $_{(2)} = 2728_{(10)}$ groups of consecutive zero-fill 31 bits succeeded by (iii) a literal word with a "1" in the bit position 01000 $_{(2)} = 8_{(10)}$ and (iv) another "1" in the bit position 10,011 $_{(2)} = 19_{(10)}$ on (v) the same group/literal word as the one with the "1"-bit in the 8th position (as indicated the "0" in the third bit of the 32-bit word).
- The third 32-bit word in the sequence is a zero-fill word 10 1 10011 11001 000 0000 1010 1010 0010, where (i) prefix "10" indicates that it is a zero-fill word with (ii) 1010 1010 0010 $_{(2)} = 2722_{(10)}$ groups of consecutive zero-fill 31 bits succeeded by (iii) a (first) literal word with a "1" in the bit position 10,011 $_{(2)} = 19_{(10)}$, which in turn is succeeded by (iv) a second literal word (as indicated the "1" in the third bit of the 32-bit word) with (v) a "1" in the bit position 11,001 $_{(2)} = 25_{(10)}$.
- The fourth 32-bit word in the sequence is the same as the fifth (i.e., last) word in the IPLWAH(2) compressed bitmap for Monica:
  0 00000 00000 00000 00000 00000 00000 1

In this example, MPLWAH(2) compressed bitmap combines the third and fourth words of the corresponding IPLWAH(2) bitmap into a single word as the third word of the MPLWAH(1) bitmap. Here, the two single "1"-bits are on two consecutive literal words succeeded by groups of consecutive zero-fill 31 bits. MPLWAH(2) manages to combine and encode them into a single word. The example illustrates that the uncompressed bit vector for Monica (which requires 3,858,025 words) can be compressed into a WAH, IPLWAH(1) and IPLWAH(2) compressed bitmap consisting of 8, 6 and 5 words, which can further reduced into a MPLWAH(2) compressed bitmap consisting of only 4 words.

3.6.2 An Example of MPLWAH(3) Compressed Bitmap

Note that followees of Monica can also be compressed via the MPLWAH(3) compression into a compressed bitmap consisting of a sequence of 3 literal and zero-fill 32-bit words:

- The first 32-bit word in the sequence is similar but not identical to the first word in the MPLWAH(2) compressed bitmap for Monica in the sense that MPLWAH(3) uses two bits (precisely, the third and fourth bits in the 32-bit word) to indicate (i) whether or not the second "1"-bit is on a literal word succeeding the one containing the first "1"-bit and (ii) whether or not the third "1"-bit is on a literal word succeeding the one containing the second "1"-bit. Hence, the resulting 32-bit word in the sequence is a zero-fill word 10 0 0 10100 00000 00000 0 1010 1010 1010, where (i) prefix "10" indicates that it is a zero-fill word with (ii) 1010 1010 1010 $_{(2)} = 2730_{(10)}$ groups of consecutive zero-fill 31 bits succeeded by (iii) a literal word with a "1" in the bit position 10100 $_{(2)} = 20_{(10)}$.

- The second 32-bit word in the sequence is similar but not identical to the first word in the MPLWAH(2) compressed bitmap for the same reason. Hence, the resulting second 32-bit word in the sequence is a zero-fill word 10 0 0 01000 10011 00000 0 1010 1010 1000.
- The third 32-bit word in the sequence is a zero-fill word 10 1 1 10011 11001 11111 0 1010 1010 0010 is formed by combining the third and fourth words of MPLWAH(2) bitmap. Here, (i) prefix "10" indicates that it is a zero-fill word with (ii) 1010 1010 0010 $_{(2)} = 2722\ _{(10)}$ groups of consecutive zero-fill 31 bits succeeded by (iii) a (first) literal word with a "1" in the bit position 10011 $_{(2)} = 19\ _{(10)}$, which in turn is succeeded by (iv) a second literal word (as indicated the "1" in the third bit of the 32-bit word) with (v) a "1" in the bit position 11001 $_{(2)} = 25\ _{(10)}$ such that (vi) this second literal word is then succeeded by a third literal word (as indicated the "1" in the fourth bit of the 32-bit word) with (vii) a "1" in the bit position 11111 $_{(2)} = 31\ _{(10)}$.

In this example, MPLWAH(3) compressed bitmap combines the third and fourth words of the corresponding MPLWAH(2) bitmap into a single word as the third word of the MPLWAH(3) bitmap. The example illustrates that the uncompressed bit vector for Monica (which requires 3,858,025 words) can be compressed into a WAH, IPLWAH(1), IPLWAH(2) and MPLWAH(2) compressed bitmap consisting of 8, 6, 5 and 4 words, which can further reduced into a MPLWAH(3) compressed bitmap consisting of only 3 words.

3.6.3 Other Examples of MPLWAH(3) Compressed Bitmaps

For completeness and for preparation of our social network mining, we compute the MPLWAH(3) compressed bitmaps for capturing followees of other followers in our aforementioned sample social networks:

- MPLWAH(3) compressed bitmap for Monica can be represented as:
  10 0 0 10100 00000 00000 0 1010 1010 1010
  10 0 0 01000 10011 00000 0 1010 1010 1000
  10 1 1 10011 11001 11111 0 1010 1010 0010
- MPLWAH(3) compressed bitmap for Leo can be represented as:
  10 0 0 10100 00000 00000 0 1010 1010 1010
- MPLWAH(3) compressed bitmap for Kat can be represented as:
  10 0 0 10100 00000 00000 0 1010 1010 1010
  10 0 0 01000 10011 00000 0 1010 1010 1000
  10 0 0 10011 00000 00000 0 1010 1010 0010
  0 0 11111 00000 00000 0 0000 0000 0001
- MPLWAH(3) compressed bitmap for John can be represented as:
  10 0 0 01000 10011 00000 1 0101 0101 0011
  10 0 0 11001 00000 00000 0 1010 1010 0011
- MPLWAH(3) compressed bitmap for Iris can be represented as:
  10 0 0 10100 00000 00000 0 1010 1010 1010

10 0 0 11001 00000 00000 1 0101 0100 1100
- MPLWAH(3) compressed bitmap for Henry can be represented as:
  10 0 0 10100 00000 00000 0 1010 1010 1010
  10 0 0 10011 00000 00000 0 1010 1010 1000
  10 0 0 11001 00000 00000 0 1010 1010 0011
- MPLWAH(3) compressed bitmap for Gigi can be represented as:
  10 0 0 01000 00000 00000 1 0101 0101 0011
  10 0 0 11001 00000 00000 0 1010 1010 0011

These MPLWAH(3) compressed bitmaps for the six followers can be denoted in their hexadecimal notation as follows:

- MPLWAH(3) compressed bitmap for Monica can be denoted as:
  0x8A000AAA 844C0AA8 B9E7EAA2
- MPLWAH(3) compressed bitmap for Leo can be denoted as:
  0x8A000AAA
- MPLWAH(3) compressed bitmap for Kat can be denoted as:
  0x8A000AAA 844C0AA8 89800AA2 8F800001
- MPLWAH(3) compressed bitmap for John can be denoted as:
  0x844C1553 8C800AA3
- MPLWAH(3) compressed bitmap for Iris can be denoted as:
  0x8A000AAA 8C80154C
- MPLWAH(3) compressed bitmap for Henry can be denoted as:
  0x8A000AAA 89800AA8 8C800AA3
- MPLWAH(3) compressed bitmap for Gigi can be denoted as:
  0x84001553 8C800AA3

4 Our Data Science Solution for Social Network Mining on MPLWAH Compressed Bitmaps

To find frequently followed groups, we present a data science solution for social network mining on MPLWAH compressed bitmaps. Specifically, our solution algorithm mines a collection of MPLWAH(k) compressed bitmaps, each bitmap represents information—capturing a list of followees—of a follower.

The algorithm recursively mines the frequently followed groups as follows. It first locates the first followee X (in terms of user ID) in each bitmap, and groups the followers following the same first followee X. If the number of followers in a group of the first followee X (e.g., who has the smallest user ID) meets or exceeds a user-specified frequency threshold, then the followee X is considered a *frequently followed followee*. Then, the algorithm focuses on those followers who are following this frequently followed followee X. The algorithm then locates the second followee Y in the bitmap of these focused followers, and groups the followers following the same second followee Y. If the number of followers in a group of the second

followee Y meets or exceeds a user-specified frequency threshold, then the group of followees {X, Y} is considered a *frequently followed group of followees*. This step is repeated recursively to find all the frequently followed groups of followees containing/including the followee X.

After finding all the frequently followed groups of followees containing/including the followee X, the algorithm backtracks to focus on the suffix of each bitmap (i.e., bits representing followees with user IDs bigger than that of X). It locates the first followee Y in each suffix of the bitmap, and groups the followers following the same first followee Y. If the number of followers in a group of the first followee Y (e.g., who has the smallest user ID within the suffix) meets or exceeds a user-specified frequency threshold, then the followee Y is considered a frequently followed followee. Then, the algorithm focuses on those followers who are following this frequently followed followee Y. The algorithm then locates the second followee Z in the bitmap of these focused followers, and groups the followers following the same second followee Z. If the number of followers in a group of the second followee Z meets or exceeds a user-specified frequency threshold, then the group of followees {Y, Z} is considered a frequently followed group of followees. This step is repeated recursively to find all the frequently followed groups of followees containing/including the followee Y.

The aforementioned step (i.e., backtracking to focus on the suffix of each bitmap) is repeated recursively to find all the frequently followed groups of followees containing/including the remaining followees.

4.1 An Example of Discovering Frequently Followed Groups of Followees from a Social Network Represented by a Collection of MPLWAH(3) Compressed Bitmaps

Reconsider our aforementioned social network sample with seven followers (Gigi, Henry, Iris, John, Kat, Leo and Monica). Each follower is following a list of followees represented by MPLWAH(3) compressed bitmaps. To discover frequently followed groups of followees, our data science solution first locates first followee in each bitmap:

- As the first 32-bit word in the MPLWAH(3) compressed bitmap for Monica is 10 0 0 10100 00000 00000 0 1010 1010 1010, her first followee (in terms of user ID) is at position $31 \times 1010\ 1010\ 1010_{(2)} + 10100_{(2)} = 31 \times 2730 + 20 = 84{,}650$, i.e., Alice (user #84650).
- As the first 32-bit word in the MPLWAH(3) compressed bitmaps for Leo, Kat, Iris and Henry is identical to that for Monica, their first followee is also at position 84,650, i.e., Alice (user #84650).
- In contrast, the first 32-bit word in the MPLWAH(3) compressed bitmaps for John and Gigi is 10 0 0 01000 00000 00000 1 0101 0101 0011, their first followee is

at position $31 \times 1\ 0101\ 0101\ 0011_{(2)} + 01000_{(2)} = 31 \times 5459 + 8 = 169{,}237$, i.e., Bob (user #169237).

Our data science solution then groups the followers who are following the same first followee:

- Henry, Iris, Kat, Leo and Monica follow Alice (user #84650 having the smallest user ID among the followees of the seven followers in this sample social network); whereas
- Gigi and John follow another user having a bigger user ID.

For this sample social network, if the user specifies the frequency threshold at 3, then {Alice} (user #84650) is a *frequently followed followee* because she is followed by 5 followers (Henry, Iris, Kat, Leo and Monica).

Then, our solution focuses on these 5 followers, and observes their second followees: Kat and Monica follow Bob (user #169237 having the second smallest user ID among the followees of these five followers who also follow Alice). However, with only 2 followers, {Alice, Bob} is *not* considered as a frequently followed group of followees.

Continue with the 5 followers afterwards, and observe their next followees (after Bob): Henry and Monica follow Carol (user #169248 having the next smallest user ID among the followees of these five followers who also follow Alice). Hence, {Alice, Carol} is a *frequently followed pair of followees* because Alice & Carol are followed together by 3 followers (Henry, Kat and Monica).

Focusing on these 3 followers, and observes their third followees: Kat and Monica follow Eva (user #253667 having the third smallest user ID among the followees of these three followers who also follow both Alice and Carol). However, with only 2 followers, {Alice, Carol, Eva} is *not* considered as a frequently followed group of followees.

Then, backtrack and continue with the 5 followers (Henry, Iris, Kat, Leo and Monica), and observe their next followees (after Carol): Kat and Monica follow Don (user #253661 having the next smallest user ID among the followees of these five followers who also follow Alice). However, with only 2 followers, {Alice, Don} is *not* considered as a frequently followed group of followees.

Continue with the 5 followers (Henry, Iris, Kat, Leo and Monica), and observe their next followees (after Don): Henry, Iris and Monica follow Eva (user #253667 having the next smallest user ID among the followees of these five followers who also follow Alice). Hence, {Alice, Eva} is a *frequently followed pair of followees* because Alice & Eva are followed together by 3 followers (Henry, Iris and Monica). Among these 3 followers, only Monica follows one more followee (Fred the user #253673). Hence, {Alice, Eva, Fred} is *not* is a frequently followed group.

After discovering all frequently followed groups containing Alice, our data science solution backtracks, locates the next followee (after Alice) in each bitmap, and groups the followers who are following the same next followee:

- Gigi, John, Kat and Monica follow Bob (user #169237 having the smallest user ID among the followees—after Alice—of the seven followers in this sample social network); whereas
- Henry follows a user having a bigger user ID;
- Iris follows another user having an even bigger user ID; but
- Leo does not follow any other user.

Hence, {Bob} is a *frequently followed followee* because he is followed by 4 followers (Gigi, John, Kat and Monica).

Then, our solution focuses on these 4 followers, and observes their second followees: John, Kat and Monica follow Carol (user #169248 having the second smallest user ID among the followees of these four followers who also follow Bob). Hence, {Bob, Carol} is a *frequently followed pair of followees* because Bob & Carol are followed together by 3 followers (John, Kat and Monica). Focusing on these 3 followers, and observes their next few followees: With only Kat and Monica follow Don, {Bob, Carol, Don} is *not* a frequently followed group. Similarly, with only John and Monica follow Eva, {Bob, Carol, Eva} is also *not* a frequently followed group.

Backtrack to focuses on these 4 followers (Gigi, John, Kat and Monica), observe their next few followees:

- With only Kat and Monica follow Don, {Bob, Don} is *not* a frequently followed group.
- However, with Gigi, John and Monica follow Eva, {Bob, Eva} is a *frequently followed pair of followees*. Focusing on these 3 followers (Gigi, John and Monica), with only Monica follows Fred, {Bob, Eva, Fred} is *not* a frequently followed group.

After discovering all frequently followed groups containing Alice and groups containing Bob, our data science solution backtracks, locates the next followee (after Bob) in each bitmap, and groups the followers who are following the same next followee:

- Henry, John, Kat and Monica follow Carol (user #169248 having the smallest user ID among the followees—after Bob—of the seven followers in this sample social network); whereas
- Gigi and Iris follow another user having a bigger user ID; but
- Leo does not follow any other user

Hence, {Carol} is a *frequently followed followee* because she is followed by 4 followers (Henry, John, Kat and Monica). Applying the same procedure recursively, our data science solution finds {Carol, Eva} as a *frequently followed pair of followees*.

After discovering all frequently followed groups containing Alice, groups containing Bob, and groups containing Carol, our data science solution applies the same procedure—i.e., backtracks, locates the next followee after Carol (then Don, etc.) in each bitmap. Consequently, it finds {Eva} as a *frequently followed followee*.

To summarize, by applying the social network mining procedure recursively, our data science solution discovers nine *frequently followed groups*:

- {Alice}, who is followed by Henry, Iris, Kat, Leo and Monica;
- {Alice, Carol}, who are followed by Henry, Kat and Monica;
- {Alice, Eva}, who are followed by Henry, Iris and Monica;
- {Bob}, who is followed by Gigi, John, Kat and Monica;
- {Bob, Carol}, who are followed by John, Kat and Monica;
- {Bob, Eva}, who are followed by Gigi, John and Monica;
- {Carol}, who is followed by Henry, John, Kat, Leo and Monica;
- {Carol, Eva}, who are followed by Henry, John and Monica; and
- {Eva}, who is followed by Gigi, Henry, Iris, John and Monica.

5 Evaluation

To evaluate the performance of our efficient and flexible compression model, we compared the memory usage and runtime of the uncompressed bit vectors with bit compressed bitmaps using WAH, IPLWAH(k) for $1 \leq k \leq 5$, and MPLWAH(k) for $1 \leq k \leq 4$ by using the same datasets used for evaluation in related works. Specifically, we used real-life datasets from the Stanford Large Network Dataset Collection [49] from the Stanford Network Analysis Platform (SNAP):

- ego-Facebook [50], which captures anonymized social circles (i.e., friend lists) containing 88,234 edges among 4039 nodes from Facebook.
- ego-Gplus [50], which captures shared social circles containing 13,673,453 directed edges among 107,614 nodes from Google+ (aka G+). Here, there are 9,168,660 edges among 69,501 nodes in the largest strongly connected component (i.e., strongly connected graph or digraph, in which every node having an in-degree of at least 1).
- ego-Twitter [50], which captures social circles containing 1,768,149 directed edges among 81,306 nodes from Twitter. Here, there are 1,685,163 edges among 68,413 nodes in the largest strongly connected component.

To avoid distraction, among the consistent evaluation results on these three datasets, we show those on the ego-Twitter dataset. In the evaluation, we varied the size of the social network by selecting some subsets of the ego-users.

5.1 Evaluation on Memory Consumption

First, we measured the total number of 32-bit words in the uncompressed bit vector vs. the compressed bitmaps using different compression models ranging from WAH,

to IPLWAH(k) for $1 \leq k \leq 5$, and to MPLWAH(k) for $1 \leq k \leq 4$. Note that IPLWAH(1) is identical to MPLWAH(1).

Each ego-user in Twitter usually follows many (e.g., hundreds of) other Twitter users as followees, and multiple ego-users may follow the same frequently followed groups of followees. To evaluate the scalability of the compression models, we varied the size of the social network by selecting some subsets of the ego-users. Specifically,

- we randomly selected 20 ego-users as our first data point. Here, the size of the uncompressed bit vector (in the adjacency matrix) of this first data point is around 20 followers × their 3406 followees followed by at least one of those 20 selected followers. In the actual storage, by using a 32-bit word to represent the 'following' relationship, the uncompressed bit vector of this first data point consumes:
  20 followers × ⌈their 3406 followees ÷ 32 bits⌉ = 2,140 words.
- Similarly, we randomly selected 50 ego-users as our second data point. Here, the size of the uncompressed bit vector of this second data point is around 50 followers × their 6171 followees followed by at least one of those 50 selected followers. By using a 32-bit word to represent the 'following' relationship, the uncompressed bit vector of this second data point consumes:
  50 followers × ⌈their 6171 followees ÷ 32 bits⌉ = 9,650 words.
- We also randomly selected 200 ego-users as our third data point. Here, the size of the uncompressed bit vector of this third data point is around 200 followers × their 22,062 followees followed by at least one of those 200 selected followers. By using a 32-bit word to represent the 'following' relationship, the uncompressed bit vector of this second data point consumes:
  200 followers × ⌈their 22,062 followees ÷ 32 bits⌉ = 138,000 words.
- Similarly, we randomly selected 500 ego-users as our fourth data point. Hence, the size of the uncompressed bit vector of this fourth data point is around 500 followers × their 48,418 followees followed by at least one of those 500 selected followers. By using a 32-bit word to represent the 'following' relationship, the uncompressed bit vector of this second data point consumes:
  500 followers × ⌈their 48,418 followees ÷ 32 bits⌉ = 757,000 words.
- Finally, we used all 973 ego-users as our fifth data point. Hence, the size of the uncompressed bit vector of this fifth data point is around 973 followers × their 81,306 followees followed by at least one of those 973 followers, for a total of 1,768,149 'following' relationships. By using a 32-bit word to represent the 'following' relationship, the uncompressed bit vector of this second data point consumes:
  973 followers × ⌈their 81,306 followees ÷ 32 bits⌉ = 2,472,393 words.

Sizes of uncompressed bit vectors of other data points, as well as of other compressed bitmaps, are shown in Table 1 and plotted in Fig. 1.

Figure 1 shows that, when the number of ego-users increases, the total numbers of words required to capture their 'following' relationships also increases. The numbers grow rapidly for the uncompressed bit vectors. The numbers are lower

Table 1 Evaluation results on the total number of 32-bit words

	#ego-users (i.e., #followers)				
#words	20	50	200	500	973
Uncompressed	2140	9650	138,000	757,000	2,472,393
WAH	1538	6327	40,575	119,198	240,835
IPLWAH(1)	1373	5148	25,822	68,704	132,464
IPLWAH(2)	1302	4738	23,718	64,816	127,115
IPLWAH(3)	1274	4586	23,446	64,472	126,717
IPLWAH(4)	1263	4550	23,409	64,426	126,668
MPLWAH(2)	1198	4121	20,502	58,724	118,565
MPLWAH(3)	1033	3319	17,379	52,860	110,270
MPLWAH(4)	888	2738	15,116	48,551	105,943

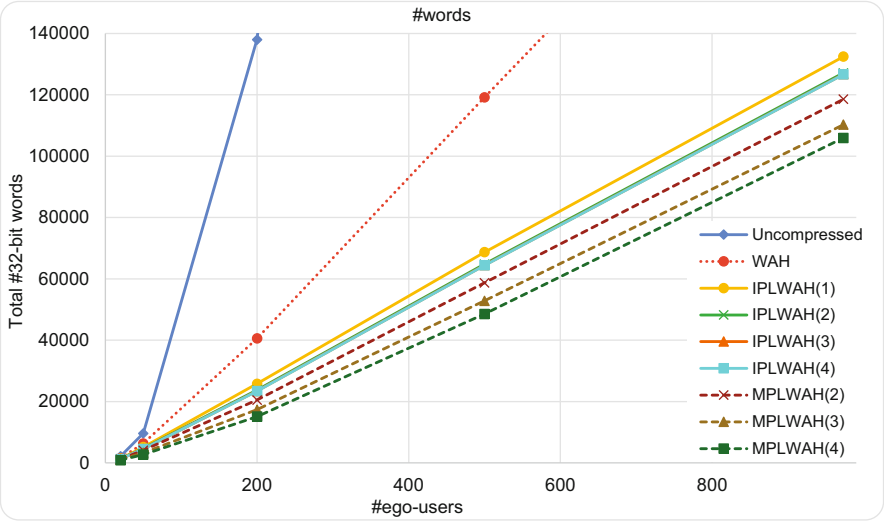


Fig. 1 Evaluation results on the total number of 32-bit words

for WAH. Those for IPLWAH(k) and MPLWAH(k) are much lower. Between the IPLWAH(k) and MPLWAH(k) compression models, the latter provide more effective compression and lead to less memory consumption required by the total numbers of 32-bit words.

Next, in addition to the total number of 32-bit words, we examined the compression ratios by measuring the relative memory consumption required by the uncompressed bit vector vs. the compressed bitmaps using different compression models ranging from WAH, to IPLWAH(k) for $1 \leq k \leq 5$, and to MPLWAH(k) for $1 \leq k \leq 4$. Both Table 2 and Fig. 2 show that, when the number of ego-users increases, the relative memory usage required to capture their 'following' relationships decreases. For high number of ego-users (i.e., followers), the relative memory usage drops from 100% (for the uncompressed bit vectors) to ~9.7% (for

Table 2 Evaluation results on the relative memory usage

	#ego-users				
%memory usage	20	50	200	500	973
Uncompressed	100%	100%	100%	100%	100%
WAH	71.87%	65.56%	29.40%	15.75%	9.74%
IPLWAH(1)	64.16%	53.35%	18.71%	9.08%	5.36%
IPLWAH(2)	60.84%	49.10%	17.19%	8.56%	5.14%
IPLWAH(3)	59.53%	47.52%	16.99%	8.52%	5.13%
IPLWAH(4)	59.02%	47.15%	16.96%	8.51%	5.12%
MPLWAH(2)	55.98%	42.70%	14.86%	7.76%	4.80%
MPLWAH(3)	48.27%	34.39%	12.59%	6.98%	4.46%
MPLWAH(4)	41.50%	28.37%	10.95%	6.41%	4.29%

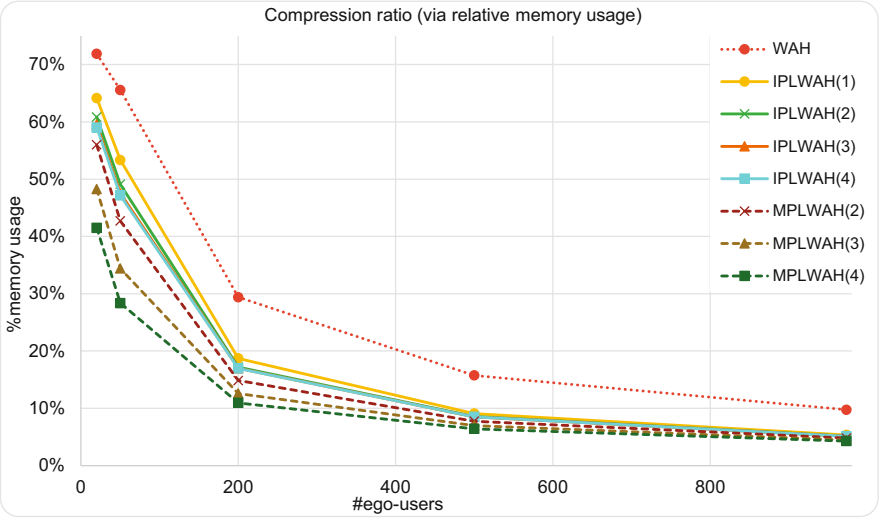


Fig. 2 Evaluation results on the relative memory usage

WAH compressed bitmap), and to ~5% (for IPLWAH compressed bitmaps) and further ~4% (for MPLWAH compressed bitmaps).

5.2 Evaluation on Runtime

Besides the memory consumption, we also evaluated the runtime of using our data science solution with different compression models. Experiments were run on a machine with Intel(R) Core(TM) i7-7700 CPU @ 3.60GHz and a memory of 16GB double data rate fourth generation synchronous dynamic random-access memory (DDR4 SDRAM) @ 2400 GHz. Moreover, the runtime includes CPU and

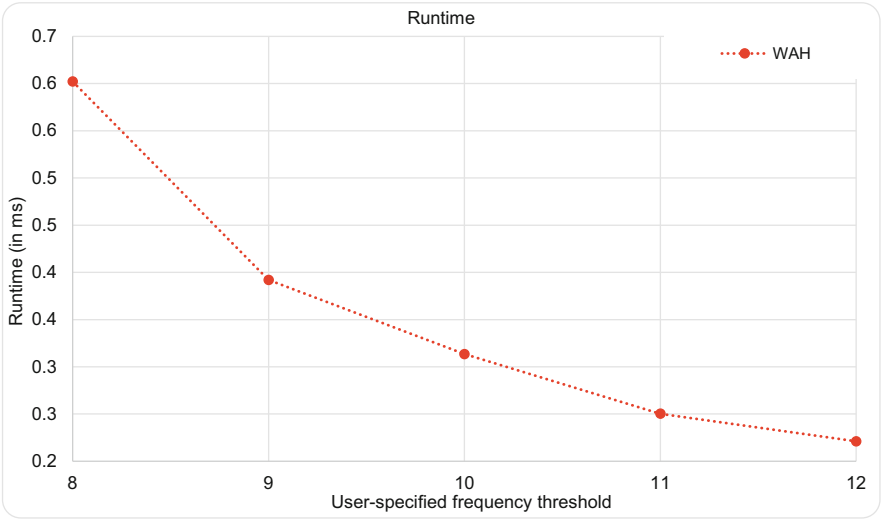


Fig. 3 Evaluation results on the runtime: WAH

I/Os, as well as encoding and decoding (compressing and uncompressing) of the bitmap. Results were the average of 10 runs. Figures 3, 4 and 5 all show that, when the user-specified frequency threshold increases, runtime decreases because the number of discovered frequently followed groups of followees increases. Among the IPLWAH(k) compression models, when k increases, the runtime slightly increases because of slightly more time spent on decoding the more compressed bitmap. This comment also applies to the runtimes of MPLWAH(k) compression models.

5.3 Evaluation on Scalability

In addition to evaluating the memory consumption and runtime, we also evaluated the scalability of our compression models and our data science solution for social network mining. Figures 1 and 2 show not only the memory consumption but also the scalability. Specifically, Fig. 1 shows that, when the number of ego-users (i.e., the number of followers) increases, the corresponding number of 32-bit words required to capture the followees of these followers also increases proportionally. This demonstrates the scalability of our compression models.

Figure 2 shows that, when the number of ego-users increases, the corresponding relative memory usage decreases. Moreover, IPLWAH bitmaps are more compressed than WAH bitmaps. Similarly, MPLWAH bitmaps are even more compressed than IPLWAH bitmaps. Hence, MPLWAH bitmaps use the least memory space, whereas IPLWAH bitmaps use more and WAH bitmaps use even more memory space. Among the variants of IPLWAH(k) bitmaps, the higher the k value,

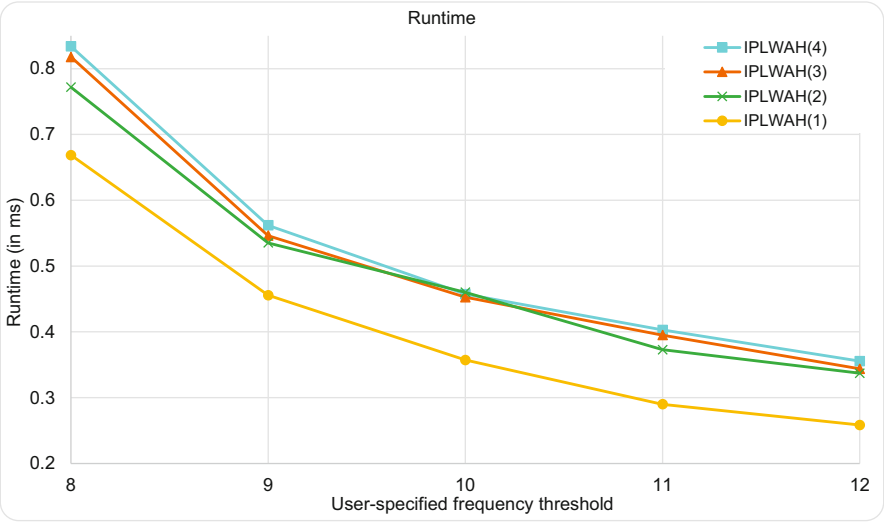


Fig. 4 Evaluation results on the runtime: IPLWAH(k)

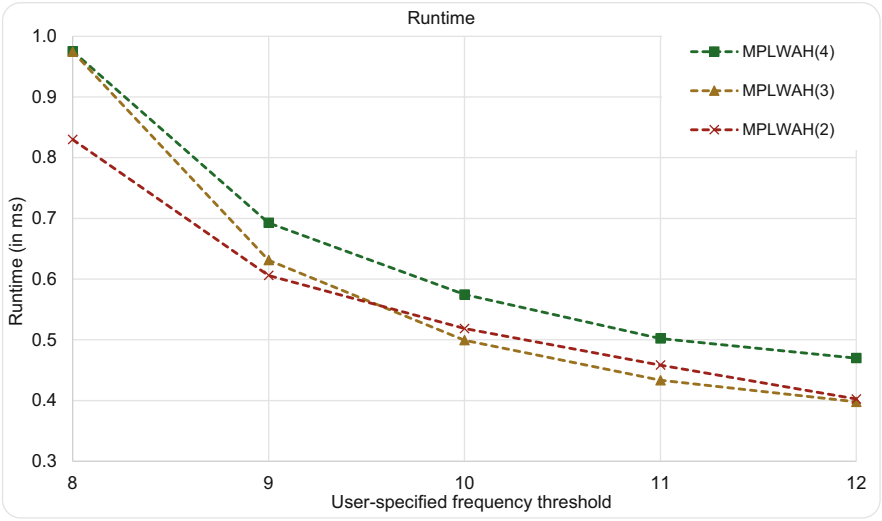


Fig. 5 Evaluation results on the runtime: MPLWAH(k)

the lower is the memory usage. Similar observations apply to the variants of MPLWAH(k) bitmaps: The higher the k value, the lower is the memory usage. This figure demonstrates, once again, the scalability of our compression models.

Figures 6, 7 and 8 shows that, when fixing the user-specified frequency threshold (to 9 followers), an increase in the number of ego-users (i.e., followers) leads to an increase in the number of frequently followed groups of followees followed by these

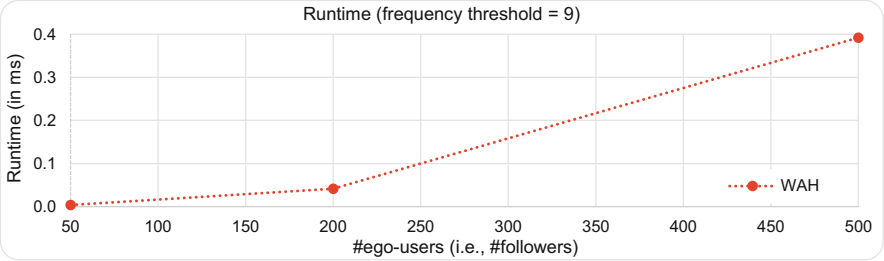


Fig. 6 Evaluation results on the scalability: WAH

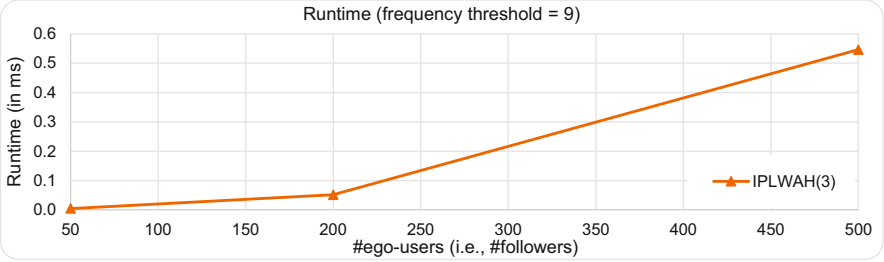


Fig. 7 Evaluation results on the scalability: IPLWAH(3)

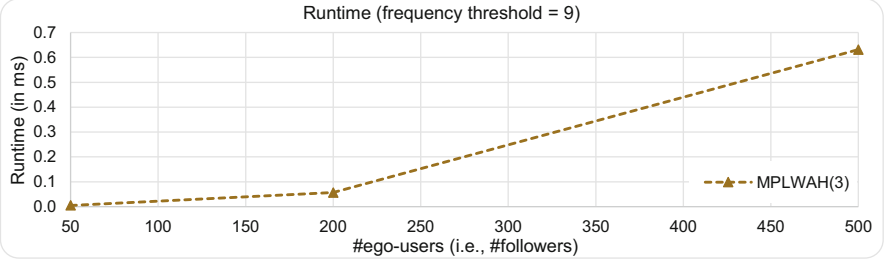


Fig. 8 Evaluation results on the scalability: MPLWAH(3)

followers. Hence, the corresponding runtime increases in running our data science solution to support social network mining in the discovery of frequently followed groups of followees. These figures demonstrate the scalability of our data science solution in supporting social network analysis and mining.

6 Conclusion

Nowadays, social networks are widely used by different users, who often express their personal opinions, browse interesting contents, and follow other favorite individuals on social networks. Consequently, plenty of valuable data can be

generated and collected, and information and knowledge embedded in these big social network data can be discovered via social network analysis and mining. One such information is the frequently followed groups of followees because these followees are usually influential and are of interest to many other followers. However, the discovering of these frequently followed groups could be challenging because the corresponding "following" relationships (from which the frequently followed groups can be mined) are sparse in very big social networks.

Hence, in the current chapter, we focused on efficient and flexible compression of very sparse networks of big data. In particular, we first discussed different representations of social networks: graphs (including bipartite graphs), adjacency matrices, and collections of bit vectors. Among them, collections of bit vectors are considered as logical representations of very sparse networks. We then presented several compression models: word-aligned hybrid (WAH), improved position list word-aligned hybrid (IPLWAH(k) for $1 \leq k \leq 5$), and multi-group position list word-aligned hybrid (MPLWAH(k) for $1 \leq k \leq 4$) compression models. Among them, MPLWAH(k) is observed to be an efficient and flexible compression model to effectively capture very sparse networks of big data. In addition, we also described a data science solution in supporting social network analysis and mining to discover frequently followed groups of followees from the MPLWAH(k) compressed bitmaps capturing important information about the "following" relationships between followers who follow their followees. Evaluation results show the effectiveness of the MPLWAH(k) compression models and their corresponding data science solution—in terms of memory consumption, runtime, and scalability—in supporting social network analysis and mining.

As ongoing and future work, we explore models to further compress the social networks, especially very sparse networks of big data. Moreover, we also explore ways to further enhance social network analysis and mining.

Acknowledgement This work is partially supported by (1) Natural Sciences and Engineering Research Council (NSERC) of Canada, and (2) University of Manitoba.

References

1. Xylogiannopoulos, K. F., Karampelas, P., & Alhajj, R. (2019). Multivariate motif detection in local weather big data. In *IEEE/ACM ASONAM 2019* (pp. 749–756). ACM.
2. Han, K., et al. (2019). Efficient and effective algorithms for clustering uncertain graphs. *Proceedings of the VLDB Endowment, 12*(6), 667–680.
3. Ke, X., Khan, A., & Quan, L. L. H. (2019). An in-depth comparison of s-t reliability algorithms over uncertain graphs. *Proceedings of the VLDB Endowment, 12*(8), 864–876.
4. Leung, C. K. (2014). Uncertain frequent pattern mining. In *Frequent pattern mining* (pp. 417–453).
5. Leung, C. K., Mateo, M. A. F., & Brajczuk, D. A. (2008). A tree-based approach for frequent pattern mining from uncertain data. In *PAKDD 2008. LNCS (LNAI)* (Vol. 5012, pp. 653–661).
6. Leung, C. K., & Carmichael, C. L. (2009). FpVAT: A visual analytic tool for supporting frequent pattern mining. *ACM SIGKDD Explorations, 11*(2), 39–48.

7. Leung, C. K., et al. (2020). Big data visualization and visual analytics of COVID-19 data. In *IV 2020* (pp. 387–392). https://doi.org/10.1109/IV51561.2020.00073.
8. O'Halloran, S., et al. (2017). Computational data sciences and the regulation of banking and financial services. In *From social data mining and analysis to prediction and community detection* (pp. 179–209).
9. Gupta, P., et al. (2020). Vertical data mining from relational data and its application to COVID-19 data. In *Big data analyses, services, and smart data* (pp. 106–116). https://doi.org/10.1007/978-981-15-8731-3_8.
10. Leung, C. K., et al. (2020). Data science for healthcare predictive analytics. In *IDEAS 2020* (pp. 8:1–8:10). ACM.
11. Olawoyin, A. M., Leung, C. K., & Choudhury, R. (2020). Privacy-preserving spatio-temporal patient data publishing. In *DEXA 2020, Part II. LNCS* (Vol. 12392, pp. 407–416).
12. Pawliszak, T., et al. (2020). Operon-based approach for the inference of rRNA and tRNA evolutionary histories in bacteria. *BMC Genomics 21*, (Supplement 2), 252:1–252:14.
13. Souza, J., Leung, C. K., & Cuzzocrea, A. (2020). An innovative big data predictive analytics framework over hybrid big data sources with an application for disease analytics. In *AINA 2020. AISC* (Vol. 1151, pp. 669–680).
14. Vural, H., Kaya, M., & Alhajj, R. (2019). A model based on random walk with restart to predict circRNA-disease associations on heterogeneous network. In *IEEE/ACM ASONAM 2019* (pp. 929–932). ACM.
15. Hoang, K., et al. (2020). Cognitive and predictive analytics on big open data. In *ICCC 2020. LNCS* (Vol. 12408, pp. 88–104).
16. Leung, C. K., et al. (2020). Data mining on open public transit data for transportation analytics during pre-COVID-19 era and COVID-19 era. In *INCoS 2020. AISC* (Vol. 1263, pp. 133–144).
17. Fan, C., et al. (2018). Social network mining for recommendation of friends based on music interests. In *IEEE/ACM ASONAM 2018* (pp. 833–840). IEEE.
18. Fariha, A., et al. (2013). Mining frequent patterns from human interactions in meetings using directed acyclic graphs. In *PAKDD 2013, Part I. LNCS (LNAI)* (Vol. 7818, pp. 38–49).
19. Ghaffar, F., et al. (2018). A framework for enterprise social network assessment and weak ties recommendation. In *IEEE/ACM ASONAM 2018* (pp. 678–685). IEEE.
20. Jiang, F., Leung, C. K., & Tanbeer, S. K. (2012). Finding popular friends in social networks. In *CGC 2012* (pp. 501–508). IEEE.
21. Leung, et al. (2018). Mining 'following' patterns from big but sparsely distributed social network data. In *IEEE/ACM ASONAM 2018* (pp. 916–919). IEEE.
22. Leung, C. K., Tanbeer, S. K., & Cameron, J. J. (2014). Interactive discovery of influential friends from social networks. *Social Network Analysis and Mining, 4*(1), 154:1–154:13.
23. Patel, H., Paraskevopoulos, P., & Renz, M. (2018). GeoTeGra: A system for the creation of knowledge graph based on social network data with geographical and temporal information. In *IEEE/ACM ASONAM 2018* (pp. 617–620). IEEE.
24. Rafailidis, D., & Crestani, F. (2018). Friend recommendation in location-based social networks via deep pairwise learning. In *IEEE/ACM ASONAM 2018* (pp. 421–4428). IEEE.
25. Tanbeer, S. K., Leung, C. K., & Cameron, J. J. (2014). Interactive mining of strong friends from social networks and its applications in e-commerce. *Journal of Organizational Computing and Electronic Commerce, 24*(2–3), 157–173.
26. Vaanunu, M., & Avin, C. (2018). Homophily and nationality assortativity among the most cited researchers' social network. In *IEEE/ACM ASONAM 2018* (pp. 584–586). IEEE.
27. Leung, C. K., et al. (2018). Big data analytics of social network data: Who cares most about you on Facebook? In *Highlighting the importance of big data management and analysis for various applications* (pp. 1–15). https://doi.org/10.1007/978-3-319-60255-4_1.
28. Mai, M., et al. (2020). Big data analytics of Twitter data and its application for physician assistants: Who is talking about your profession in twitter? In *Data management and analysis* (pp. 17–32). https://doi.org/10.1007/978-3-030-32587-9_2.
29. O'Halloran, S., et al. (2019). A data science approach to predict the impact of collateralization on systemic risk. In *From security to community detection in social networking platforms* (pp. 171–192).

30. Leung, C. K. (2020). Data science for big data applications and services: Data lake management, data analytics and visualization. In *Big data analyses, services, and smart data* (pp. 28–44). https://doi.org/10.1007/978-981-15-8731-3_3.
31. Das, A., et al. (2019). Water governance network analysis using Graphlet mining. In *IEEE/ACM ASONAM 2019* (pp. 633–640). ACM.
32. Leung, C. K. (2020). Big data computing and mining in a smart world. In *Big data analyses, services, and smart data* (pp. 15–27). https://doi.org/10.1007/978-981-15-8731-3_2.
33. Leung, C. K. (2018). Frequent Itemset mining with constraints. In *Encyclopedia of database systems* (2nd ed., pp. 1531–1536).
34. Arora, U., Paka, W. S., & Chakraborty, T. (2019). Multitask learning for blackmarket tweet detection. In *IEEE/ACM ASONAM 2019* (pp. 127–130). ACM.
35. Leung, C. K., MacKinnon, R. K., & Wang, Y. (2014). A machine learning approach for stock price prediction. In *IDEAS 2014* (pp. 274–277). ACM.
36. Leung, C. K., Jiang, F., & Zhang, Y. (2019). Flexible compression of big data. In *IEEE/ACM ASONAM 2019* (pp. 741–748). ACM.
37. Cao, Y., et al. (2020). Hybrid deep learning model assisted data compression and classification for efficient data delivery in mobile health applications. *IEEE Access, 8*, 94757–94766.
38. Jiang, H., & Lin, S. (2020). A rolling hash algorithm and the implementation to LZ4 data compression. *IEEE Access, 8*, 35529–35534.
39. Birman, R., Segal, Y., & Hadar, O. (2020). Overview of research in the field of video compression using deep neural networks. *Multimedia Tools and Applications, 79*(17–18), 11699–11722.
40. Fu, H., Liang, F., & Lei, B. (2020). An extended hybrid image compression based on soft-to-hard quantification. *IEEE Access, 8*, 95832–95842.
41. Kumar, K. S., Kumar, S. S., & Kumar, N. M. (2020). Efficient video compression and improving quality of video in communication for computer encoding applications. *Computer Communications, 153*, 152–158.
42. Liu, T., & Wu, Y. (2020). Multimedia image compression method based on biorthogonal wavelet and edge intelligent analysis. *IEEE Access, 8*, 67354–67365.
43. Hossein, S. M., et al. (2020). DNA sequences compression by GP^2 R and selective encryption using modified RSA technique. *IEEE Access, 8*, 76880–76895.
44. Kounelis, F., & Makris, C. (2020). Comparison between text compression algorithms in biological sequences. *Information and Computation, 270*, 104466:1–104466:8.
45. Hernández, C., & Marín, M. (2013). Discovering dense subgraphs in parallel for compressing web and social networks. In *SPIRE 2013. LNCS* (Vol. 8214, pp. 165–173).
46. Liu, Z., Ma, Y., & Wang, X. (2020). A compression-based multi-objective evolutionary algorithm for community detection in social networks. *IEEE Access, 8*, 62137–62150.
47. Leung, C. K., et al. (2016). Mining "following" patterns from big sparse social networks. In *IEEE/ACM ASONAM 2016* (pp. 923–930). IEEE.
48. Leung, C. K., & Jiang, F. (2017). Efficient mining of "following" patterns from very big but sparse social networks. In *IEEE/ACM ASONAM 2017* (pp. 1025–1032). ACM.
49. Leskovec, J., & Krevl, A. (2014). *SNAP datasets: Stanford large network dataset collection*. http://snap.stanford.edu/data.
50. McAuley, J., & Leskovec, J. (2012). Learning to discover social circles in ego networks. In *NIPS 2012* (pp. 548–556).

Weather Big Data Analytics: Seeking Motifs in Multivariate Weather Data

Konstantinos F. Xylogiannopoulos, Panagiotis Karampelas, and Reda Alhajj

Abstract For the past few years, climate changes and frequent disasters that are attributed to extreme weather phenomena have received considerable attention. Technical advancement both in hardware, such as sensors, satellites, cluster computing, etc., and analytical tools such as machine learning, deep learning, network analysis, etc., have allowed the collection and analysis of a large volume of complex weather related data. In this chapter, we study the European capital temperatures by implementing the novel "General Purpose Sequence Clustering" methodology (GPSC), which allows to analyze and cluster numerous long time series using commercial widely available hardware of low cost. Using the specific methodology, we have managed to cluster two-years temperature time series of 38 European capitals. This is not just based on typical seasonality but in a more in-depth level using complex patterns. The results showed the efficiency and effectiveness of the methodology by identifying several clusters showing similarities that could help weather specialists in discovering more advanced weather prediction models.

Keywords Weather analysis · Data mining · Weather data analytics · ARPaD · LERP-RSA

K. F. Xylogiannopoulos
Department of Computer Science, University of Calgary, Calgary, AB, Canada
e-mail: kxylogia@ucalgary.ca

P. Karampelas (✉)
Department of Informatics & Computers, Hellenic Air Force Academy, Dekelia Air Force Base, Attica, Greece
e-mail: panagiotis.karampelas@hafa.haf.gr

R. Alhajj
Department of Computer Science, University of Calgary, Calgary, AB, Canada
e-mail: alhajj@ucalgary.ca

© The Author(s), under exclusive license to Springer Nature Switzerland AG 2021

M. Çakırtaş, M. K. Ozdemir (eds.), *Big Data and Social Media Analytics*, Lecture Notes in Social Networks, https://doi.org/10.1007/978-3-030-67044-3_10

1 Introduction

Human activities are interwoven with the weather conditions either for deciding what is the most appropriate clothes for commuting to one's workplace, organizing outdoor activities, travel plans, or for cultivating and growing long-term agricultural production in an area, among others. Recognizing the high dependency of their activities on the weather phenomena, humans started early their effort to predict the weather by visually searching the sky for cues that will lead them to sense the anticipated evolution of the weather the next day, the following days, or even weeks. They usually based their prediction on their knowledge of historical trends combined with their common sense.

Progressively, with the evolution of sciences, other techniques and methodologies were used in the effort to forecast the weather phenomena. Early in the twentieth century, several scientists such as Vilhelm Bjerknes, Felix Exner and Lewis Richardson who are considered the grand fathers of meteorology attempted to model weather phenomena using hydrodynamic and thermodynamic equations [1]. Then, using numerical algorithms and providing them with weather data observed, they attempted to solve the equations and use the results to forecast specific weather variables. The first Numerical Weather Predictions (NWPs) were a reality, albeit inaccurate or even completely wrong due to the scarce weather data available at the time. Progressively, the advancement of the computer systems (including hardware and software) and weather sensors such as radars, satellites, etc., helped in the development of more accurate weather forecasting models [2–4] since it was possible to collect and analyze several different weather data. Recently, the government of Great Britain announced a huge investment of 1.2 billion pounds in a supercomputer that will be able to analyze weather and clime data shortening the time period needed for weather prediction from 3 to 1 h and thus improve "daily to seasonal forecasts and longer-term climate projections" [5]. Cloud computing infrastructures, on the other hand, make up appropriate ground in many countries in the race of more accurate weather prediction aspiring to supersede traditional supercomputers. This is possible because cloud computing offers greater scalability with less cost [6–7].

Another reason that led to extensive interest in improving weather forecasting is climate change which is defined as the long-term weather changes which have lately affected many areas around the world [8]. The cost of climate change which is translated to the destruction caused by wildfire, hurricanes and typhoons was estimated to be more than 85 million dollars in 2018 [9], while in 2019 the cost for typhoons, wildfires and flooding around the world was estimated to be more than a billion dollars [10]. According to United Nations Weather agency [11] more than 62 million people were affected by phenomena associated with climate change in 2018. On the other hand, in 2019 this number raised to 72 million people being either displaced or became food insecure due to extreme weather phenomena [12]

because of limited natural resources such as water or crops and livestock [13–14]. According to various scientific reports based on weather predictions, the following years are expected to be warmer [15] something which will have direct impact to the sea-ice all over the world. An immediate effect of such a global warming will be higher risk for flooding in areas due to abnormal high precipitation, and thus novel weather forecasting methods are required to better predict such phenomena [16].

In this context, various organizations all over the world collect and analyze weather data, including temperature, wind direction, speed and gust, visibility, pressure and pressure tendency collected either locally or worldwide using a series of different sensors such terrestrial weather stations, weather nano satellites, weather balloons, and weather satellites. Analyzing this volume of historical data would provide a first-class opportunity to improve existing weather forecasting models or developing novel techniques to analyze such big data sets.

In this respect, this book chapter proposes a novel data analytics methodology that attempts to take advantage of the LERP-RSA data-structure [17–19] and the ARPaD algorithm [17, 19] which are able to analyze billions of data points and discover all repeated patterns in extremely big data series. Thus, it has been able to detect similar patterns in different weather variables that influence each other. The specific methodology was first introduced in [20] by analyzing data, such as the temperature at the 2m height above ground and the geopotential height (gpm) for the isobaric surface 500mbar from the National Oceanic and Atmospheric Administration (NOAA) [21] for a particular country. The experiments revealed that there are associations between the different atmospheric variables that are not known yet and can play an important role in future forecasting models. Extending the work in [20], in this book chapter the temperature in 41 European capitals is analyzed by applying also an additional clustering technique which reveals similar patterns between the temperatures in different areas.

The rest of the book chapter is organized as follows: Section 2 presents the related work in the field of weather prediction and curve clustering. Section 3 presents the proposed methodology for Weather Time Series Analysis and Clustering. Section 4 describes the experiment conducted using the available dataset [21] and presents the clusters identified. Section 5 discusses the potential of the proposed work and future extensions.

2 Related Work

While scientific interest in weather prediction was drawn very early in the 1900s attempting to study a limited range of available weather variables at that time using mainly numerical or stochastic methods [1] with a little success, it was not until later when computers were invented that was possible to develop more accurate models

for predicting temperature or rain. Progressively, with the use of more powerful computing infrastructure, it was possible to develop more complicated weather models able to analyze a wider range of weather variables and improve the accuracy of the predictions. At the same time, a massive number of weather sensors became available that were able to be used not only in the surface of the earth but also in different atmospheric heights. Weather balloons, weather satellites, nanosatellites, and earth sensors are used to collect and analyze measurements of a great number of weather variables such as air temperature, cloud frequency, geopotential height, gravity wave, heat flux, humidity, hydrostatic pressure and several other variables [21]. Based on those measurements, scientists have developed a very large number of prediction models either generic for short period weather prediction or more specialized forecasting, i.e., storm detection, downpour, maritime, agriculture, flight safety, climate change, etc. [3, 22–24]. Over the years, traditional numerical and stochastic models have been shifted to other scientific techniques and more specifically to techniques that can analyze large volumes of data. Thus, lately, data mining has been used in weather prediction since there are many data mining techniques that are used to discover knowledge based on historical data, something that now can be done in weather measurements since there are several datasets available. For example, there are historical weather data recorded from February 20, 1820 to March 1821 for the city Little Rock at Arkansas, USA. The data available in this dataset include daily measurements for temperature, wind direction and relative wind force [25]. As historical data plays important role in knowledge discovery [26], it is evident that data mining techniques can be used to analyze the available data and discover patterns that can help in weather prediction. Li et al. [24] have used several different classification methods in an attempt to detect severe storm patterns. They finally propose an evolution of the DBSCAN clustering algorithm [27] in order to form spatial clusters which then are ranked according to a measure that they propose called Storm Severity Factor (SSF).

In [23], the authors, based on a literature review they performed, reported all the data mining techniques that have been used for weather prediction related to agriculture. In another work, the authors [28] used k-means clustering algorithm to forecast air pollution in the atmosphere, while a multivariate analysis in simulating time series values for weather variables such as solar radiation, maximum and minimum temperature, average dew point temperature, average wind speed, and precipitation has been used for predicting precipitation [29]. The work described in [30] used a Support Vector Machine model to predict future precipitation based on previous weather data. A similar model based on Support Vector Machine was used in [31] for small wildfires by considering various variables such as rain, wind, temperature and humidity. However, the specific approach requires extensive offline training with historical data and cannot predict larger wildfires in real time. Another approach described in [22] is similar to the work presented in this book chapter; it uses deep learning to discover the interdependencies of geopotential

height, temperature, wind and dew point for short term weather prediction. The dataset used in the specific approach comes exclusively from data collected by considering weather balloons, while in our case the data comes from different sources, including terrestrial and satellite, and we are looking for repeated patterns in our multivariate analysis. In [32] a Bayesian Network in combination with two data mining algorithms, namely Tetrad II (a causal discovery program) and Causal MML (which discovers linear causal models) have been used in order to improve weather prediction in Australian coastline.

Kitamoto [33] used principal component analysis, k-means clustering methods, self-organizing maps and instance-based learning with k-nearest neighbors search in order to predict typhoons. In an effort to predict tornados, McGovern et al. [34] mined multi-variate time series with the purpose to identify patterns or motifs for weather variables interdependency. The specific technique is very similar to our approach even though the analysis of the weather variables was used in tornado prediction while our approach is more generic and can be applied for any combination of weather variables. Another difference is that in the specific approach the weather data used are not real data but simulated for that purpose; the authors have reported very good results. In our case, the data come from real weather measurements which are available through NOAA [21]. In addition, the authors in [34] applied a sampling method to select a small number of data points to analyze and in particularly extreme values. On the other hand, in our case, we perform exhaustive analysis and all repeated pattern detection in the complete data set. Discovering the interdependencies in weather variables also requires further meta-analysis, and in many cases clustering techniques have been used for improving weather forecasting [35–37]. Gutiérrez et al. [35] reported that the clustering method they have used in their model is more sensitive to detect extreme events than the traditional methods while computationally wise it is less demanding and more efficient. Additionally, as reported in [36] time-hierarchical clustering also led the authors to propose novel visualization methods that allow an intuitive observation of specific weather phenomena over time.

As highlighted in [37], an important factor in weather clustering is the selection of the appropriate distance measure in order to achieve the best results. As Ganguly and Steinhaeuser concluded [38], there is always a need for novel approaches in the analysis of weather variables. As it can be seen in [30–38], data mining techniques can be used to improve traditional numeric and stochastic methods and can help in improving weather forecasting. Along this line, we propose a novel methodology for analyzing weather time series in order to detect patterns and analyze them using a novel graph clustering technique.

3 Temperatures Time Series Analysis and Clustering

For the past few decades, weather analysis has accumulated significant amount of data due to advances in technology. Thousands of ground and satellite sensors measure practically in real time every possible value of land, atmospheric and oceanic parameters. Analyses in huge scales are performed because of advanced computational systems and supercomputers which allow the calculation of complex systems and models. Scientists are able to compare, validate and correlate several parameters from many different geolocations and draw very useful and important conclusions.

In this chapter, we perform an analysis on the temperature time series of all European capitals. The focus of the analysis is to try to identify possible similarities in temperature time series based on the shape of time series curves with significant accuracy and detail rather than simple seasonal occasions and characteristics.

In order to perform such analysis, the General Purpose Sequence Clustering methodology (GPSC) is used [39]. It allows the detection of similarities between any number of sequences, or more specifically time series. It is based on the novel data structure LERP-RSA [12–13] and the high-performance pattern detection algorithm ARPaD [12, 14]. Furthermore, GPSC allows the detection of similarities based on a variety of parameterizations such as the length of patterns, positions in the time series, skewness, etc. The time complexity of GPSC depends on the construction of the multivariate LERP-RSA data structure. It is $O(mn \log n)$ with regard to the input of m time series of length n. ARPaD algorithm time complexity is also $O(mn \log n)$ as it has been proved in [12, 13], and thus the overall performance of the proposed methodology is $O(mn \log n)$ although it can be further simplified to $O(mn)$.

In summary, the objectives of the proposed methodology may be listed as:

- To analyze temperature time series and identify all common repeated patterns over time by transforming weather data to LERP-RSA data structures, and then applying the ARPaD algorithm;
- To construct clusters by discovering similarities between temperature time series of different European capitals by discovering common repeated patterns.

The proposed methodology consists of five different stages in which the data are collected, transformed and analyzed. As a sixth step, it is possible to consider the meta-analysis of the results where the similarity thresholds are defined, and the clusters are constructed. The specific stages can also be executed on demand when new data become available, i.e., from the connected sensors or from the available weather databases. Accordingly, the results will be updated.

3.1 Data Acquisition

There is a plethora of weather datasets collected worldwide by different types of satellite and ground sensors which are owned by global meteorological observatories such as World Meteorological Organization [12], European Centre for Medium-Range Weather Forecasts [40], National Oceanic and Atmospheric Administration [21], or Universities and National Weather Services. Usually, these datasets are processed and organized and then they are made available to other stakeholders such as other weather services, researchers, etc. The available datasets most of the times contain multiple weather variables for a great region such as Europe, North America and other continents or greater areas of the globe. For the purpose of testing the specific methodology, a global dataset coming from the National Oceanic and Atmospheric Administration (NOAA) [21] was used. The analysis started with the selection of the main weather variable for the time series analysis and clustering and subsequently selecting from the map the different areas to analyze. It was decided to select all the main capitals of Europe, for the years 2015 and 2016, and explore whether their temperatures of height at 2m above the ground, in Kelvin units, allow the formation of time series clusters in the context of 2 years of observations. Since the dataset did not had coordinates of the cities explicitly but rather coordinates with 0.25° step, it was further processed and the most approximate coordinates to the real capital's coordinates were extracted. As described in Sect. 4, 38 capitals were selected and clustered following the proposed methodology.

3.2 Data Preparation and Curation

The first step after the collection of the data is to prepare them for the analysis. In order to execute the analysis though, it is fundamental to curate the data from extreme or missing values. First, we need to perform a reliability check for the data points in order to make sure that values represent real temperature values. For example, extreme values such as very low or high, below 250K and above 32K, have to be removed and replaced accordingly. Second, we need to check for missing values and try to fill them with the best possible candidate value. In order to achieve such curation, we have to execute the Least Common Multiple (LCM) method [39] which allows to perform a time series length transformation without losing information. After removing the extreme values and in combination with any missing values, we will have time series that are shorter than the expected length and they need to be expanded to the original length. Using LCM, we expand the time series to the LCM value between its length and the original length using interpolation. Then we reduce the time series to the original, correct, length.

3.3 Discretization

After the data are collected and curated, we need to proceed to the next step which is the discretization of the time series continuous values. Discretization is very important because it allows the creation of the LERP-RSA data structure and the execution of the ARPaD algorithm. The time series are first standardized using the Z-Scoring methodology by subtracting from each value the mean and divide the outcome by the standard deviation. This transformation will give us new time series with mean zero and standard deviation one for all the time series. Moreover, the time series maintain their shape which is the most important aspect since it will be used to compare them.

Furthermore, such transformation allows us to perform a very easy discretization. Having the values standardized means that the min max values are very small. For example, if we assume a Normal distribution then the min max values are minus three and plus three. Therefore, we can determine the number of classes we need and divide the min max spread with the number of classes. Each class has a character representation from the Latin alphabet. Using this alphabet, we transform the continuous values time series to a discrete sequence representation by a string.

3.4 LERP-RSA Construction

The next stage of the GPSC is the construction of the multivariate data structure LERP-RSA. The discretized temperature sequences are stored in the LERP-RSA data structure using three different columns. The first two columns define the standard information of the LERP-RSA which is the suffix string and the position in which the specific substring has been found. The third column defines the time series index, i.e., the capital city, from which the suffix string comes.

The process starts with the analysis of each capital time series into suffix strings which are stored in individual LERP-RSA data structures that are lexicographically sorted based on the suffix strings. Then all the different capital LERP-RSA data structures are merged together. The specific process, depending on the available hardware, can run in parallel by sorting the suffix strings based on their initial letter of the alphabet separately, and thus run each sorting process in different cores or threads in the processor (CPU) depending on the letter of the alphabet used.

3.5 ARPaD Pattern Discovery

The ARPaD algorithm can run when the common LERP-RSA data structure for all the weather variables is lexicographically sorted. The algorithm identifies all the repeated patterns found in the stored data structure irrespective of the length of the

pattern. For example, if the pattern *ababab* has been found 100 times, it is possible a smaller pattern such as *abab* to be found, e.g., 200 times since the second pattern is a subpart of the first pattern but also can be for 100 times subpart of another pattern such as *ababbb*. Of course, the analysis can be performed using patterns of length one, which is practically similar to distance-based algorithms. However, an important attribute of GPSC is that it can use longer patterns and eliminate noise in data, for example, by using patterns with length two or three characters. Moreover, the APRaD algorithm can detect patterns not only in the same sequence, but also between different sequences using the multivariate LERP-RSA. This is important because in order to determine the similarity between different sequences our main task is to identify patterns that occur at exactly the same positions on different time series (capitals).

3.6 Similarity Meta-analysis

The final stage of the sequence clustering is to identify the similarity score between all temperature time series. For this purpose, GPSC uses the Sequence Commonality Matrix and the Sequence Commonality Grouping matrix. These matrices, with one-pass scanning, record the percentage of similarities between all sequences. Based on a threshold that we can define and alter as needed, we can determine how the clusters are formed, and thus which time series are similar between them. This process has significant advantages because it allows, for example, to check similarities between small regions or regions with gaps, etc. Moreover, we can check for specific behaviors such as patterns, for example, *aaaa* representing constant values and horizontal lines or increasing/decreasing patterns, as for example, *abcd* or *dcba* representing diagonal lines.

4 Experimental Analysis

Based on the aforementioned methodology, an experiment was conducted for clustering the time series of 38 European capitals. The weather data that were used were analysis data extracted from NCEP's Global Forecast System (GFS) [21]. The specific dataset is mainly used in the numerical weather prediction models as initial conditions coming from all the available weather (satellite and ground) sensor observations using the Global Data Assimilation System (GDAS). From all the available weather variables the temperature at 2m height above ground was selected.

Concerning the infrastructure, a laptop with an Intel i7 CPU has been used, with quad core processors and 16GB RAM. A solid-state disk with 256GB has been also used to store, process and present analysis results. The results were then transferred

Table 1 Selected European capitals

Country name	Country code	Capital name	City code	Latitude	Longitude
Netherlands	NL	Amsterdam	AMS	52.35	4.92
Greece	GR	Athens	ATH	37.98	23.73
Romania	RO	Bucharest	BCH	44.43	26.1
Germany	DE	Berlin	BER	52.52	13.4
Serbia	RS	Belgrade	BLG	44.83	20.5
Slovakia	SK	Bratislava	BRA	48.15	17.12
Switzerland	CH	Bern	BRN	46.92	7.47
Belgium	BE	Brussels	BRU	50.83	4.33
Hungary	HU	Budapest	BUD	47.5	19.08
Moldova	MD	Chisinau	CHS	47	28.85
Denmark	DK	Copenhagen	COP	55.67	12.58
Finland	FI	Helsinki	HLS	60.17	24.93
Ukraine	UA	Kyiv	KIV	50.43	30.52
Slovenia	SI	Ljubljana	LJB	46.05	14.52
United Kingdom	GB	London	LON	51.5	−0.08
Luxembourg	LU	Luxembourg	LUX	49.6	6.12
Belarus	BY	Minsk	MIN	53.9	27.57
Monaco	MC	Monaco	MON	43.73	7.42
Cyprus	CY	Nicosia	NIC	35.17	33.37
Norway	NO	Oslo	OSL	59.92	10.75
France	FR	Paris	PAR	48.87	2.33
Montenegro	ME	Podgorica	PDG	42.43	19.27
Czech Republic	CZ	Prague	PRG	50.08	14.47
Latvia	LV	Riga	RIG	56.95	24.1
Italy	IT	Rome	ROM	41.9	12.48
North Macedonia	MK	Skopje	SKP	42	21.43
San Marino	SM	San Marino	SMR	43.93	12.42
Bulgaria	BG	Sofia	SOF	42.68	23.32
Bosnia and Herzegovina	BA	Sarajevo	SRJ	43.87	18.42
Sweden	SE	Stockholm	STO	59.33	18.05
Estonia	EE	Tallinn	TAL	59.43	24.72
Albania	AL	Tirana	TIR	41.32	19.82
Liechtenstein	LI	Vaduz	VAD	47.13	9.52
Malta	MT	Valletta	VAL	35.88	14.5
Austria	AT	Vienna	VIE	48.2	16.37
Lithuania	LT	Vilnius	VLN	54.68	25.32
Poland	PL	Warsaw	WRS	52.25	21
Croatia	HR	Zagreb	ZGR	45.8	16

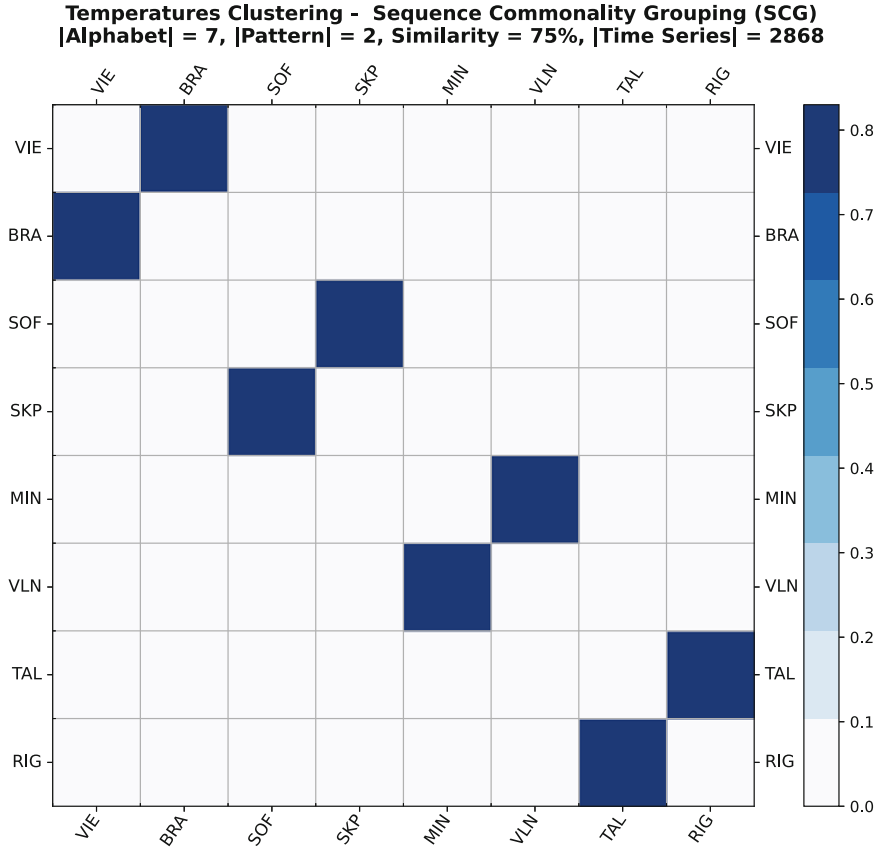


Fig. 1 Sequence Commonality Grouping matrices for 75% threshold

to a relational database, Microsoft SQL Server 2016, to facilitate meta-analysis with SQL.

For the experimental analysis, 38 European capitals were selected as listed in Table 1. The selection was merely guided by the acquired dataset which contained all the areas from longitude 0 to the East, and thus Madrid (Spain), Lisbon (Portugal) and Dublin (Ireland) are not included in the analysis (grayed in Fig. 5).

As mentioned above, in the dataset there were missing values for specific dates that were corrected using the methodology described in Sect. 3.2. For the experiment, we have used for discretization a seven characters alphabet. The discretization method is subjective to the desired period of the analysis, the model sensitivity, the number of time series or domain expert's experience.

For the GPSC meta analyses, we have used several similarity scores from 75% down to 60%. As we can observe in "Fig. 1" with threshold 75%, there are only

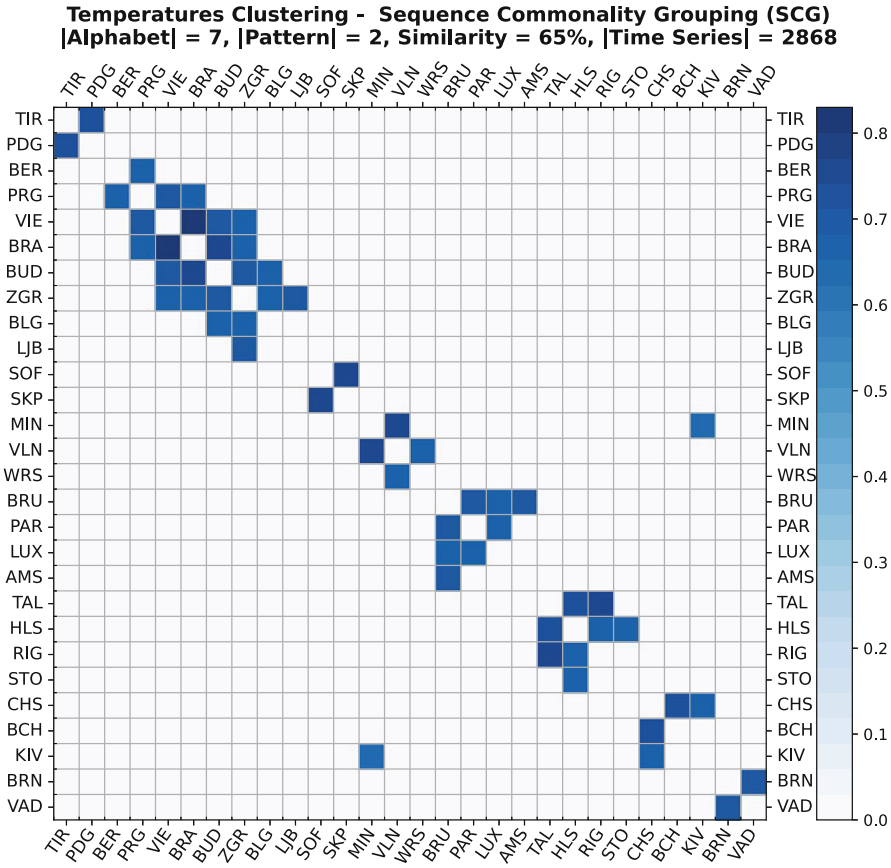


Fig. 2 Sequence Commonality Grouping matrices for 65% threshold

four clusters of two capitals and a total of just eight capitals. Yet, it is important that the temperature time series of the specific capitals are very similar compared to threshold 60% (Fig. 3) where we have practically ten clusters, but with some noise between the time series of 36 capitals. The 70% threshold (Fig. 2) works much better although it is high, it forms eight clusters with 28 capitals and is practically noiseless.

In “Fig. 4”, we can observe how each clustered time series performs. Moreover, on the Z-Scored time series transformation, a second-degree polynomial curve fitting has been applied; this show that the time series follow the same trends. In “Fig. 5”, the detected clusters have been illustrated on the map of Europe. Strong clusters with high threshold 75% are denoted with bold characters while clusters with lower threshold of 60% are denoted with the same color, but normal characters. It is interesting to observe how the temperatures of the capital cities cluster, and also how we experience some, maybe, unusual behaviors. For example,

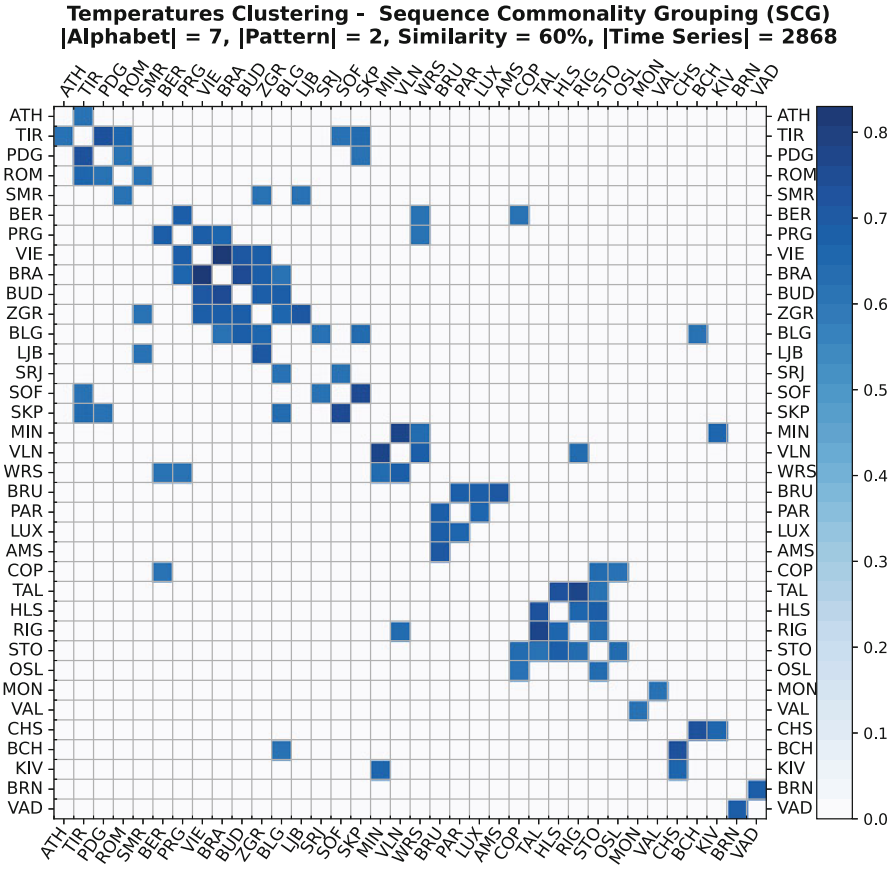


Fig. 3 Sequence Commonality Grouping matrices for 60% threshold

Monaco and Valletta have more similar temperature behavior compared to Rome which is practically between them. Also, Sarajevo has a fuzzy behavior since it is approximately in the same way similar to the Sofia cluster on the East and the much larger cluster of Belgrade on the North, but it is significantly less similar to the Podgorica cluster on the South ("Fig. 5").

5 Conclusions

In this chapter, we have attempted to study European capitals temperature by applying a sophisticated sequence clustering methodology (GPSC) in order to understand whether there are detailed similarities rather than only seasonal among the temperature of the cities in the span of multiple years. The specific methodology

could be expanded for weather specialists to further search and discover interdependencies between more than one weather variables which have not been studied together so far.

The experimental analysis runs on a subset of real analysis data coming from National Oceanic and Atmospheric Administration (NOAA) for finding similarity patterns among 38 European capital cities for temperature at 2m height above ground. The results are very promising since several clusters with different scoring thresholds have been identified showing interdependencies that may help the weather specialists to better understand local and global weather temperature patterns and possibly use the knowledge discovered to improve the existing weather prediction models.

We plan to continue working on the specific methodology using multi-variable datasets in collaboration with a weather specialist. The possibility of using multiple instead of single variable datasets would help the specialist to extract useful conclusions from multiple clustering and motifs discovery. Applying different visualization techniques related to the clustering of multiple weather variables will also enhance researcher's understanding of the identified results.

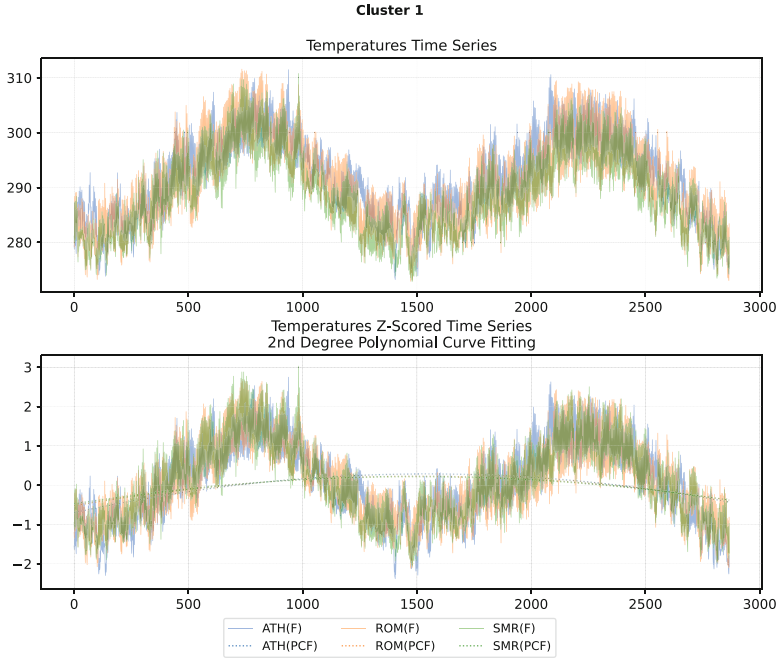


Fig. 4 Temperatures with Z-Scoring and second degree polynomial curve fitting per cluster

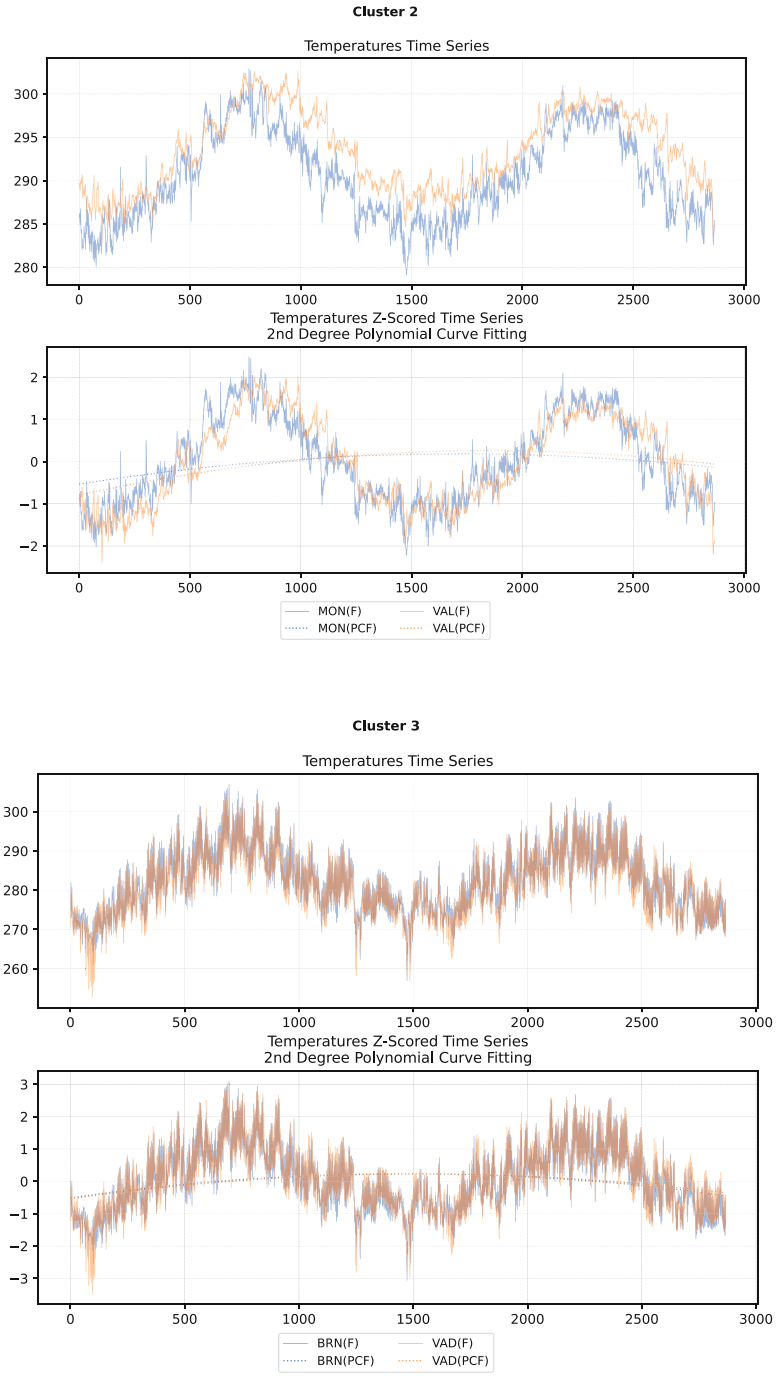


Fig. 4 (continued)

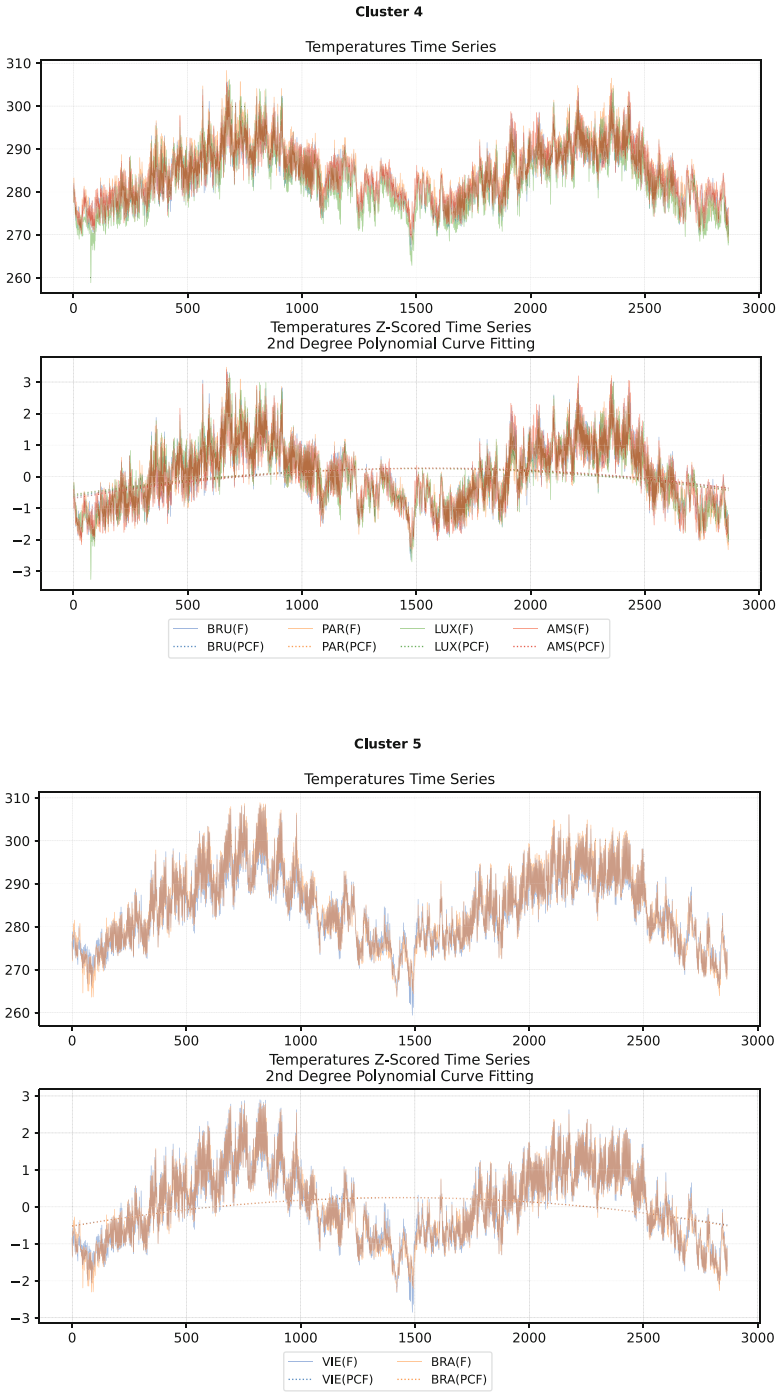
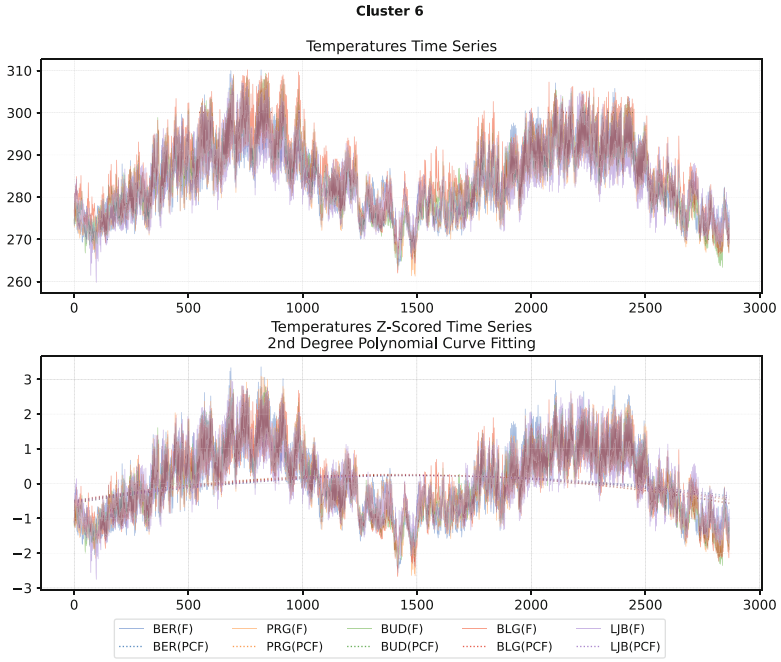


Fig. 4 (continued)



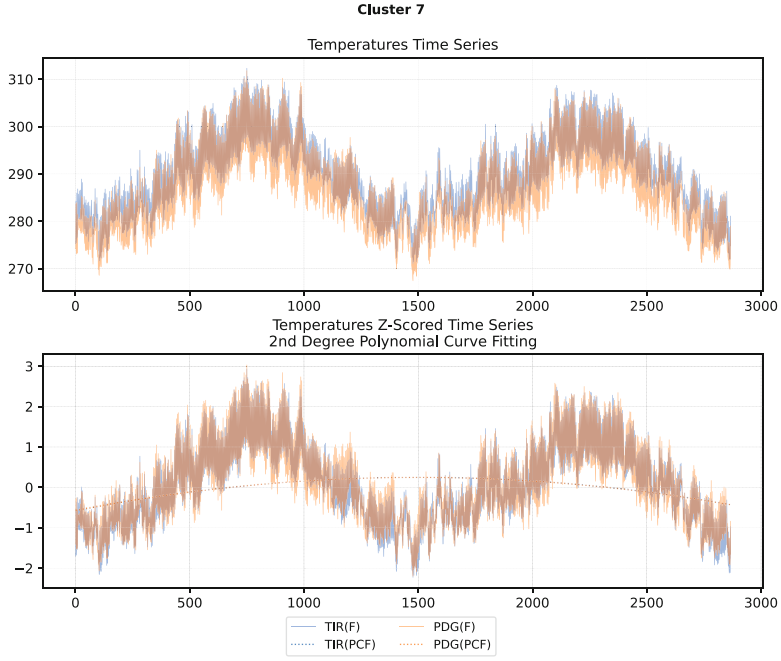
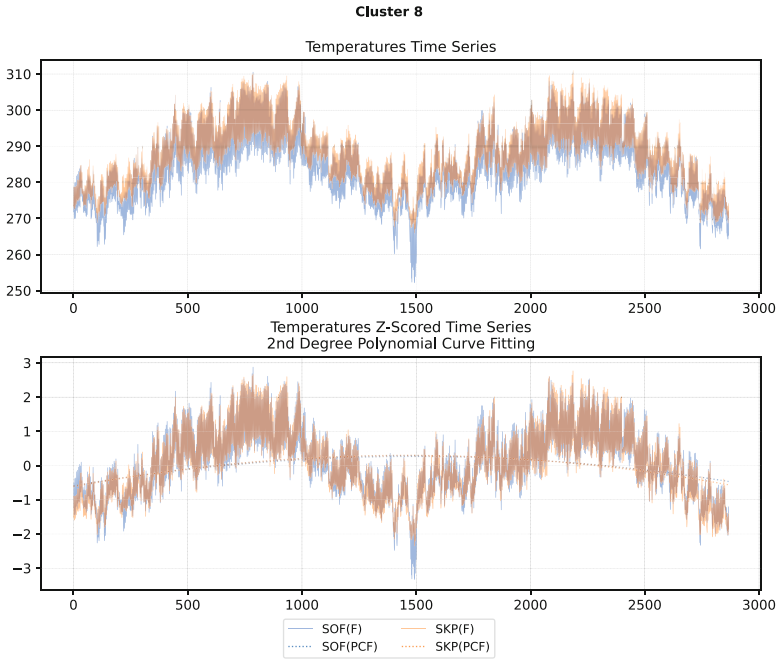


Fig. 4 (continued)



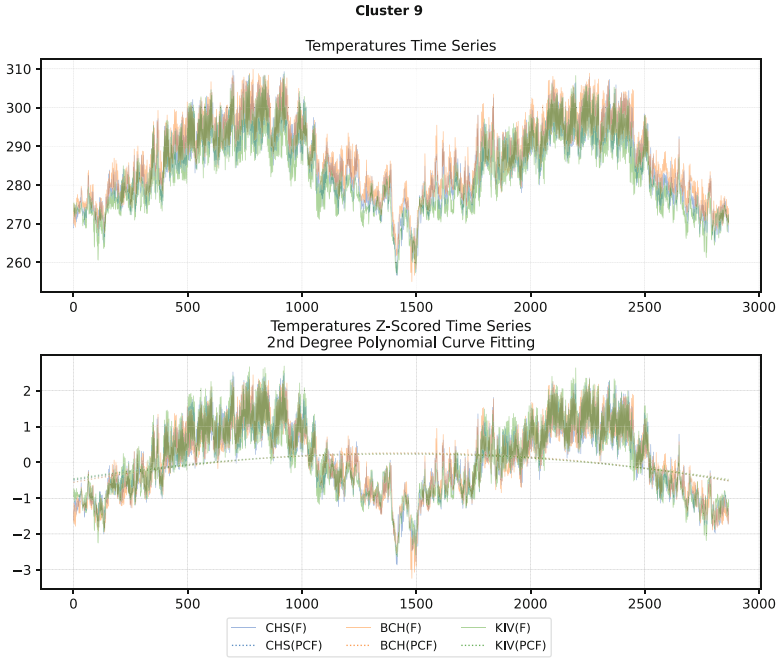
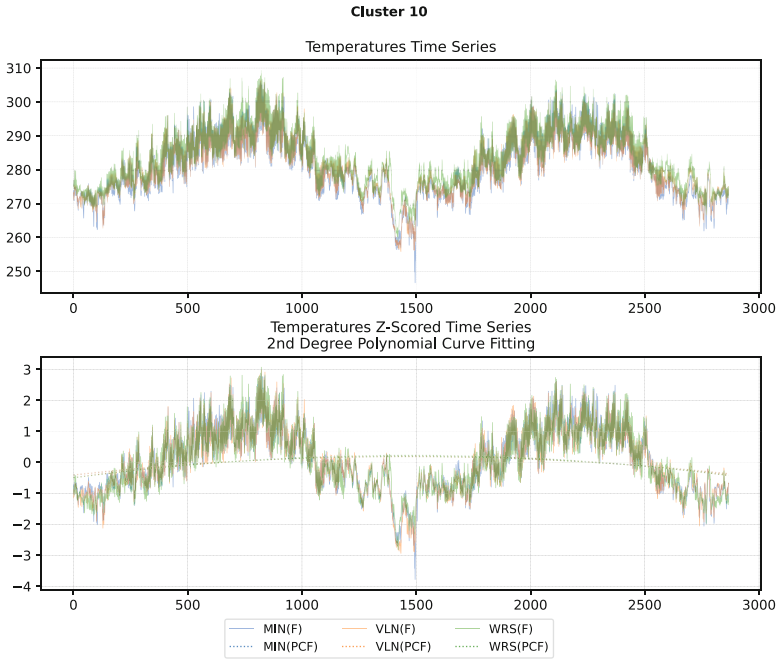


Fig. 4 (continued)



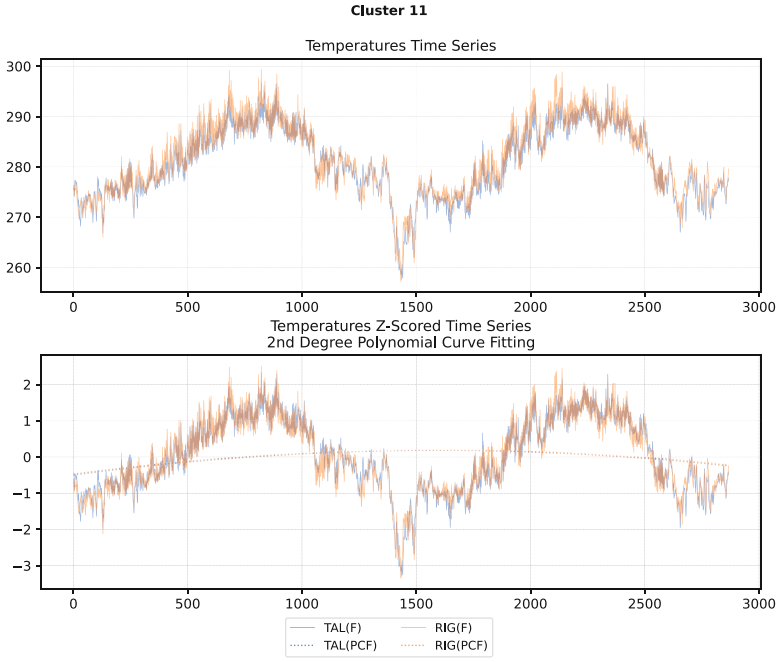


Fig. 4 (continued)

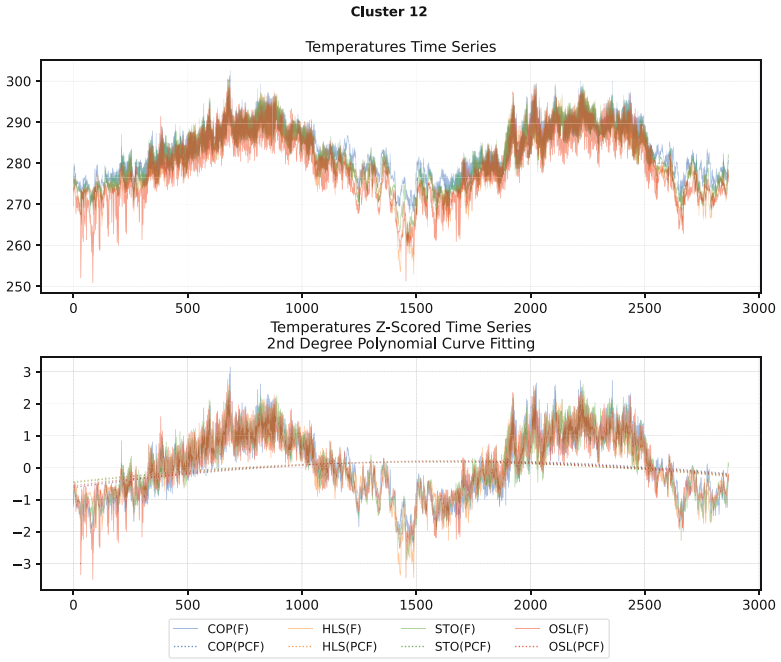


Fig. 4 (continued)

Fig. 5 Europe map with capital cities clustering

References

1. Lynch, P. (2006). *The emergence of numerical weather prediction: Richardson's dream*. Cambridge: Cambridge University Press.
2. Cressman, G. P. (1959). An operational objective analysis system. *Monthly Weather Review, 87*(10), 367–374.

3. Mahapatra, P., Doviak, R. J., Mazur, V., & Zrnić, D. S. (1999). *Aviation weather surveillance systems: Advanced radar and surface sensors for flight safety and air traffic management* (no. 8). IET.
4. Rao, P. K., Holmes, S. J., Anderson, R. K., Winston, J. S., & Lehr, P. E. (1990). *Weather satellites: Systems, data, and environmental applications*. Cham: Springer.
5. UK Gov. (2020). *£1.2 billion for the world's most powerful weather and climate supercomputer*. Retrieved February 20, 2020 from https://www.gov.uk/government/news/12-billion-for-the-worlds-most-powerful-weather-and-climate-supercomputer
6. Molthan, A. L., Case, J. L., Venner, J., Schroeder, R., Checchi, M. R., Zavodsky, B. T., Limaye, A., & O'Brien, R. G. (2015). Clouds in the cloud: Weather forecasts and applications within cloud computing environments. *Bulletin of the American Meteorological Society, 96*(8), 1369–1379.
7. Chang, V. (2017). Towards data analysis for weather cloud computing. *Knowledge-Based Systems, 127*, 29–45.
8. USGS. (2019). *How can climate change affect natural disasters?* Retrieved May 15, 2019 from https://www.usgs.gov/faqs/how-can-climate-change-affect-natural-disasters-1
9. Banis, D. (2018). 10 Worst climate-driven disasters of 2018 cost $85 billion. *Forbes*. Retrieved May 15, 2019 from https://www.forbes.com/sites/davidebanis/2018/12/28/10-worst-climate-driven-disasters-of-2018-cost-us-85-billion/#90f5aa22680b
10. Mack, E. (2019). In 2019 climate change made these 15 natural disasters even worse. *Forbes*. Retrieved February 20, 2020 from https://www.forbes.com/sites/ericmack/2019/12/27/climate-change-drove-the-price-tag-for-15-disasters-over-a-billion-dollars-each/#f3614a678441
11. Rice, D. (2019). Fueled by climate change, extreme weather disasters hit 62 million people in 2018, U.N. says. *USA Today.* Retrieved May 15, 2019 from https://eu.usatoday.com/story/news/nation/2019/03/29/extreme-weather-fueled-climate-change-disasters-hit-62-m-last-year/3304707002/
12. World Meteorological Organization (WMO). (2019). *2019 concludes a decade of exceptional global heat and high-impact weather*. Retrieved February 20, 2020 from https://public.wmo.int/en/media/press-release/2019-concludes-decade-of-exceptional-global-heat-and-high-impact-weather
13. UNHCR. (2019). *Climate change and disaster displacement*. Retrieved May 15, 2019 from https://www.unhcr.org/climate-change-and-disasters.html
14. NASA. (2019). *The impact of climate change on natural disasters*. Retrieved May 15, 2019 from https://earthobservatory.nasa.gov/features/RisingCost/rising_cost5.php
15. Harvey, F. (2019). *2020 to be one of hottest years on record, met office says*. Retrieved February 20, 2020 from https://www.theguardian.com/environment/2019/dec/19/2020-to-be-one-of-hottest-years-on-record-met-office-says
16. Kendon, E. J., Roberts, N. M., Fowler, H. J., Roberts, M. J., Chan, S. C., & Senior, C. A. (2014). Heavier summer downpours with climate change revealed by weather forecast resolution model. *Nature Climate Change, 4*(7), 570.
17. Xylogiannopoulos, K. F. (2017). *Data structures, algorithms and applications for big data analytics: Single, multiple and all repeated patterns detection in discrete sequences*. PhD thesis, University of Calgary.
18. Xylogiannopoulos, K. F., Karampelas, P., & Alhajj, R. (2016). Repeated patterns detection in big data using classification and parallelism on LERP reduced suffix arrays. *Applied Intelligence, 45*(3), 567–561.
19. Xylogiannopoulos, K. F., Karampelas, P., & Alhajj, R. (2014). Analyzing very large time series using suffix arrays. *Applied Intelligence, 41*(3), 941–955.
20. Xylogiannopoulos, K., Karampelas, P., & Alhajj, R. (2019, August). *Multivariate motif detection in local weather big data*. In Proceedings of the 2019 IEEE/ACM international conference on advances in Social Networks Analysis and Mining, pp. 749–756.
21. National Centers for Environmental Prediction/National Weather Service/NOAA/U.S. Department of Commerce. (2015). *Updated daily. NCEP GFS 0.25 Degree global forecast grids historical archive*. Research Data Archive at the National Center for Atmospheric Research,

Computational and Information Systems Laboratory. https://doi.org/10.5065/D65D8PWK. Accessed 19.05.2019.
22. Grover, A., Kapoor, A., & Horvitz, E. (2015, August). *A deep hybrid model for weather forecasting*. In Proceedings of the 21th ACM SIGKDD international conference on knowledge discovery and data mining (pp. 379–386). ACM.
23. Mucherino, A., Papajorgji, P., & Pardalos, P. M. (2009). A survey of data mining techniques applied to agriculture. *Operational Research, 9*(2), 121–140.
24. Li, X., Plale, B., Vijayakumar, N., Ramachandran, R., Graves, S., & Conover, H. (2008). Real-time storm detection and weather forecast activation through data mining and events processing. *Earth Science Informatics, 1*(2), 49–57.
25. National Centers for Environmental Prediction/National Oceanic and Atmospheric Administration/Paleoclimatology Program. (2020). *Historical climate data catalog*. https://www.ncdc.noaa.gov/data-access/paleoclimatology-data/datasets/historical. Accessed 25.02.2020.
26. Bramer, M. (2007). *Principles of data mining* (Vol. 180). London: Springer.
27. Ester, M., Kriegel, H. P., Sander, J., & Xu, X. (1996, August). *A density-based algorithm for discovering clusters in large spatial databases with noise*. In Kdd (Vol. 96, No. 34, pp. 226–231).
28. Jorquera, H., Perez, R., Cipriano, A., Acuna, G., & Zannetti, P. (2001). Short term forecasting of air pollution episodes. *Environmental Modeling, 4*, 3089–3101.
29. Rajagopalan, B., & Lall, U. (1999). A k-nearest-neighbor simulator for daily precipitation and other weather variables. *Water Resources Research, 35*(10), 3089–3101.
30. Tripathi, S., Srinivas, V. V., & Nanjundiah, R. S. (2006). Downscaling of precipitation for climate change scenarios: A support vector machine approach. *Journal of Hydrology, 330*(3–4), 621–640.
31. Cortez, P. & Morais, A. (2007, December). *A data mining approach to predict forest fires using meterological data* (pp. 512–23). Proceedings of the 13th EPIA 2007 – Portuguese Conference on Artificial Intelligence. Guimaraes, Portugal.
32. Kennett, R. J., Korb, K. B., & Nicholson, A. E. (2001, April). Seabreeze prediction using Bayesian networks. In *Pacific-Asia conference on knowledge discovery and data mining* (pp. 148–153). Berlin/Heidelberg: Springer.
33. Kitamoto, A. (2001, August). *Data mining for typhoon image collection*. In Proceedings of the second international conference on multimedia data mining (pp. 68–77). Springer.
34. McGovern, A., Rosendahl, D. H., Brown, R. A., & Droegemeier, K. K. (2011). Identifying predictive multi-dimensional time series motifs: An application to severe weather prediction. *Data Mining and Knowledge Discovery, 22*(1–2), 232–258.
35. Gutiérrez, J. M., Cofiño, A. S., Cano, R., & Rodríguez, M. A. (2004). Clustering methods for statistical downscaling in short-range weather forecasts. *Monthly Weather Review, 132*(9), 2169–2183.
36. Ferstl, F., Kanzler, M., Rautenhaus, M., & Westermann, R. (2016). Time-hierarchical clustering and visualization of weather forecast ensembles. *IEEE Transactions on Visualization and Computer Graphics, 23*(1), 831–840.
37. Izakian, H., Pedrycz, W., & Jamal, I. (2012). Clustering spatiotemporal data: An augmented fuzzy C-means. *IEEE Transactions on Fuzzy Systems, 21*(5), 855–868.
38. Ganguly, A. R., & Steinhaeuser, K. (2008, December). *Data mining for climate change and impacts*. In 2008 IEEE international conference on data mining workshops (pp. 385–394). IEEE.
39. Xylogiannopoulos, K. F. (2020). Data curves clustering using common patterns detection. *arXiv*.
40. European Centre for Medium-Range Weather Forecasts. Retrieved May 22, 2019 from https://www.ecmwf.int/en/forecasts/datasets

Analysis of Link Prediction Algorithms in Hashtag Graphs

Logan Praznik, Mohiuddin Md Abdul Qudar, Chetan Mendhe, Gautam Srivastava, and Vijay Mago

Abstract Twitter is a prominent multilingual social networking site where users can post messages known as "tweets". Twitter, like other social networking sites such as Facebook, allows users to categorize tweets by the use of "hashtags". Communication on Twitter can be mapped in terms of hashtag graphs, where vertices correspond to hashtags, and edges correspond to co-occurrences of hashtags within the same distinct tweet. Furthermore, a vertex in hashtag graphs can be weighted with the number of tweets a hashtag has occurred in, and edges can be weighted with the number of tweets both hashtags have co-occurred in, creating a "weighted hashtag graph". In this chapter, we describe additions to some well-known link prediction methods that allow the weights of both vertices and edges in a weighted hashtag graph to be taken into account. We base our novel predictive additions on the assumption that more popular hashtags have a higher probability to appear with other hashtags in the future. We then apply these improved methods to three sets of Twitter data with the intent of predicting hashtag co-occurrences in the future. In addition to these methods, we investigate the performance of a new, graph neural network-based framework, SEAL, which has been shown in past trials to perform better than heuristic-based approaches such as the Katz index, SimRank and rooted PageRank. Experiments were conducted on real-life data sets consisting of over 3,000,000 combined unique tweets and over 250,000 combined unique hashtags. Results from the experiments show that simpler heuristic-based scoring methods have marginal performance that decreases with the addition of more data

L. Praznik
Department of Mathematics and Computer Science, Brandon University, Brandon, MB, Canada

M. Md. Abdul Qudar · C. Mendhe · V. Mago
DaTALab, Department of Computer Science, Lakehead University, Thunder Bay, ON, Canada
e-mail: mabdulq@lakeheadu.ca; cmendhe@lakeheadu.ca; vmago@lakeheadu.ca

G. Srivastava (✉)
Research Center for Interneural Computing, China Medical University, Taichung, Taiwan, Republic of China

Department of Mathematics and Computer Science, Brandon University, Brandon, MB, Canada
e-mail: srivastavag@brandonu.ca

© The Author(s), under exclusive license to Springer Nature Switzerland AG 2021

M. Çakırtaş, M. K. Ozdemir (eds.), *Big Data and Social Media Analytics*, Lecture Notes in Social Networks, https://doi.org/10.1007/978-3-030-67044-3_11

over time. On the other hand, SEAL is shown to have superior performance in hashtag graph link prediction over the approaches it has been previously compared against in other domains. The AUC score of 0.959 obtained in our experiments by using SEAL significantly exceeds those of our benchmark approaches for link prediction, which include the Katz index, SimRank, and rooted PageRank.

Keywords Twitter · Link prediction · Hashtags · Data mining · Graphs

1 Introduction

In the popular micro-blogging platform known as Twitter,[1] users post short messages known as *tweets*. These tweets frequently contain *hashtags*, which are words, titles or phrases that can be used to connect a tweet to a specific topic, event, theme or conversation. Twitter is not alone in implementing this feature; other social networking platforms, such as Facebook and Instagram, also make use of hashtags. On Twitter, hashtags must begin with a pound sign (#) (eg: #worldcup) to be properly embedded in a tweet. Visible hashtags embedded in tweets can be clicked on to view other tweets using the same hashtag. In this way, they can help conversations to be centered around the same themes or topics, thus making it easier to search for and find tweets regarding those topics. Users can use as many hashtags as they want in a tweet, so long as they conform to Twitter's 280-character limit on the total length of tweets in general. This research work focuses on predicting links between the co-occurrences of hashtags, which are described in greater detail below.

Because multiple hashtags can be used in the same tweet, it is possible to construct a *hashtag graph* from a set of tweets, which shows how frequently different hashtags are used together. A hashtag graph provides information regarding how certain hashtags may be connected with each other, and therefore, may concern the same or related topics. It is possible to predict which pairs of hashtags that have not yet appeared in the same tweet will do so at some time in the near future [38]. These hashtag graphs can be used with appropriate link prediction methods to help forecast future topic directions within Twitter and other social media outlets. This fact alone makes it an important area of study. For example, the use of hashtags can not only help users identify posts as belonging to specific topics, but also help track the developments of a topic over time and help predict the future course a topic may take. For example, given a hashtag concerning a sports tournament, if we build a hashtag graph based on its usage with other connected hashtags, the links between the different hashtags could be used to predict who the perennial "fan favorite" is for that tournament and how it may change as the tournament progresses. This type of prediction may be used to cap betting payouts and set prices on tickets and merchandise for given teams and matches.

[1] www.twitter.com

Previously, traditional, heuristic-based methods such as the Katz index [11], rooted PageRank [14], and SimRank [10] have been shown to work very well in the domain of link prediction. However, these very approaches have been shown to fail in protein-protein interaction (PPI) networks, in which two proteins share many common neighbors but are actually less likely to interact [13]. Recently, a new framework for link prediction, SEAL, has been proposed, which uses a graph neural network (GNN) to acquire knowledge of the general graph structure. SEAL has been shown to perform better than these heuristic-based methods [40]. Therefore, the performance of SEAL when predicting links in hashtag graphs, versus that of the above named heuristic-based methods, is also deserving of investigation.

Since hashtag graphs are based on a different kind of connection than social graphs, such as between social media profiles or academic collaborators, there is a potential for hashtag graphs to exhibit a different structure than social graphs. In addition, nothing regarding the similarity or difference in structure between the two kinds of graphs has been proven, to our knowledge [19]. Therefore, the study of hashtag graphs, and more specifically, hashtag graph link prediction, is in its own class, and is still in its preliminary stages.

This chapter is an extended version of our work initially presented at the IEEE/ACM International Conference on Advances in Social Networks Analysis and Mining (ASONAM) in Vancouver, Canada in August of 2019 [25]. The rest of the chapter is organized as follows. Sections 2 and 3 discuss the background and motivation, and the basis of hashtag graphs and hashtag graph link prediction. Sections 4, 5, and 6 highlight the methodology, experimental setup, and results respectively. Finally, the chapter is concluded with a detailed look at future directions for research in Sect. 7.

2 Background and Motivation

The aim of link prediction in general is to predict whether two nodes in a network are likely to have a link [14]. It has been applied to multiple domains, including but not limited to movie recommendations [12] and relational machine learning for knowledge graphs [22].

Many researchers have made attempts to extract meaning from Twitter data using a variety of methods. For example, Bollen et al. [5] used Twitter sentiment analysis to predict the value of the Dow Jones Industrial Average. Varying degrees of success were experienced depending on the mood dimensions that were used, and a 87.6% overall rate of accuracy was achieved in predicting the stock index's daily upward and downward changes at market close [32].

Much of the existing literature on link prediction on Twitter focuses on links between Twitter profiles, rather than hashtags [2, 33]. Quercia et al. [27] evaluated the Latent Dirichlet Allocation (LDA) in the context of classifying the topics of Twitter profiles. Meanwhile, Valverde-Rebaza et al. [36] found that combining community information with structural information of networks of Twitter profiles could be used to increase the performance of link prediction.

In the area of hashtag graph link prediction, Wang et al. [38] used hashtag graphs to approach the problem of classifying hashtags by their sentiment polarity. Meanwhile, Wang et al. [39] used the hashtag graph topic model to discover semantic relations between words without requiring them to appear in the same tweet. Closely related to our work, Martinčić-Ipšić et al. [17] have studied the performance of link prediction in hashtag graphs in comparison to link prediction in *all-words graphs*, graphs of all words used in tweets. Using a number of existing link prediction methods, as well as two methods introduced in their work, they found that hashtag graphs exhibited similar characteristics to all-words graphs, and concluded that hashtags alone can be used to capture the context of tweets [31].

Researchers have also found success with link prediction using more sophisticated, non-heuristic methods. Zhang and Chen [40] argue that heuristic methods for link prediction, such as the ones explored and built upon in our work, are ineffective for graphs where the assumptions behind those heuristic methods fail. Using a GNN, they were able to consistently outperform a variety of heuristic methods, over graphs from different domains. Wang et al. [37] employed a deep convolutional neural network in the same way, with a similar improvement in performance over heuristic methods [35].

The SEAL model, which we are investigating in this chapter, has demonstrated good performance in multiple small datasets such as the `yeast` and `airbus` datasets, compared to more traditional approaches [40]. Unlike some previous heuristic approaches that can only consider up to 2-hop neighbours (i.e. predict links between two nodes if there is a distance of one or two "hops", or existing edges, between them) before losing sight of the structural information of the entire graph, SEAL has also been shown to work well with up to 3-hop neighbours.

For our contributions to the area of hashtag graph link prediction, we propose a novel graph model that takes into account both the hashtag occurrences, co-occurrences and weights of both edges and nodes to track this information. We attempt to improve the performance of a few well-known scoring methods used in link prediction by taking the weights of a graph's vertices as well as its edges into account [24, 29]. We evaluate their performance on hashtag graphs constructed from datasets collected from Twitter, and compare them to the established versions, which either ignore all weights or only consider edge weights, of those methods. In addition, we compared the performance of SEAL to the Katz index, SimRank and rooted PageRank in the domain of hashtag graph link prediction. We find the collection of work presented here to be novel in nature, and in the case of our application of SEAL, to be more accurate.

3 Foundation of the Hashtag Graph

We begin this section with some notation regarding hashtag graphs. We define the hashtag graph $G = \langle V, E \rangle$ to be an ordered pair, consisting of a set of vertices V, and a set of edges E, which itself consists of unordered pairs of vertices in V. Each

vertex in V represents a certain unique hashtag, while each edge in E represents an undirected connection between two hashtags. Furthermore, for a vertex $v \in V$, $\Gamma(v)$ is defined as the set of "neighbours" of v, meaning those vertices $u \in V$ for which there exists an edge $\{u, v\} \in E$. Wang et al. [38] have explored a hashtag graph model similar to the one defined up to this point, in the context of sentiment analysis.

For the model used in this chapter, both the vertices and edges of G are weighted. Each vertex $v \in V$ has an associated weight $w(v)$, while each edge $\{u, v\} \in E$ has an associated weight $w(u, v)$. In hashtag graphs in particular, $w(v)$ is the number of tweets that the hashtag represented by the vertex v has appeared in, while $w(u, v)$ is the number of tweets in which the hashtags represented by the vertices u and v have appeared concurrently. Both Martinčić-Ipšić et al. [17] and Wang et al. [39] have explored hashtag graphs, but neither has considered vertex weights. To our knowledge, there are no related works that follow a similar model to the one we define here.

3.1 Unweighted Heuristic Link Prediction Methods

This chapter builds on previous work on the subject of link prediction, for which many heuristic scoring methods have been developed. All of the methods in this section and Sect. 3.2 assign a "connection weight" $score(x, y)$ to an unordered pair of vertices $\{x, y\} \in V$. This connection weight, though lacking in any absolute scale, can be used as a relative measure within the same method to compare proximities between vertices [14].

We begin by introducing the three unweighted scoring methods that we have trialed, which take neither the weights of vertices nor the weights of edges into account. The first of these to be used in this chapter, The Common Neighbours (CN) method is based on the principle that vertices are more likely to form a connection when they are already connected to the same vertices as one another. Newman [21] conjectures this in the context of scientific collaboration, and concludes that the probability of collaboration between scientists is strongly positively correlated with the number of mutual acquaintances they share through past collaborations. Using all of the above definitions, the unweighted CN method is defined as follows:

$$S_{CN}(x, y) = |\Gamma(x) \cap \Gamma(y)| \tag{1}$$

Adamic and Adar [1], in the context of determining the strength of connections between users of personal Web pages, developed a method that could be considered an elaboration of the CN method. When considering the neighbours of x and y, those vertices that have fewer connections to others in the graph influence the score more heavily. Our second unweighted scoring method, the unweighted Adamic/Adar (AA) method, is given as:

$$S_{AA}(x, y) = \sum_{z \in \Gamma(x) \cap \Gamma(y)} \frac{1}{\log |\Gamma(z)|} \tag{2}$$

Our third and final unweighted scoring method comes from the principle of preferential attachment by Barabási and Albert [3], who found that new vertices in networks tend to attach preferentially to existing vertices that already have many connections. Barabási et al. [4] later proposed on the basis of empirical evidence that, for scientific social graphs, the probability that two authors will collaborate in the future is proportional to the product of the number of others each author had already collaborated with. Based on this, the unweighted Preferential Attachment (PA) method is given as:

$$S_{PA}(x, y) = |\Gamma(x)| \times |\Gamma(y)| \tag{3}$$

To illustrate the differences between these methods, we apply them to vertices x and y in the graph in Fig. 1. Firstly, for the Common Neighbours method, which is defined in Equation 1, note that vertices x and y have two common neighbours z_1 and z_2. More formally, we can say that $\Gamma(x) = \{z_1, z_2, z_3\}$ and $\Gamma(y) = \{z_1, z_2, z_4, z_5\}$, therefore $\Gamma(x) \cap \Gamma(y) = \{z_1, z_2\}$, the cardinality of which is equal to 2, and therefore $S_{CN}(x, y)$ is also equal to 2. For the Adamic/Adar method in Equation 2, however, each common neighbour's own set of neighbours is also a factor. In this case, z_1 has two neighbours, x and y, and z_2 has three neighbours, x, y and z_6. Therefore, $S_{AA}(x, y) = \frac{1}{\log 2} + \frac{1}{\log 3}$, which roughly equals 5.42. Finally, for the Preferential Attachment method in Equation 3, note that $|\Gamma(x)| = 3$ and that $|\Gamma(y)| = 4$. Then, $|\Gamma(x)| \times |\Gamma(y)| = 12$, which is the value for $S_{PA}(x, y)$. Again, note that these scores for the same pair of vertices have no meaning in isolation, but only have meaning when compared to other pairs of vertices that have been evaluated with the same heuristic scoring method.

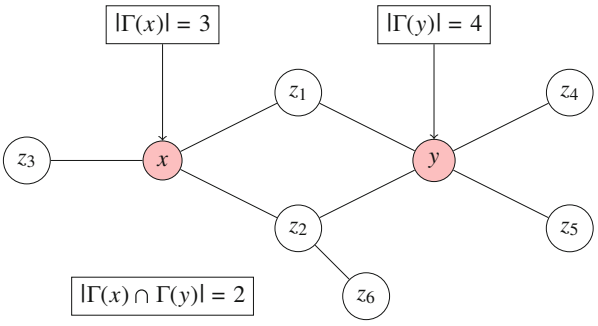


Fig. 1 An example of an unweighted graph. Here, $score(x, y)$ is being calculated between vertices x and y using the Common Neighbours, Adamic/Adar, and Preferential Attachment methods, as outlined in Equations 1, 2 and 3, respectively

3.2 *Edge-Weighted Heuristic Link Prediction Methods*

Murata and Moriyasu [20] define versions of the above methods that take the weights of edges in a graph into account. Their research was done in the context of social graphs formed on question-answering bulletin boards such as Yahoo! Answers, where edge weights correspond to the number of interactions between users. They found that taking edge weights into account improved the performance of the Adamic/Adar method in all of their experimental datasets, as well as the CN method in nearly all of the datasets used in their research. Meanwhile, they noticed only a slight improvement in the performance of the PA method, and only in those datasets where social graphs are relatively dense. Murata and Moriyasu's [20] edge-weighted versions of the CN (EN, Equation 4), AA (EA, Equation 5), and PA (EP, Equation 6) methods, respectively, are defined as follows:

$$S_{EN}(x, y) = \sum_{z \in \Gamma(x) \cap \Gamma(y)} \frac{w(x, z) + w(y, z)}{2} \tag{4}$$

$$S_{EA}(x, y) = \sum_{z \in \Gamma(x) \cap \Gamma(y)} \frac{w(x, z) + w(y, z)}{2 \log \sum_{z' \epsilon \Gamma(z)} w(z', z)} \tag{5}$$

$$S_{EP}(x, y) = \sum_{x' \cap \Gamma(x)} w(x', x) \times \sum_{y' \cap \Gamma(y)} w(y', y) \tag{6}$$

Once again, we will illustrate the differences between these methods by applying them to an example: in this instance, we will use vertices x and y in Fig. 2. The graph in this figure exhibits the same structure as Fig. 1, but has edge weights assigned to its edges. Firstly, for the edge-weighted Common Neighbours method, which is defined in Equation 4, the scores of the edges connecting x and y to their common neighbours z_1 and z_2 must also be taken into account. In the sum, the term for z_1 becomes $\frac{2+1}{2}$, while the term for z_2 becomes $\frac{1+1}{2}$. After adding the two terms, $S_{EN}(x, y) = 2.5$. Meanwhile, for the edge-weighted Adamic/Adar method in Equation 5, the edge weights in each common neighbour's connecting edges come into play in a similar manner. z_1 only has the edges that connect it to x and y, so its term in the sum becomes $\frac{2+1}{2 \log(2+1)} = \frac{2+1}{2 \log 5}$. However, z_2 also has an edge to z_6 with weight 3, making its term $\frac{1+1}{2 \log(1+1+3)} = \frac{1+1}{2 \log 5}$. $S_{EA}(x, y)$, then, comes out to approximately 4.57. Finally, the edge-weighted Preferential Attachment method in Equation 6 also considers edge weights in a similar manner, making $S_{EP}(x, y) = (2 + 1 + 1)(1 + 1 + 1 + 2) = 4 \times 5 = 20$.

As our vertex-and-edge-weighted versions of these methods, to our knowledge, are novel, we will be introducing them later in Sect. 4.

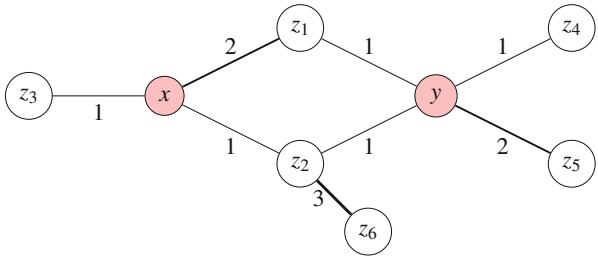


Fig. 2 An example of an edge-weighted graph. The structure is identical to 1, but with edge weights added. The edge-weighted versions of the Common Neighbours, Adamic/Adar, and Preferential Attachment methods, as outlined in Equations 4, 5 and 6, respectively, are being used to calculate $score(x, y)$. Thicker outlines are used to indicate weights greater than 1

3.3 Graph Neural Network Link Prediction with SEAL

3.3.1 SEAL

SEAL [40] is a recently-proposed method for link prediction. Instead of employing any one heuristic for all graphs, it learns from the graph it is given and "derives" a function that can explain and predict link formations. This function takes as input the links around the local enclosing subgraphs, and outputs the probability of the existence of the links. In summary, SEAL works in three steps, the first of which is extraction of enclosing subgraphs. In order to learn the appropriate function for the graph, a GNN must first be trained over the enclosing subgraphs. Therefore, the enclosing subgraphs must be extracted from a set of sampled positive (observed) links and a set of sampled negative (unobserved) links [9]. Secondly, the node information matrix X for each enclosing subgraph must be built. X has three main components: structural node labels, node embeddings, and node attributes. This step is highly important for training a successful GNN link prediction model. Finally, the GNN learns from its input (A, X), where A is the adjacency matrix of the enclosing subgraph input, and X is the node information matrix in which each row corresponds to a node's feature vector.

3.3.2 Node Labelling

The very first component in X is each node's structural label. A node labelling is a function $f_l : V \rightarrow N$ in which an integer is assigned as a label to every node i in the enclosing subgraph. When the target nodes are the center nodes x and y, where the link is located, the nodes have different relative positions with respect to the center, these nodes tend to have different structural importance to the link. If nodes are labelled properly, then the GNN should be able to mark these type of

differences. Without proper labelling, the GNN will not be able to identify the target nodes between which links exist, and will therefore lose structural information.

Node labels must follow two criteria. Firstly, the target nodes should have label 1, so that these nodes can be distinguished from other nodes. Secondly, nodes i and j should have the same label if $d(i,x) = d(j,x)$ and $d(i,y) = d(j,y)$. This is because a node i's topological position within an enclosing subgraph can be described by its radius with respect to the two center nodes, namely $(d(i,x), d(i,y))$. Therefore, nodes of the same orbit will have the same label. From these two criteria, the double-radius node labeling (DRNL) method was proposed [40, 41]. In DRNL, 1 is first assigned as a label to x and y, then, for any node i with $(d(i,x), d(i,y)) = (1,1)$, the label $f_l(i) = 2$ is assigned. Nodes with radius $(1,2)$ or $(2,1)$ get label 3, so repeatedly larger values are assigned as labels to nodes with larger radius with respect to the center nodes, in which double-radius $(d(i,x), d(i,y))$ satisfy the following:

1. If $d(i,x) + d(i,y) \neq d(j,x) + d(j,y)$, then $d(i,x) + d(i,y) < d(j,x) + d(j,y)$, $fl(i) < fl(j)$.
2. If $d(i,x)+d(i,y) = d(j,x)+d(j,y)$, then $d(i,x)d(i,y) < d(j,x)d(j,y)$, $fl(i) < fl(j)$.

Although there are other methods of node labeling, DRNL was empirically verified to result in better performance with SEAL than not using labeling, or instead using one of many naive labeling methods. DRNL's perfect hashing function is one of its main advantages. The method has the condition that $f_l(i) = 1+min(d_x, d_y)+(d/2)[(d/2)+(d\%2)-1]$, where $d_x := d(i,x)$, $d_y := d(i,y)$, $d := d_x + d_y$, $(d/2)$ and $(d\%2)$ are the integer quotient and remainder of d divided by 2, respectively. In addition, perfect hashing allows fast closed-form computations. For nodes with $d(i,x) = 1$ or $d(i,y) = 1$, a null label 0 is assigned. After the labels are obtained, one-hot encoding vectors are used to construct X.

3.4 Other Heuristic Link Prediction Methods

In this section, we introduce some other heuristic methods for link prediction that will be used as a benchmark for the SEAL model. As mentioned earlier, the Katz index, SimRank, and rooted PageRank excel at link prediction when only one or two hops are involved between the target vertices. However, when used on graphs where more than 2-hop neighbours are likely to form connections, heuristic scoring methods, in addition to the ones introduced earlier, start to lose sight of the structure of the graph. We use these methods in particular as benchmarks against SEAL so that a basis can be formed on earlier research that also uses these methods as benchmarks [40]. In the subsections of this section, we briefly introduce each of these methods.

3.4.1 Katz Index

The Katz index [11] is a path-ensemble-based method based on the intuition that the more paths there are between two nodes, and the shorter these paths are, the stronger the relationship between the two nodes, and the more likely they are to discover and interact with each other. It sums over all paths existing between pairs of nodes [28, 42].

Before introducing the Katz index, we must first define some terminology. Firstly, a *walk* is a succession of vertices $x, \ldots, y$ in which each pair of vertices (x, y) is connected by an edge. Secondly, a *multigraph* is a graph where any two vertices can have multiple edges between one another. From this, we can say that different links between two nodes can also be composed with different walks. That is, the set of lengths of all possible walks between x and y in a multigraph is equal to $\prod_{i=1}^{m-1} N(x, y)$, in which $N(x, y)$ is the number of links between x and y. Then, the Katz index is defined as follows:

$$katz_{x,y} = \sum_{l=1}^{\infty} \beta^l |walks^{(l)}(x, y)| = \sum_{l=1}^{\infty} \beta^l [A^l]_{x,y} \tag{7}$$

In Equation 7, $walk^{(l)}(x, y)$ is the set of walks of length l between x and y, and A^l is the power of the adjacency matrix of the network, and β is a parameter $\in (0, 1)$ that represents the probability of effectiveness of a single link [34]. The Katz index adds over the collection of all walks between x and y, where a walk of length l is restricted by β^l. A higher exponent, and therefore, a longer walk length, results in a smaller value for β^l, so nodes that are farther away from each other will have less of an impact on the index [16].

3.4.2 SimRank

SimRank [10] is a metric for assessing the similarity between two objects based on the degree to which they are linked to similar objects. In SimRank, much like the unweighted and edge-weighted heuristic measures named before, each pair of vertices (x, y) is given a score, or "similarity", $s(x, y)$ which is the average similarity between the in-neighbours of x and y, or the sets $I(x)$ and $I(y)$ respectively. In addition, we define $I_i(x)$ to be the ith in-neighbour in $I(x)$, where $1 < i < |I(x)|$. Then, the SimRank score $s(x, y)$ for a given pair of vertices (x, y) as:

$$s(x, y) = \frac{C}{|I(x)||I(y)|} \sum_{i=1}^{|I(x)|} \sum_{j=1}^{|I(y)|} s(I_i(x), I_j(y)) \tag{8}$$

In Equation 8, C is a constant parameter $\in (0, 1)$. If x and y do not have any in-neighbours, we cannot come to a conclusion that any similarity exists between x and y, since $I(x) =$ or $I(y) = \{\}$, and in this case, $s(x, y) = 0$.

3.4.3 Rooted PageRank

Rooted PageRank [14] is a customized version of PageRank [7] that uses "random walks", or walks with randomly chosen sequences of steps, on a graph to calculate the probability that two vertices will gain an edge between each other in the future. For a pair of vertices (x, y), and given a parameter $\beta \in (0, 1)$, the rooted PageRank value $RPR(x, y)$ is defined as the stationary probability of y under a random walk with probability $1 - \beta_{RPR}$ of jumping to vertex x, and probability β_{RPR} of moving to a random neighbour of the current vertex. Furthermore, let D be a diagonal matrix with $D|i, i| = \sum_j A|i, j|$, and let $T = D^{-1}A$ be the adjacency matrix with the summation of the rows normalized to 1. Then, to compute the rooted PageRank matrix of an entire graph, we only need to compute the matrix inverse $(I - \beta_{RPR} \cdot T)^{-1}$. As a whole, the rooted PageRank of a graph can be calculated by:

$$RPR = (1 - \beta_{RPR})(I - \beta_{RPR} \cdot T)^{-1} \tag{9}$$

From this, we can also find the standard PageRanks of a graph by averaging of all columns of the rooted PageRank matrix.

4 Methodology: Vertex-and-Edge-Weighted Heuristic Link Prediction Methods

In this chapter, we propose new heuristic methods which take the weights of vertices as well as edges into account. These measures are based on Murata and Moriyasu's [20] edge-weighted methods. For each of these methods, we define $w'(u, v)$ as a common weight factor that combines edge and vertex weights for two vertices u and v, as follows:

$$w'(u, v) = w(u) \cdot w(v) \cdot w(u, v) \tag{10}$$

This definition of $w'(u, v)$ follows the principle of preferential attachment [3] that was mentioned earlier in the introduction of the PA measure. We hypothesize that hashtag graphs follow the principle of preferential attachment based on two intuitions. Firstly, hashtags which have seen more frequent use in the past will be used more in the future in comparison to those which have not seen as much use. Secondly, if a tweet featuring a relatively popular hashtag and a relatively unknown

hashtag is posted, the unknown hashtag may enjoy increased exposure due to the exposure given to the tweet itself by the inclusion of the popular hashtag [6].

With this definition of $w'(x, y)$, the resulting vertex-and-edge-weighted methods become similar in appearance to Murata and Moriyasu's [20] edge-weighted methods. However, we eliminate the 2 in the denominators of the edge-and-vertex-weighted versions of the Common Neighbours and Adamic/Adar methods, in hopes of reducing the computational load for each. Since it is a common factor, removing it will not change the relative ordering by score of each pair of vertices in the graph. We define, as follows, vertex-and-edge-weighted versions of the CN (WN, Equation 11), AA (WA, Equation 12), and PA (WP, Equation 13) methods, respectively:

$$S_{WN}(x, y) = \sum_{z \in \Gamma(x) \cap \Gamma(y)} w'(x, z) + w'(y, z) \tag{11}$$

$$S_{WA}(x, y) = \sum_{z \in \Gamma(x) \cap \Gamma(y)} \frac{w'(x, z) + w'(y, z)}{\log \sum_{z' \epsilon \Gamma(z)} w'(z', z)} \tag{12}$$

$$S_{WP}(x, y) = \sum_{x' \cap \Gamma(x)} w'(x', x) \times \sum_{y' \cap \Gamma(y)} w'(y', y) \tag{13}$$

As before, we demonstrate these methods by applying them to an example graph. Here, we will use vertices x and y in Fig. 3, which uses the same structure and edge weights as Fig. 2 while featuring vertex weights assigned to its vertices as well. Firstly, for the vertex-and-edge-weighted Common Neighbours method, as defined in Equation 11, the vertex weights on x, y and each common neighbour all come into play. Since the edge $\{x, z_1\}$ has a greater weight than $\{y, z_1\}$, it has a larger impact on the overall value of $S_{WN}(x, y)$. However, z_2 has a larger weight than z_1, which also amplifies the effect its connections have on the score. Here, the overall score becomes $(1 \times 1 \times 2 + 1 \times 1 \times 1) + (1 \times 2 \times 1 + 1 \times 2 \times 1 = 7$. Meanwhile, for the vertex-and-edge-weighted Adamic/Adar method in Equation 12, the role z_2 has in adding to the overall score is diminished similarly to the exclusively edge-weighted version. In this case, the z_1 term becomes $\frac{1 \times 1 \times 2 + 1 \times 1 \times 1}{\log[1 \times 1 \times 2 + 1 \times 1 \times 1]}$, which evaluates to roughly 6.29. The z_2 term becomes $\frac{1 \times 2 \times 1 + 1 \times 2 \times 1}{\log[2 \times 1 \times 1 + 2 \times 1 \times 1 + 2 \times 1 \times 3]} = 4$, making the overall value of $S_{WA}(x, y)$ approximately 6.29. Finally, for the vertex-and-edge-weighted Preferential Attachment method defined in Equation 13, and x contributes a term of $1 \times 1 \times 2 + 1 \times 2 \times 1 + 1 \times 1 \times 1 = 5$, while y contributes a term of $1 \times 1 \times 1 + 1 \times 2 \times 1 + 1 \times 1 \times 1 + 1 \times 2 \times 2$. Once the two are multiplied together, $S_{WP}(x, y) = 5 \times 8 = 40$.

No additions are made to the SEAL method; instead, we apply it in its existing form to hashtag graphs to judge its performance against our benchmark heuristic methods (Katz index, SimRank, and rooted PageRank).

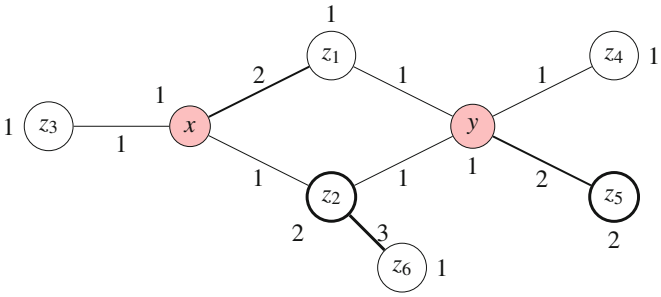


Fig. 3 An example of a vertex-and-edge-weighted graph. The structure is identical to 2, but with vertex weights added. The vertex-and-edge-weighted versions of the Common Neighbours, Adamic/Adar, and Preferential Attachment methods, as outlined in Equations 11, 12 and 13, respectively, are being used to calculate $score(x, y)$. Thicker outlines are used to indicate weights greater than 1

5 Experimental Setup

5.1 Data Collection

To evaluate the link prediction methods introduced in Sects. 2 and 4, we used 3 Twitter datasets described in Table 1. Where search terms are given, a string in double-quotes (") denotes a search including tweets containing a literal string, a string beginning with an at symbol (@) denotes a search including tweets from a certain account with the name after the at symbol, and a string beginning with a pound sign (#) denotes a search including tweets containing a hashtag. All searches are inclusive, meaning that a tweet need only fit one of the search terms to be collected in the dataset.

Table 1 Datasets used in our experiment. For each dataset, "Tweets" is the number of unique tweets contained in the dataset, "Hashtag Occurrences" is the number of times any hashtag appeared in any tweet, "Hashtag Co-Occurrences" is the number of times any two hashtags appeared together in the same tweet, "Hashtag/Tweet" ratio is the number of hashtag occurences divided by number of unique tweets and "Unique Hashtags" is the number of unique hashtags

Dataset	`worldcup`	`oscars`	`cities`
Tweets	931,894	1,326,545	719,974
Hashtag occurrences	10,833,196	1,967,287	1,778,554
Hashtag/tweet ratio	11.62	1.48	2.47
Hashtag co-occurrences	12,887,787	1,171,292	2,594,847
Unique hashtags	112,109	40,295	100,507

The `worldcup` dataset consists of tweets surrounding the topic of the 2018 FIFA World Cup, gathered between June 30, 2018 and July 17, 2018 with the aid of

the TwitterSearch package[2] for Python.[3] The search terms used when collecting the data were:

- #worldcup
- #worldcup2018
- #worldcup18
- @FIFAcom
- @FIFAWorldCup
- "world cup"

The `oscars` dataset consists of tweets surrounding the 91st Academy Awards, gathered between February 17, 2019 and February 27, 2019 by the Social Data Repository at Lakehead University's DATALab [18, 23]. The search terms used when collecting the data were:

- oscars
- "academy awards"

The `cities` dataset consists of tweets surrounding the topics of some of Canada's major metropolitan centres, gathered between February 17, 2019 and March 14, 2019, also by the Social Data Repository [18]. The search terms used when collecting the data were:

- "toronto"
- "montreal"
- "calgary"
- "ottawa"
- "edmonton"
- "winnipeg"
- "vancouver"

5.2 Data Pre-processing

We split each dataset into four time-based quartiles, each containing tweets from a period of time encompassing a quarter of the entire time over which the data was collected. From these splits, four graphs were constructed. The Q1 graph encompasses only tweets from the first quartile, the Q2 graph encompasses tweets from the first and second quartiles, and the Q3 graph encompasses tweets from the first, second and third quartiles. All of these graphs will be used for prediction. Meanwhile, the total graph encompasses all tweets from all four time-based quartiles, and though link prediction will not be conducted on it, it is used to evaluate the predictions made on the Q1, Q2 and Q3 graphs. In each graph, a vertex is created for each unique hashtag and weighted with the number of tweets in which that hashtag appeared, and an edge is created for each pair of hashtags that appeared together in the same tweet, and weighted with the number of tweets those two hashtags appeared together in. The numbers of vertices and edges in each quartile's

[2] https://pypi.org/project/TwitterSearch/

[3] https://docs.python.org/3.5/

Table 2 The numbers of vertices and edges present in the graphs for each dataset at each of the four time-based quartiles

Datatset		worldcup	oscars	cities
Q1	Vertices	37,918	5,343	41,146
	Edges	205,185	14,674	162,888
Q2	Vertices	65.730	14,460	62,223
	Edges	375,012	47,314	266,742
Q3	Vertices	92,384	37,439	83,976
	Edges	533.533	149,462	379,982
Total	Vertices	112,109	40,295	100,507
	Edges	658.489	164,895	471,531

Table 3 The number of edges present in the graphs constructed from only the top 200 hashtags in each dataset or each of the four time-based quartiles

Dataset	worldcup	oscars	cities
Q1	5,896	790	1,960
Q2	8,239	1,557	2,386
Q3	v9,856	3,327	2,746
Total	10,567	3,320	2960

graph for each of our datasets are given in Table 2. Due to dataset size, as well as the consequent computational costs of running the heuristic scoring methods over the entire datasets, we follow the lead of Martinčić-Ipšić et al. [17] and only consider the top 200 most used hashtags from each dataset, regardless of the time of usage. The numbers of edges in each quartile's graph for each dataset when only the top 200 hashtags are considered are given in Table 3.

From each of the Q1, Q2 and Q3 graphs, a score graph is generated by applying each of the heuristic scoring methods mentioned above: the unweighted, edge-weighted, and vertex-and-edge-weighted versions of the Common Neighbours, Adamic/Adar and Preferential Attachment methods. In each case, the score graph is a complete graph where each vertex is unweighted and each edge is weighted with the score resulting from the pair of vertices connected by the edge. Then, each score graph for Q1, Q2 and Q3 is evaluated against the total graph and given a single numerical score by each of the three evaluation metrics mentioned above: precision, F_1 score and AUC. In each case, the Q1, Q2 or Q3 graph is said to contain E_T as its set of edges, while the total graph is said to contain E_P as its set of edges. Vertices that appear in the total graph, but not the graph being evaluated, are ignored, as the link prediction measures have no way of predicting when and where new vertices will appear. Storage of datasets and graphs constructed from those datasets was accomplished with MySQL,[4] while link prediction and evaluation were implemented in the Java[5] programming language.

[4] https://dev.mysql.com/doc/refman/5.7/en

[5] https://docs.oracle.com/javase/8/docs/api/overview-summary.html

6 Results

In this section we use a number of existing algorithms to evaluate the performance of scoring methods. We define a training set E_T and a test set E_P from a graph of collected hashtags $G = \langle V, E \rangle$, where E_T contains only edges formed by co-occurrences of hashtags in tweets sent before a given time T. E_P contains only edges formed by co-occurrences of hashtags in tweets sent after T, not counting edges that were already formed in E_T; therefore, $E_T \cap E_P = \emptyset$. Finally, the set of non-existent edges E_N is taken to contain all possible edges given V, except for those that exist in E; therefore, $E \cup E_N$ is the set of edges for the complete graph $K_{|V|}$.

The precision score [14] is the simplest of the evaluation metrics we use given in Equation 14. It is defined as the ratio of true positives to both true and false positives, as such:

$$P = \frac{|TP|}{|TP| + |FP|} \tag{14}$$

where $|TP|$ is the number of true positives, or the number of edges that are both in the set of the top $|E_P|$ possible vertices and in E_P, and $|FP|$ the number of false positives, or the number of edges in the set of the top $|E_P|$ possible vertices, but absent from E_P.

The F_1 score extends upon precision, and can in fact be defined as the harmonic mean of precision and recall [30] given in Equation 15. In the case of link prediction [17], F_1 is defined as follows:

$$F_1 = \frac{2|TP|}{2|TP| + |FP| + |FN|} \tag{15}$$

where $|FN|$ is the number of false negatives, or the number of edges in E_P that are absent from the set of the top $|E_P|$ possible vertices.

Area under the receiver operating curve (AUC) [8], the last of our evaluation metrics, determines the probability that a classifier will rank a randomly chosen positive instance higher than a randomly chosen negative instance. It describes the performance of discrete classifiers in terms of the rate of true positives to the rate of false positives. While a receiver operating curve is a two-dimensional curve, the AUC score reduces this to a single scalar value.

We follow the approach to calculating AUC taken by Lu and Zhou [15]. For $n = |E_P| \times |E_N|$ independent comparisons between all possible combinations of edges $\{x_P, y_P\} \in E_P$ and $\{x_N, y_N\} \in E_N$, AUC is given by

$$AUC = \frac{n' + 0.5n''}{n} \tag{16}$$

where n' is the number of comparisons in which $S(x_P, y_P) > S(x_N, y_X)$, and n'' is the number of comparisons in which $S(x_P, y_P) = S(x_N, y_N)$.

6.1 Heuristic Link Prediction Methods

The link prediction for datasets `worldcup`, `oscars` and `cities` are presented in Figs. 4, 5 and 6, respectively. A trend is present across all three datasets in which the precision and F_1 scores for each heuristic scoring method decrease over time from Q1 to Q3 as more links are added. This is in line with the trend observed by Martinčić-Ipšić et al. [17] from their hashtag graphs constructed from exactly 25%, 50% and 75% of all tweets. As more data is included, more of the highly probable links are already included in the resulting graph, making the task of prediction more difficult. Also similarly to their experiment, our results show that AUC scores do not seem to be prone to the same effect. In fact, we note that while AUC scores show no certain trend over time in the `worldcup` and `cities` datasets, they increase over time for the unweighted class of heuristic scoring methods on the `oscars` dataset. This indicates that while the task of predicting the specific set of correct links increases in difficulty as time goes on, the task of ranking edges in E_P above those in E_N in general does not.

We also note that, in all datasets, the precision and F_1 scores rise and fall with each other across different quartiles and heuristic scoring methods. This means that if the scoring methods themselves were to be ranked, the ranks would not differ between the precision and F_1 scores. Because of this, and because the F_1 score already accounts for precision, we recommend following the advice of Martinčić-Ipšić et al. [17] and using the F_1 and AUC scores in future research into link prediction on hashtag graphs.

Out of the unweighted, edge-weighted and vertex-and-edge-weighted classes of heuristic scoring methods, the unweighted methods exhibit the best performance across all three evaluation scores, while the edge-weighted methods perform only slightly worse than the unweighted methods, and the vertex-and-edge-weighted methods perform, in most cases, slightly worse than the edge-weighted methods. This runs contrary to Murata and Moriyasu's [20] conclusion, and likely points to there being some property that differs between the hashtag graphs examined in our work, and the social graphs constructed from question-answering bulletin board data that were examined in theirs. Given the manner in which weights were taken into account in each scoring method, versions of the CN, AA and PA methods that "punish" pairs of vertices for having greater weights may exhibit greater performance than their unweighted counterparts, and are worthy of future study.

Of the 3 data sets, the common feature was that all methods had much greater predictive power from Q1 and Q2 than in Q3. This is as expected because the denser the graph is, and the more edges it has between vertices, the harder it becomes

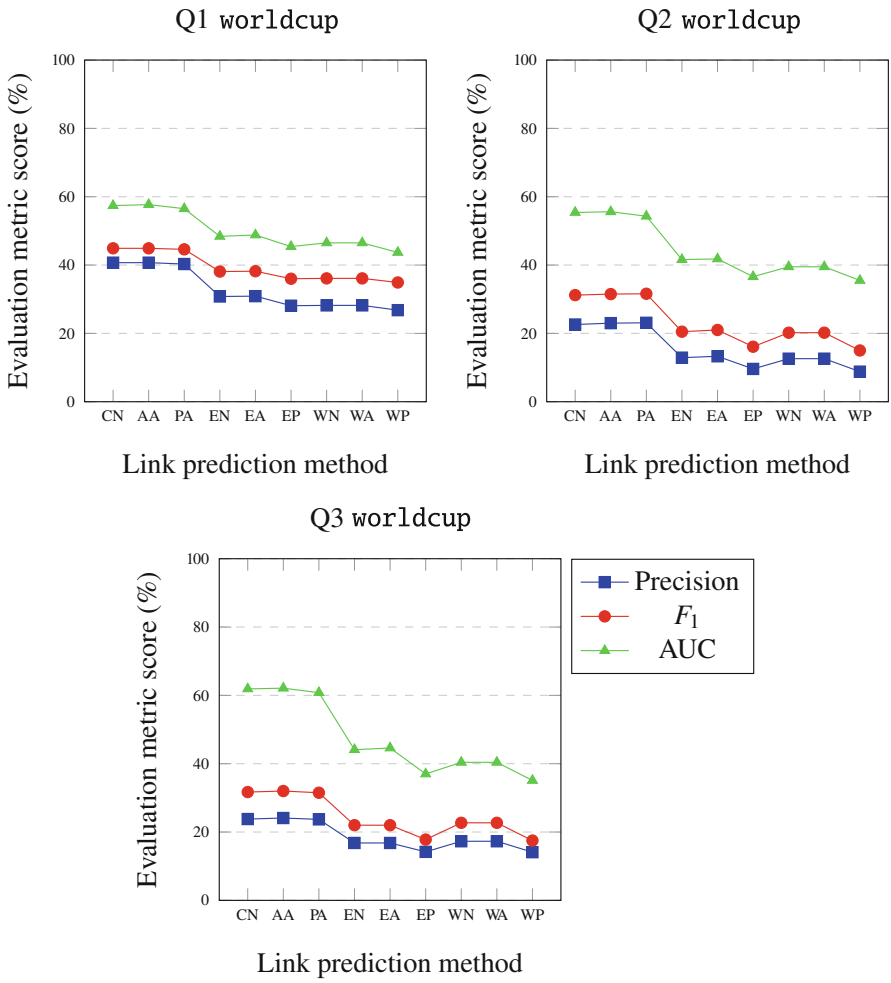


Fig. 4 Link prediction in the Q1, Q2 and Q3 graphs constructed from the `worldcup` dataset. Shown are the evaluation metric scores, namely the precision, F_1 and AUC scores, obtained from the predictions of the heuristic scoring methods, namely Common Neighbours (CN), Adamic/Adar (AA), and Preferential Attachment (PA), plus their edge-weighted (EN, EA, EP) and vertex-and-edge-weighted (WN, WA, WP) counterparts

to predict new edges not already present in the graph. Focusing on the Q1 and Q2 results for the 3 data sets, the `cities` data set performed significantly worse than the other 2. We feel this is explained by the nature of the datasets. With both `oscars` and `worldcup`, the tweets were centered around a singular event or topic. However, the `cities` dataset was about several Canadian cities, therefore tweets, topics, and definitely hashtags may not have any correlation with each other

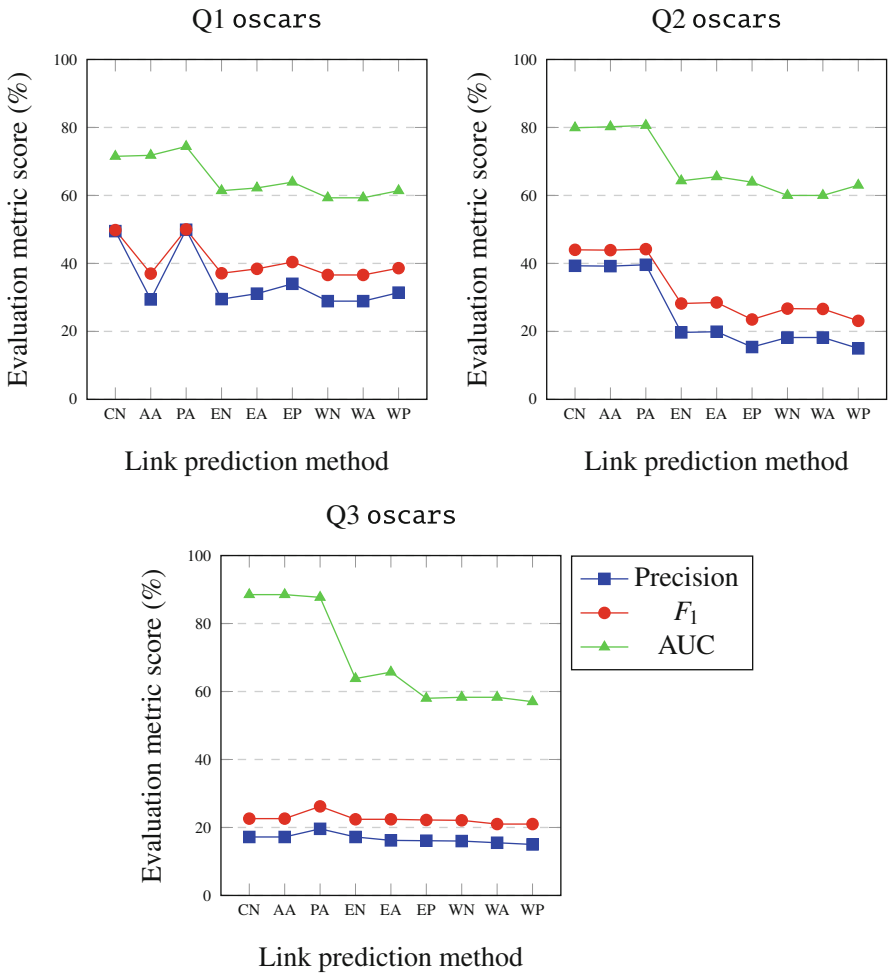


Fig. 5 Link prediction in the Q1, Q2 and Q3 graphs constructed from the `oscars` dataset. Shown are the same quantities as in Fig. 4

over the specified time frame. This leads us to believe that hashtag graphs are best constructed around a distinct topic, but this hypothesis is also worthy of future study.

When comparing the heuristic scoring methods in terms of their membership in the Common Neighbours, Adamic/Adar and Preferential Attachment classes, the Common Neighbours and Adamic/Adar methods with the same weight consideration (unweighted, edge-weighted, or vertex-and-edge-weighted) perform nearly identically across all three evaluation methods, with the Adamic/Adar methods having a slight edge in those cases where the two do differ. This confirms the results found by Murata and Moriyasu [20] and Lieben-Nowell and Kleinberg [14]. The Preferential Attachment methods, meanwhile, tend to exhibit worse performance

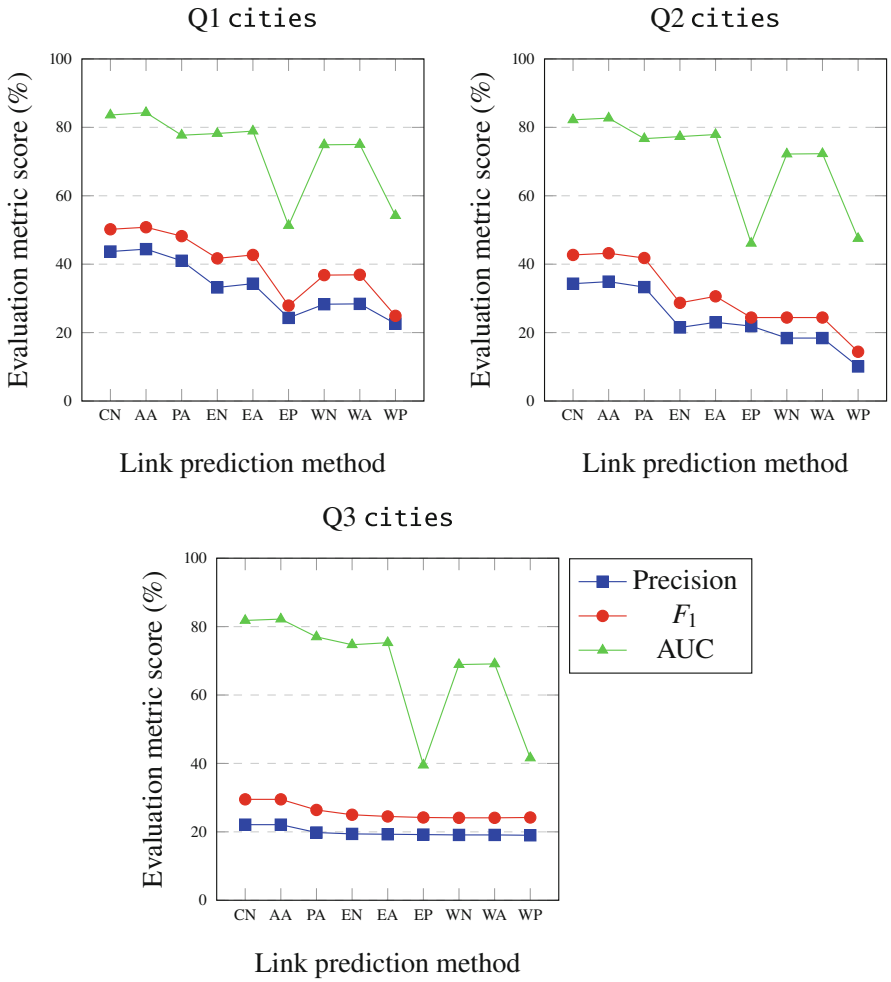


Fig. 6 Link prediction in the Q1, Q2 and Q3 graphs constructed from the `cities` dataset. Shown are the same quantities as in Fig. 4

than the others with the same weight consideration in most cases. This effect is not as pronounced when predicting from the Q1 graphs, with the `oscars` dataset even showing slightly better performance from the Preferential Attachment methods, but it makes itself more apparent when predicting from the Q2 and Q3 graphs in most cases. Since more possible edges in a hashtag graph are filled in as time goes by, the degree distribution of the graph becomes more uniform, and as Murata and Moriyasu noted [20] in their experiment, the strategy of preferential attachment becomes less appropriate in such cases.

Table 4 Link prediction in the `cities` dataset. Shown are the evaluation metric scores, namely AUC and precision

	Katz index	Rooted pagerank	SimRank	SEAL
AUC	0.908	0.919	0.789	0.959
Precision	0.917	0.917	0.720	0.921

6.2 SEAL

In order to evaluate the performance of SEAL against our chosen benchmark heuristic methods (Katz index, SimRank and rooted PageRank), we conduct our second set of experiments exclusively on the `cities` dataset. The choice of only working on the `cities` dataset is due to the above heuristic methods' significantly worse performance on that dataset compared to the `oscars` and `worldcup` datasets. Hence, our main focus was to find a better-performing method for that dataset and to examine whether SEAL can detect any hidden correlations between tweets, topics and hashtags in that dataset that a heuristic method cannot. SEAL is mainly used with Katz, SimRank, and PageRank to make a comparative analysis to these heuristic methods without employing them on all of our datasets, or using all of the methods explored previously. A recent architecture, Deep Graph Convolutional Neural Network (DGCNN) [41], was used as the default GNN for SEAL. We chose a hop value of 3, due to SEAL exhibiting optimal performance with that hop value, according to Zhang et al. [40]. Using AUC and precision as evaluation metrics [41], our results show that SEAL, with the aid of a GNN, performs better than the benchmark approaches for link prediction.

Table 4 shows the results of the experiment. Here, SimRank exhibits the lowest AUC and precision scores, being outclassed by both the Katz index and rooted PageRank methods. However, SEAL exceeds the other two heuristic measures slightly in precision scores, and significantly in AUC scores. This demonstrates that the improved link prediction abilities that SEAL exhibits over simpler, heuristic-based methods are also applicable to hashtag graphs. It could be said that SEAL, by learning from the hashtag graph, is able to develop a model that better reflects the graph's properties than the other, more traditional methods.

7 Conclusions and Future Research

In this work, we show that hashtag graphs can be used to map the communications on Twitter, in which the hashtags are represented in terms of vertices, and edges are equivalent to frequency of hashtags within the same tweet [12]. In addition, vertices in a hashtag graph are assigned with weights by the number of tweets a hashtag has occurred in, and edges are weighted with the number of tweets both hashtags have occurred in, creating a "weighted hashtag graph". Moreover,

in this chapter we have discussed some well established link prediction methods which has enabled the weights of both vertices and edges to be considered in a weighted hashtag graph [32]. We analyzed the performance of several unweighted, edge-weighted, and edge-and-vertex-weighted heuristic scoring methods, originally used for social graphs, on hashtag graphs constructed from Twitter data. We also compared the performance of the SEAL method against another, commonly-used set of traditional, heuristic-based methods as benchmarks.

Experiments were carried out on real-life Twitter datasets combining over 3,000,000 unique tweets and 250,000 unique hashtags. The results have demonstrated marginal performance with simpler heuristic-based scoring methods and the performance decreases when more data is added over the course of time. SEAL on the other hand has outperformed the other heuristic scoring methods in predicting links of the hashtag graphs. AUC score was used as one of the evaluation metrics. As AUC is used to compute the probability that whether a classifier will be ranking randomly a selected positive instance over a randomly chosen negative instance. AUC determines the performance in terms of the rate of true positives to the rate of false positives. Our experimental results have shown that as more data is added, the score decreases as the highly probable links are already existing in the resulting graph thus making the task of predicting links more challenging. Moreover, the AUC scores were unable to show any certain trends over the period of time in the `worldcup` and `cities` datasets. However, for the `oscars` datasets, the links have seemed to increase over the time for the unweighted class of the heuristic scoring methods. Thus, the task of predicting a specific set of correct links becomes difficult over the course of time. The unweighted methods have shown the best performance out of the unweighted, edge-weighted and vertex-and-edge-weighted classes of heuristic scoring methods. Meanwhile, the edge-weighted methods did not have a satisfactory performance compared to the unweighted methods. In most of the cases, the edge-weighted methods has out performed vertex-and-edge-weighted methods. Regarding the edge-and-vertex-weighted heuristic scoring methods, it was shown that the inclusion of weights worsened the performance of the three heuristic scoring methods that were trialed in all cases where the top 200 hashtags were used. This conclusion is contrary to that of earlier research by Murata and Moriyasu [20] regarding solely edge-weighted heuristic scoring methods. Although this result is interesting, we believe that including more than the top 200, or even all, of the hashtags in the graph could lead to better results across the board. Due to the difficulty of efficiently and reasonably quickly handling such a large amount of data, however, we leave further investigation into this result for future research. Moreover, Martinčić-Ipšić et al. [17] explored some other heuristic-based link prediction methods that were not covered in this work, but could be modified to include vertex weights in future research.

Precision and AUC scores of 92.1% and 95.9% were achieved using SEAL on the `cities` dataset. Our goals were to examine how the inclusion of the weights of edges, as well as the weights of both vertices and edges, from a hashtag graph in heuristic scoring methods affected link prediction performance, as well as to confirm the viability of GNN-based approaches to link prediction for hashtag graphs

in particular. We predict that the vast majority of practical applications of hashtag graph link prediction will be driven by SEAL or similar methods.

The structure of Twitter hashtag graphs could be further analyzed. Since much research has been done on social graphs of followers [2, 33] and collaborators [4, 21], they could be compared to hashtag graphs to search for any structural differences. In addition, social media platforms other than Twitter that also make use of hashtags have the potential to exhibit a slightly different structure as well, even if the networks they represent are fundamentally similar. This may aid in the development or refinement of heuristics to be used specifically for link prediction in hashtag graphs, or help in further refining GNN-based approaches such as SEAL for this domain.

Finally, there are opportunities for increasing the richness of data encoded in hashtag graphs from other data surrounding tweets, such as the time and date that a tweet was posted (and therefore, the time a hashtag usage occurred), as well as information about the account that posted a tweet (and therefore, used a hashtag in a tweet). SEAL can also be used to solve other Twitter hashtag graph issues such as knowledge graph completion and more effective recommender systems. Moreover, knowledge graphs can enhance several Twitter text analysis tasks such as word sense disambiguation (identifying which sense of a word is used in tweet), semantic search (to look for meaningful information and not based on lexical match) [26], and relation extraction (identify and categorize semantic relations between entities).

References

1. Adamic, L. A., & Adar, E. (2003). Friends and neighbors on the web. *Social Networks, 25*(3), 211–230.
2. Badami, M., & Nasraoui, O. (2018). Cross-domain hashtag recommendation and story revelation in social media. In *2018 IEEE International Conference on Big Data (Big Data)* (pp. 4294–4303). IEEE.
3. Barabási, A. L., & Albert, R. (1999). Emergence of scaling in random networks. *Science, 286*(5439), 509–512.
4. Barabâsi, A. L., Jeong, H., Néda, Z., Ravasz, E., Schubert, A., & Vicsek, T. (2002). Evolution of the social network of scientific collaborations. *Physica A: Statistical Mechanics and Its Applications, 311*(3–4), 590–614.
5. Bollen, J., Mao, H., & Zeng, X. (2011). Twitter mood predicts the stock market. *Journal of Computational Science, 2*(1), 1–8.
6. Bruna, J., Zaremba, W., Szlam, A., & LeCun, Y. (2013). Spectral networks and locally connected networks on graphs. arXiv preprint arXiv:1312.6203.
7. Chakrabarti, S. (2007). Dynamic personalized pagerank in entity-relation graphs. In *Proceedings of the 16th International Conference on World Wide Web* (pp. 571–580). ACM.
8. Fawcett, T. (2006). An introduction to roc analysis. *Pattern Recognition Letters, 27*(8), 861–874.
9. Grover, A., & Leskovec, J. (2016). node2vec: Scalable feature learning for networks. In *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining* (pp. 855–864).

10. Jeh, G., & Widom, J. (2002). 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). ACM.
11. Katz, L. (1953). A new status index derived from sociometric analysis. *Psychometrika, 18*(1), 39–43.
12. Koren, Y., Bell, R., & Volinsky, C. (2009). Matrix factorization techniques for recommender systems. *Computer, 42*(8), 30–37.
13. Kovács, I. A., Luck, K., Spirohn, K., Wang, Y., Pollis, C., Schlabach, S., Bian, W., Kim, D. K., Kishore, N., Hao, T., et al. (2019). Network-based prediction of protein interactions. *Nature Communications, 10*(1), 1240.
14. Liben-Nowell, D., & Kleinberg, J. (2007). The link-prediction problem for social networks. *Journal of the American Society for Information Science and Technology, 58*(7), 1019–1031.
15. Lü, L., & Zhou, T. (2011). Link prediction in complex networks: A survey. *Physica A: Statistical Mechanics and Its Applications, 390*(6), 1150–1170.
16. Luxburg, U. V., Radl, A., & Hein, M. (2010). Getting lost in space: Large sample analysis of the resistance distance. In *Advances in Neural Information Processing Systems* (pp. 2622–2630).
17. Martinčić-Ipšić, S., Močibob, E., & Perc, M. (2017). Link prediction on twitter. *PloS one, 12*(7), e0181079.
18. Mendhe, C. H., Henderson, N., Srivastava, G., & Mago, V. (2020). A scalable platform to collect, store, visualize, and analyze big data in real time. *IEEE Transactions on Computational Social Systems, 2020*, 1–10.
19. Monti, F., Bronstein, M., & Bresson, X. (2017). Geometric matrix completion with recurrent multi-graph neural networks. In *Advances in Neural Information Processing Systems* (pp. 3697–3707).
20. Murata, T., & Moriyasu, S. (2007). Link prediction of social networks based on weighted proximity measures. In *Proceedings of the IEEE/WIC/ACM International Conference on Web Intelligence* (pp. 85–88). IEEE Computer Society.
21. Newman, M. E. (2001). Clustering and preferential attachment in growing networks. *Physical Review E, 64*(2), 025102.
22. Nickel, M., Murphy, K., Tresp, V., & Gabrilovich, E. (2015). A review of relational machine learning for knowledge graphs. *Proceedings of the IEEE, 104*(1), 11–33.
23. Patel, K. D., Zainab, K., Heppner, A., Srivastava, G., & Mago, V. (2020). Using Twitter for diabetes community analysis. *Network Modeling Analysis in Health Informatics and Bioinformatics. 9*, 1–6.
24. Perozzi, B., Al-Rfou, R., & Skiena, S. (2014). Deepwalk: Online learning of social representations. In *Proceedings of the 20th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining* (pp. 701–710).
25. Praznik, L., Srivastava, G., Mendhe, C., & Mago, V. (2019). Vertex-weighted measures for predicting links in hashtag graphs (pp. 1–8).
26. Qudar, M., & Mago, V. (2020). A survey on language models. https://www.researchgate.net/publication/344158120_A_Survey_on_Language_Models.
27. Quercia, D., Askham, H., & Crowcroft, J. (2012). Tweetlda: Supervised topic classification and link prediction in twitter. In *Proceedings of the 4th Annual ACM Web Science Conference* (pp. 247–250). ACM.
28. Ribeiro, L. F., Saverese, P. H., & Figueiredo, D. R. (2017). struc2vec: Learning node representations from structural identity. In *Proceedings of the 23rd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining* (pp. 385–394).
29. Sandhu, M., Giabbanelli, P. J., & Mago, V. K. (2019). From social media to expert reports: The impact of source selection on automatically validating complex conceptual models of obesity. In *International Conference on Human-Computer Interaction* (pp. 434–452). Springer.
30. Sasaki, Y., et al. (2007). The truth of the f-measure. *Teach Tutor Mater, 1*(5), 1–5.
31. Sharma, G., Srivastava, G., & Mago, V. (2019). A framework for automatic categorization of social data into medical domains. *IEEE Transactions on Computational Social Systems, 7*(1), 129–140.

32. Shervashidze, N., Schweitzer, P., Van Leeuwen, E. J., Mehlhorn, K., & Borgwardt, K. M. (2011). Weisfeiler-lehman graph kernels. *Journal of Machine Learning Research, 12*(77), 2539–2561.
33. Sokolova, K., & Perez, C. (2018). Elections and the twitter community: The case of right-wing and left-wing primaries for the 2017 french presidential election. In *2018 IEEE/ACM International Conference on Advances in Social Networks Analysis and Mining (ASONAM)* (pp. 1021–1026). IEEE.
34. Tang, J., Qu, M., Wang, M., Zhang, M., Yan, J., & Mei, Q. (2015). Line: Large-scale information network embedding. In *Proceedings of the 24th international conference on world wide web* (pp. 1067–1077).
35. Tassone, J., Yan, P., Simpson, M., Mendhe, C., Mago, V., & Choudhury, S. (2020). Utilizing deep learning to identify drug use on twitter data. arXiv preprint arXiv:2003.11522.
36. Valverde-Rebaza, J., & de Andrade Lopes, A. (2013) Exploiting behaviors of communities of twitter users for link prediction. *Social Network Analysis and Mining, 3*(4), 1063–1074.
37. Wang, W., Wu, L., Huang, Y., Wang, H., & Zhu, R. (2019). Link prediction based on deep convolutional neural network. *Information, 10*(5), 172.
38. Wang, X., Wei, F., Liu, X., Zhou, M., & Zhang, M. (2011). Topic sentiment analysis in twitter: a graph-based hashtag sentiment classification approach. In *Proceedings of the 20th ACM International Conference on Information and Knowledge Management* (pp. 1031–1040). ACM.
39. Wang, Y., Liu, J., Huang, Y., & Feng, X. (2016). Using hashtag graph-based topic model to connect semantically-related words without co-occurrence in microblogs. *IEEE Transactions on Knowledge and Data Engineering, 28*(7), 1919–1933.
40. Zhang, M., & Chen, Y. (2018). Link prediction based on graph neural networks. In *Advances in Neural Information Processing Systems* (pp. 5165–5175).
41. Zhang, M., Cui, Z., Neumann, M., & Chen, Y. (2018). An end-to-end deep learning architecture for graph classification. In *Thirty-Second AAAI Conference on Artificial Intelligence*.
42. Zhao, H., Du, L., & Buntine, W. (2017). Leveraging node attributes for incomplete relational data. In *Proceedings of the 34th International Conference on Machine Learning-Volume 70* (pp. 4072–4081). JMLR. org.