Chapter 17
Emerging Research Trends and New Horizons

17.1 Introduction

The upcoming new horizons and recent research trends in Big Data Analytics frameworks, techniques and algorithms are as reflected in research papers recently published in conferences such as ACM International Conference on Knowledge Discovery and Data Mining (ACM SIG KDD), SIAM International Conference on Data Mining (SDM), IEEE International Conference on Data Engineering (ICDE) and ACM International Conference on Information and Knowledge Management (CIKM). In this chapter, we shall survey the research trends and the possible new horizons coming up in Big Data Analytics.

Current research areas include data mining, pattern recognition, natural language processing, business intelligence, collective intelligence, machine intelligence and web intelligence.

17.2 Data Mining

Data mining problems of interest include outlier detection, community detection, sequential pattern mining, network clustering, feature extraction, causal inference, parallel and distributed mining and predictive analytics. The problem of representing complex training data in a simple form suitable for statistical analysis permeates a host of applications, including network traffic and medical and biological data analysis, social web mining, e-Commerce, recommender systems and computational advertising.

C. S. R. Prabhu et al., *Big Data Analytics: Systems, Algorithms, Applications*,
https://doi.org/10.1007/978-981-15-0094-7_17

17.3 Data Streams, Dynamic Network Analysis and Adversarial Learning

The literature taking into account the management of uncertain data at scale focuses on the management of streaming and dynamic data, likely to be changing in time. In this chapter, we shall survey the state-of-the-art in adversarial machine learning and dynamic network analysis for Big Data samples [1–3].

Adversarial learning [4–7] accounts for changes to data by an intelligent adversary. Dynamic network analysis accounts for changes to data modeling real-world phenomena as complex networks in time. The dynamic data usually considered is either physical measurements or the World Wide Web data at different levels of granularity and complicity. With the emergence of Big Data systems in computer science, research into adversarial learning and dynamic networks borrow ideas from a wide spectrum of subjects such as mathematics, statistics, physics, computer science and social science. Thus, the research method evaluating the computational machine learning, data mining and data analytics models varies by research area. However, the common elements of Knowledge Discovery in Databases (KDD) process and algorithm design are seen in all the models. Here, the algorithmic design is concerned with reducing the computational complexity of implicit patterns found in the data. Such patterns are compressed in terms of computer programs with corresponding inputs and expected outputs. The program design is then specified in terms of approaches and improvements to existing algorithms and systems used in data analytics.

17.4 Algorithms for Big Data

The existing algorithms designed to handle very large volumes of data by indexing and search are not optimum for tasks requiring more precise and accurate analysis. In a static context, within time windows and tasks concern the study and analysis of incremental, heterogeneous data over search spaces issued from complex random processes.

17.5 Dynamic Data Streams

Dynamic data streams impose new challenges on the analytic methodologies since it is usually impossible to store an entire high-dimensional data stream or to scan it multiple times due to its tremendous volume and changing dynamics of the underlying data destruction over time.

17.6 Dynamic Network Analysis

Dynamic networks are an emerging topic of research in the area of graph mining. The edges and vertices of a graph can change over time. This is called a dynamic network which needs to be analyzed. The network can also be modeled as properties computed over one or more of weighted or un-weighted, directed or undirected; homogenous or heterogeneous networks found in the real world. Common sources and examples for dynamic networks are biological networks, Internet networks, bibliographic networks, email networks, health networks and road networks.

A variety of classical data mining techniques can be adapted for dynamic network analysis. Such techniques include classification, clustering, associations and inference under supervised and unsupervised learning categories.

Applications of dynamic network analysis can be found extensively in system diagnosis, financial markets, industrial surveillance, computer networks, space telemetry and information networks.

Research challenges in applying dynamic networks techniques include the algorithmic ability to deal with multidimensional data analysis, temporal data analysis, complex data type recognition and supervised classification learning.

17.7 Outlier Detection in Time-Evolving Networks

Outlier detection in the time-evolving networks, a special case of dynamic networks is a focus area in research. A data structure to summarize the changes in a time-evolving network is provided in [8]. In [9], an algorithm that evaluates the changes in a time-evolving network is proposed. In [8], stochastic guarantees on the performance of the data structure are presented by designing mathematical theorems on set properties and hash properties. In [9], a search algorithm is proposed which uses local optimization algorithm for searching dense blocks of tensors. The search criteria in the algorithm are defined in terms of certain statistics that are of interest in a real-world tensor. The input data may be represented as both a complex network and a complex tensor. Therefore, for dynamic network analysis, the graph search techniques based on optimization criteria become very much relevant. Both [8] and [9] deal in networks with a notion of change over time periods or ontologies or both. The accuracy of the data structure proposed in [8] depends on its ability to summarize the underlying data stream. For the definition of outlier given in terms of edge cuts in the graph, [8] provides approximate guarantees on the performance of the proposed data structure. The performance guarantees vary and depend upon sampling bias in reservoir sampling procedure and hashing procedure. The accuracy of algorithm prepared in [8] depends on the suitability of proposed statistical evaluation metrics for various complex networks. There, metrics may be readily applicable to data mining methods from tensor decompositions but not for data mining methods derived from

graph search. Thus, the reliability of statistics in [9] may be guaranteed only after significant amount of data preparation.

Adopting static network analysis techniques, static sequence analysis techniques and static set analysis techniques to a dynamic network is an emerging area of research. With the increasing emergence and availability of dynamic networks in application areas such as social networks or bio informatics networks, the relevance of techniques of mining and analysis of dynamic networks is only increasing gradually.

17.8 Research Challenges

The emerging research challenging includes the generalizing of the data mining methods for various data types combining structured and unstructured data, developing online versions of data mining methods and also formulating network analysis problems in the context of various application domains. The deployment of parallel and distributed data mining algorithms suitable for large changing datasets synthesized from multiple structured data sources may be useful in addressing these above-identified research problems. Examples of such data sources include Microsoft Academic Graph Network, Internet Movie Database, Wikimedia Commons, Web Data Commons, Apache Foundation Mail Archives and Git Hub Code Repositories.

17.9 Literature Review of Research in Dynamic Networks

We will now review the existing research literature on dynamic network analysis and adversarial learning. The papers reviewed in this section have been ordered by the ideas relevant for Knowledge Discovery in Databases (KDD) with an emphasis on dynamic network analysis and adversarial learning. KDD is a research methodology and thought process recommended for developing a data mining algorithm and solution. KDD has been formalized as the specification of Cross-Industry Standard Process for Data Mining (CRISP-DM). In its simplest form, KDD has three steps—preprocessing, modeling, post-processing—that must be addressed by any data mining method.

17.10 Dynamic Network Analysis

For dynamic network analysis, the properties or dynamics of the network to be modeled need to be defined. This is possible by sampling the dynamic network to be able to represent changes in the network. The changes are statistically quantified by certain validation metrics defined on the sample. The ideas of sampling, change detection, validation metrics affect the preprocessing and post-processing phases of

the KDD process. Once a suitable sample is available, we need to define the data mining model to be built on the current sample that is then validated on future samples. The elements of the data mining model include the type of the input complex network (such as a labeled network) and the data mining problem (such as evolutionary clustering and block modeling) that is being solved. The proposed solution of the data mining problem is then packaged and presented for real-world usage (through event mining, for instance). Thus, this literature review progresses along the lines of preprocessing with sampling and validation metrics, post-processing for change detection and event mining over labeled graphs that is then modeled as the data mining problems of evolutionary clustering and block modeling.

We conclude the literature review with a discussion on the existing data mining algorithms suitable for dynamic network analysis by summarizing the research gaps discussed in survey papers and books.

17.11 Sampling [8]

In [8], the authors discuss a sampling procedure to compress structural summaries underlying data stream of networks. The data used in the paper is from co-authorship networks and social media networks. The sampling is a modification of the well-known reservoir sampling procedure. In compressing the data, outlier detection is defined as identifying graph objects which contain unusual bridging edges.

This definition of outlier detection generalizes to unusual connectivity structure among different nodes found with respect to the historical connectivity structure. The algorithm maintains the information about the connected components dynamically during stream processing. The connected components are dynamically tracked by using the spanning forests of each of the edge samples.

Each outlier is assumed to satisfy a general condition on the structural criteria referred to as set monotonicity. Set monotonicity is determined as stochastic stopping criteria by examining the behavior of the edges in the individual graph objects. Set monotonicity is suitable for processing stream objects that can be examined at most once during the computation. The algorithm not only dynamically maintains node partitioning for the graph stream but also maintains a statistical model for outlier determination. Reservoir sampling is used to create data partitions that are then sorted by hashing with fixed random hash function. A current hash threshold is maintained to make decisions on whether or not incoming elements are to be included in the reservoir. The algorithm processes the edges in the reservoir in decreasing order of the hash function value until we are satisfied that the resulting reservoir is the smallest possible set which satisfies the stopping constraint. In the algorithm, current sample is the only set we need for any future decisions about reservoir sample maintenance.

The output partitions represent densely connected nodes with a small number of bridge edges. A cross-validation approach is used to model the likelihood statistics in a more robust way that avoids over fitting of the likelihood statistics to the particular structure induced by a reservoir sample. The paper presents first known real-time

and dynamic method for outlier detection in graph streams with structural statistics. The problems addressed in the paper are comparable to distance-based or density-based methods for outlier detection in multidimensional data. Adaptations of such research work are suitable for Big Data structures constructed around subgraphs. A subgraph is stated to be more suitable for Big Data mining. While it may not be possible to construct a statistically robust model on the basis of edge presence or absence between individual nodes, it may be possible to create a statistically more robust model for groups of nodes which correspond to the partitions. The paper's focus is on graph partitioning and reservoir sampling by estimating moments of data stream. Both partition sizes and interaction sizes are considered for estimating moments. Thus, possible extensions to the paper are in the direction of sampling with ensemble methods and estimation of higher-order moments. Furthermore, the paper uses only co-authorship networks. The methods in paper can be extended toward various kinds of heterogeneous networks, by and dynamic networks considering semantic features also in addition to other features.

17.12 Validation Metrics [9]

The authors in [9] discuss a tensor search algorithm applied to pattern discovery applications. The tensor is constructed with an intention of modeling suspicious behavior in twitter networks. The search algorithm finds dense blocks across modes and dimensions of the tensor. The tensor is used to represent high-dimensional features in the original dataset. The outlier detection algorithm then computes dense subgraphs or labeled subsets of tensor. The output of algorithm is used for anomaly or fraud detection. The search algorithm does a greedy search of the tensor by parameterizing the tensor dimensions in terms of validation metrics defined on subgraph densities. The notion of density is defined in terms of mass and sparsity of the tensor dimensions. The subgraph density is assumed to be generated from an underlying probability model generating a random graph. Specifically, the Erdos-Renyi model is assumed to be the ideal probability density function for tensor density. Then discrete data distribution actually found in the tensor is measured against the ideal probability density function using KL divergence. The local greedy search method estimating density is an alternating maximization procedure. The procedure searches each mode of the tensor for changes while assuming the remaining modes in tensor are fixed. The dynamics of the network are measured by the output of the search method on both un-weighted and weighted graphs.

The statistics proposed in the paper are useful for feature extraction, evolutionary clustering, event mining on labeled graphs and heterogeneous networks. The statistics are also useful for parallel data mining on the cloud provided issues in parallelizing the sparse matrix representations in search algorithm are well understood. The tensor search method in the paper is comparable to tensor decomposition methods where high-order singular values represent the importance of the dense block found in the

tensor. The statistical metrics of mass and size proposed in the paper are related to notion of singular values in tensor decomposition.

However, all the statistics in the paper put together generalize low-rank matrices found in tensor decompositions. The method proposed in paper is suitable for Big Data algorithms. The algorithm is suitable for dynamic, big, distributed processing of streaming data. The algorithm can be extended by constrained optimization and multiobjective optimization methods. To apply the method to a particular domain of application, we need to define the features, patterns and labels in the dataset that can act as constraints on static and streaming data. The alternating maximization procedure is comparable to coordinate descent in optimization methods. That is why, the proposed local search can be considered to be one step in the more general expectation–maximization procedure for searching tensors. To evaluate various block modeling methods with the proposed statistics, alternatives to KL divergence can be studied to measure distance between probability distributions.

17.13 Change Detection [10]

The authors of [10] propose a graph compression algorithm on a series of graphs where each graph is represented as an adjacency matrix. The algorithm takes into account both the community structures, as well as their change points in time, in order to achieve a concise description of the data. Standard benchmark graph data is used to evaluate the algorithm. The compression algorithm is derived from the Maximum Description Length (MDL) principle. Intuitively, the encoding objective tries to decompose graph into subgraphs or cliques that are either fully connected or fully disconnected. Specifically, MDL is used to define the objective function quantifying the cost of encoding an adjacency matrix as a binary matrix compression problem. The changes across compressed graphs are summarized by a cross-association method. The cross-association method incrementally summarizes tensor streams (or high-order graph streams) as smaller core tensor streams and projection matrices. The compression algorithm is a lossy compression that results in loss of information while capturing changes across graphs. The fundamental trade-off is between the number of bits needed to describe the communities and the number of bits needed to describe the individual edges in the stream. This algorithm belongs to the class of graph compression algorithms that use community discovery and optimization algorithms to compress the changes in graphs. Thus, the main challenge in the algorithm is to define objective function over a series of graphs and select corresponding objective function, search method in a parametric model. The objective function estimates the number of bits needed to encode the full graph stream (partition, community and edge) so far in space and time.

An alternating minimization method coupled with an incremental segmentation process performs the overall search over space and time. Such algorithms can be adapted into parallel, distributed algorithms over complex, dynamic networks. Such

an adaptation would have to be benchmarked for the often competing objectives—accuracy, speed and scale. If the compression algorithm defines the compression objective in terms of both rows and columns of the adjacency matrix, the algorithm can be adapted for information-theoretic correlation and biclustering-driven compression. Thus, various methods for grouping communities and time segments constitute future work in the paper. A data stream management system can also aid with data retrieval in the algorithm.

17.14 Labeled Graphs [11]

In [11], the authors discuss structural anomaly detection using numeric features. A TCP/IP access control database and a ATM bank transaction database are used as the input data. The anomalous security patterns investigated by the algorithm include temporal patterns, repetitive access patterns, displacement patterns and out-of-sequence patterns. The distribution of edge weights in a graph is used to compute anomaly scores. The edge weights, in turn, are computed from a structural function defined on the numeric labels on vertex or edge in a labeled graph. An anomaly is defined with reference to infrequent substructures in the graph. The infrequent substructures are computed by matching against frequent subgraphs. The anomaly scores are computed as parametric statistics from a Gaussian mixture model. In general, the anomaly scores can be computed by various data mining models defined in the context of various graph structures. The models can be determined by tradeoffs in accuracy and efficiency of the algorithm. The proposed data mining model is useful for forensic analysis of graph transaction databases and online detection of anomalies using dynamic graphs. The proposed algorithm is extensible to dynamic graphs and Big Data. For extending the algorithm, a variety of feature extraction techniques in general and unsupervised discretization methods, in particular, can be defined for vertex and edge attributes on the graph. Both parametric and non-parametric anomaly scoring models can also be investigated as an extension to the algorithm. As an extension to the model, a distance and density-based clustering can be used to snapshot clustering quality, social stability and sociability of objects computed from the anomaly scores.

17.15 Event Mining [12]

In [12], the authors define community dynamics and event mining in a time-evolving network. Events are summarized in terms of the heaviest dynamic subgraphs computed over a time sequence. The heaviest dynamic subgraph is supposed to indicate the behavior of an edge among neighbors in a graph. A search heuristic is used to compute local and global anomaly scores for an edge. In the search heuristic, edge behavior conditioned by neighborhood is generalized to a heavy subgraph. Then

heavy subgraph behavior is studied over a time sequence. The computed network can be correlated with external events for semantic event mining in time-evolving networks. By mining the full history of a large road network, a typical application of such event mining would compute a comprehensive summary of all unusual traffic congestions and their time of occurrence, and report it to the police or to urban planners.

Community dynamics and evolution in event mining is an active area of research. Naive adaptations of existing methods for dynamic network analysis are inefficient on large problem instances. So, future work would involve designing efficient ways to summarize the heaviest dynamic subgraphs. In such a subgraph, significant anomalous regions would involve very large-scale neighborhood search for both the heaviest subgraph and the maximum score subsequence. Thus, data mining issues include the need to reduce network scans and generate effective initial solutions (seeds) for the search heuristic to converge onto a globally optimum solution. Rather than a minimum spanning tree, an arbitrary shaped subgraph may also be defined to summarize the current graph in search algorithm.

17.16 Evolutionary Clustering

The authors of [13] propose an evolutionary bibliographic network summarization algorithm. The algorithm does iterative ranking and agglomerative evolutionary clustering according to an underlying probabilistic generative model. In the probabilistic generative model, an expectation–maximization estimates prior probabilities while maximum likelihood algorithm estimates posterior probabilities. A heterogeneous bibliographic information network is input to the algorithm where vertices correspond to different entities in the information network. For example, in DBLP Graphs the vertices are taken to be author, conference, paper and term. In general, the approach is suitable for star schema-based diagnosis of heterogeneous information networks. To introduce a temporal smoothness approach into the graph analysis, the core idea in proposed approach toward agglomerative evolutionary clustering is to cluster the entire entity or group of related objects as a whole, rather than clustering of individual types separately.

Such evolutionary clustering is combined with careful evolutionary diagnosis and metrics to determine merges and splits of different topical areas, authorship evolution and topical evolution. The new vector space model proposed in the algorithm can be leveraged for similarity computation and object assignment. Algorithmic tradeoffs exist between clustering quality and clustering consistency across time. The paper summarizes a sequence of graphs as a sequence of trees such that trees represent high-quality clusters. The maximum likelihood-based tree estimation model is only suitable for log-linear probabilities found in the dataset. This approach can be extended to nonlinear estimation of probabilities using probabilistic graphical models. Specifically, conditional random fields and dynamic Bayesian networks are well suited for mixing multiple probability generation mechanisms. The cluster evolution

metrics may also be adapted for quantifying the appearance and disappearance rate of objects across different time granularities. Also, to be adapted for complex network analysis, the approach needs to incorporate variable number of clusters across different time periods. This would be possible by diagnosing of features in the clusters across time. Looking into the clustering methods for data streams is a direction for improving the proposed algorithm. Commonly used clustering methods include partitioning clustering, hierarchical clustering, model-driven clustering, biclustering and multilevel clustering. Another possible extension is to integrate the evolutionary clustering with research into mixed membership stochastic block models used for iterative search and optimization.

17.17 Block Modeling [14]

Block modeling methods are a generalization of community detection and graph clustering methods. Reference [14] gives a block modeling method suitable for vertex, edge, subgraph data distributions found in a temporal graph. The block modeling is done with respect to validation metrics on weighted graphs as well as un-weighted graphs. The metrics are defined as objective functions measuring data distribution. These objectives are defined over subspaces and clusters to consider both local and global optimization criteria. The solution to the optimization problem is found by tensor decomposition methods. The solution of optimization can be used to measure changes in data distribution as well as compare changes across multiple levels of data. As future work, the proposed validation metrics on dynamic networks can be integrated with static graph mining algorithms. For example, graph distance metrics can be used to sequentially compare graphs by creating a time series of network changes from adjacent periods. The time series then act as the data distribution for the proposed validation metrics. Sensitivity functions and reconstruction errors in the data distribution can also be used as optimization objectives. These various objectives can be combined using multiobjective optimization measures. The solution to multiobjective optimization can then be found in terms of metaheuristic search that converges onto a globally optimum solution. Another topic for investigation is the encoding, indexing and retrieval of spatiotemporal networks as high-dimensional time series data distributions.

17.18 Surveys on Dynamic Networks

References [1–3] provide a comprehensive survey of the most recent developments in dynamic network analysis, evolutionary network analysis and temporal data analysis. Presently, the focus is on discussing the problems of outlier detection methods over complex networks with reference to these survey papers. Reference [3] classifies anomaly detection algorithms in dynamic networks by the types of anomalies

they detect—nodes, edges, subgraphs and events. For each of these methods, the available graph mining models are community-based models, compression-based models, decomposition-based models, clustering-based models, probability-based models, sequence-based models and time-based models. The above papers summarized fall into the categories of decomposition-based models, compression-based models, clustering-based models and probability-based models. Such models are an interesting extension to the emerging methods on graph behavior, change detection, event mining, outlier detection and evolution over dynamic networks. The features detecting spatiotemporal properties for machine learning over dynamic networks are commonly obtained from samples, trees, clusters, wavelets, kernels, splines, nets, filters, wrappers and factors in data series, sequences, graphs and networks. Reference [1] overviews the vast literature on graph evolution analysis by assuming the dynamic networks are composed of data streams. In data stream mining, all data mining models must be adapted to the one-pass constraint and high dimensionality of data streams. Thus, evolution analysis on data streams focuses on either modeling the change in data stream or correcting for results in data mining, or both. Direct evolution analysis is closely related to the problem of outlier detection in temporal networks because temporal outliers are often defined as abrupt change points. By this definition of outlier detection, most data mining problems such as clustering, classification, association rule mining can be generalized to network data. In the context of evolutionary graphs, additional graph-specific problems suitable for outlier detection include link prediction, preferential attachment, counting triangles, spectral methods, shortest path methods, graph matching, graph clustering, dense pattern mining and graph classification. Aggarwal and Subbian [1] also give an interesting summary of application domains for complex network analysis. These application domains include World Wide Web, Telecommunication Networks, Communication Networks, Road Networks, Recommendations, Social Network Events, Blog Evolution, Computer Systems, News Networks, Bibliographic Networks and Biological Networks. Future work in these domains includes content-centric analysis, co-evolution of the content with network structure, collective classification and domain problems to streaming networks. In this context, system structure and network topology can be studied by outlier detection methods. Gupta et al. [2] are focused on outlier detection for temporal data. Thus, the data under discussion may or may not be graph data. The temporal data collection mechanisms are classified into data streams, spatiotemporal data, distributed streams, temporal networks and time series data, generated by a multitude of applications. In this context, outlier detection has been studied on high-dimensional data, uncertain data, streaming data, network data and time series data. A common characteristic in all these problem formulations is that temporal continuity has key role in formulations of anomalous changes, sequences and patterns in the data. The paper then goes on to summarize the various outlier detection methods by four facets important to data mining and computational algorithms: Time series versus Multidimensional Data Facet, the Point versus Window Facet, the Data Type Facet and the Supervision Facet. The paper also provides an extensive section on applications and usage scenarios of outlier detection in dynamic networks. These scenarios span Environmental Sensor Data, Industrial Sensor Data, Surveillance and

Trajectory Data, Computer Networks Data, Biological Data, Astronomy Data, Web Data, Information Network Data and Economics Time Series Data. Thus, the existing survey papers on dynamic network analysis are focused on particular application domain or on a single research area. The assumptions made by various data mining models in each data mining model for dynamic network analysis are critical for design and analysis of computational algorithms.

17.19 Adversarial Learning—Secure Machine Learning [4–7, 15, 16]

Many algorithms in use today are not provably secure. Adversarial learning is concerned with making machine learning algorithms secure. Here, security is measured in terms of an objective function which also considers the reliability, diversity and significance of the algorithm. Thus, the objective of adversarial learning is to show that a specific machine learning algorithm or a data mining model or an analytics solution can be made insecure only with an impractical amount of computational work. Adversarial learning assumes that the underlying pattern in the data is non-stationary when training or testing this algorithm. In the context of such non-stationary data, the standard machine learning assumption of identically distributed data is violated.

Furthermore, it is assumed that the non-stationary data is caused by an intelligent adversary who is attacking the algorithms. In other words, the feature test data is the data generated at application time to minimize a cost function.

Thus, adversarial learning incorporates defense mechanisms into the algorithmic design and action. Thus, in literature, the analysis frameworks for adversarial learning specify the learning and attack processes in terms of the corresponding objective functions.

The attack processes also specify the attacker's constraints and optimal attack policy. The learning processes also specify the algorithm's gain and attacker's gain under the optimal attack policy.

Adversarial learning has application in areas such as spam filtering, virus detection intrusion detection, fraud detection, biometric authentication, network protocol verification, computed advertising, recommender systems, social media web mining and performance monitoring.

Improvements to the existing status of adversarial learning are concerned with dynamic features and regularized parameters computed while processing high-dimensional data for correlations. Such correlations may exist between one and more of samples, features, constraints, indicators and classes in computational algorithms. With changing spatiotemporal data, these correlations also change within the purview of conflicting goals of data mining algorithms such as accuracy, robustness, efficiency and scalability. Such conflicting properties of the algorithms are cast in the garb of a search and optimization framework finding the best (linear and nonlinear)

decision boundaries between classes of interest. The objective function of the optimization algorithm may have either open or closed-form solutions. The objective of optimization can be either one or multiple objectives functions. Deep learning over time has been found to be susceptible to adversarial examples. Thus, applying the ideas of adversarial learning to make robust deep learning algorithms is an interesting area of research.

In terms of machine learning, the supervision facet of deep learning algorithms connects the data analytics output of dynamic networks and adversarial learning since single or multiple adversaries must learn about the classifier using some condition of prior knowledge, observation and experimentation formulated as cost functions or loss functions or utility functions or payoff functions.

The supervision over time facet of deep learning is also observed in related areas of research like dictionary learning, representation learning, semi-supervised learning, similarity learning, ensemble learning, online learning and transfer learning.

To discuss computational complexity of such kinds of machine learning, adversarial learning and dynamic network analysis, we can use ideas from game theory, linear algebra, causal inference, information theory and graph theory. The features (of networks, sets, sequences and series) for adversarial learning can also be obtained from complex networks, dynamic networks and knowledge graphs.

Ideas for feature extraction can be obtained from research into data mining problems like biclustering, structure similarity search, quasi cliques mining, dense subgraph discovery, multilevel graph partitioning, hypo-graph partitioning, agglomerative and divisive graph clustering.

17.20 Conclusion and Future Emerging Direction

In this chapter, we have surveyed in detail the current research trends and also identified the future emerging trends of research in Big Data Analytics. Evidently, some of the future emerging, critical research directions are in Adversarial Machine Learning in the context of dynamic networks.

17.21 Review Questions

1. What are the various data mining techniques and what are their application domains?
2. What is data streaming?
3. What is Adversarial Machine Learning?
4. What is dynamic network analysis? Explain.
5. How Adversarial Machine Learning and dynamic network analysis are related?
6. What algorithms of data mining can be adopted for dynamic network analysis?

7. What are the application domains for (a) dynamic network analysis? (b) What are the research challenges in applying them?
8. What are the various outline detection and analysis techniques in dynamic networks?
9. How states network analysis techniques, state sequence analysis techniques and state set analysis techniques are applicable to dynamic network analysis problems?
10. What is practical real-life illustration of dynamic networks?
11. What are the existing research challenges in Big Data Analytics?
12. Explain salient features of approaches in literature for research in dynamic network analysis.
13. Explain how dynamic network analysis can be made? What are the salient features?
14. Explain sampling.
15. Explain validation metrics.
16. Explain block modeling.
17. How secure machine learning can be adopted in adversarial conditions?
18. What is Adversarial Security Mechanism?

References

1. C. Aggarwal, K. Subbian, Evolutionary network analysis: a survey. ACM Comput. Surv. **47**(1):10:1–10:36 (2014)
2. M. Gupta, J. Gao, C.C. Aggarwal, J. Han, Outlier detection for temporal data: a survey. IEEE Trans. Knowl. Data Eng. **26**(9), 2250–2267 (2014)
3. S. Ranshous, S. Shen, D. Koutra, S. Harenberg, C. Faloutsos, F.N. Samatova, Anomaly detection in dynamic networks: a survey. **7**, 223–247 (2015)
4. I.J. Goodfellow, J. Shlens, C. Szegedy, Explaining and Harnessing Adversarial Examples (2014). ArXiv e-prints
5. W. Liu, S. Chawla, J. Bailey, C. Leckie, K. Ramamohanarao, AI 2012: advances in Artificial Intelligence: 25th Australasian Joint Conference, Sydney, Australia, 4–7 Dec, 2012, in *Proceedings, Chapter An Efficient Adversarial Learning Strategy for Constructing Robust Classification Boundaries* (Springer, Berlin, Heidelberg, 2012), pp. 649–660
6. N. Papernot, P. McDaniel, I. Goodfellow, S. Jha, Z. Berkay Celik, A. Swami, Practical Black-Box Attacks against Deep Learning Systems using Adversarial Examples (2016). ArXiv e-prints
7. M. Vidyadhari, K. Kiranmai, K.R. Krishniah, D.S. Babu, Security evaluation of pattern classifiers under attack. Int. J. Res. **3**(01), 1043–1048 (2016)
8. C.C. Aggarwal, Y. Zhao, P.S. Yu, Outlier detection in graph streams, in *Proceedings of the 2011 IEEE 27th International Conference on Data Engineering, ICDE'11*. IEEE Computer Society Washington, DC, USA, 2011, pp. 399–409
9. M. Jiang, A. Beutel, P. Cui, B. Hooi, S. Yang, C. Faloutsos, A general suspiciousness metric for dense blocks in multimodal data, in *2015 IEEE International Conference on Data Mining, ICDM 2015*, Atlantic City, NJ, USA, 14–17 Nov 2015, pp. 781–786
10. J. Sun, C. Faloutsos, S. Papadimitriou, P.S. Yu, Graphscope: Parameter-free mining of large time-evolving graphs, in *Proceedings of the 13th ACM SIGKDD International Conference*

on Knowledge Discovery and Data Mining, KDD'07 (New York, NY, USA. ACM, 2007), pp. 687–696
11. M. Davis, W. Liu, P. Miller, G. Redpath, Detecting anomalies in graphs with numeric labels, in *Proceedings of the 20th ACM International Conference on Information and Knowledge Management, CIKM'11* (ACM, New York, NY, USA, 2011) , pp. 1197–1202
12. M. Mongiov, P. Bogdanov, R. Ranca, E.E. Papalexakis, C. Faloutsos, A.K. Singh, *NetSpot: Spotting Significant Anomalous Regions on Dynamic Networks*, Chapter 3, pp. 28–36
13. M. Gupta, C.C. Aggarwal, J. Han, Y. Sun, Evolutionary clustering and analysis of bibliographic networks, in *2011 International Conference on Advances in Social Net-Works Analysis and Mining (ASONAM)*, pp. 63–70
14. J. Chan, N.X. Vinh, W. Liu, J. Bailey, C.A. Leckie, K. Ramamohanarao, J. Pei, Advances in Knowledge Discovery and Data Mining: 18th Pacific-Asia Conference, PAKDD 2014, Tainan, Taiwan, 13–16 May 2014, in *Proceedings, Part I, chapter Structure-Aware Distance Measures for Comparing Clusterings in Graphs* (Springer International Publishing, Cham, 2014) pp. 362–373
15. F. Wang, W. Liu, S. Chawla, On sparse feature attacks in adversarial learning, in *2014 IEEE International Conference on Data Mining*, 2014, pp. 1013–1018
16. H. Xiao, B. Biggio, B. Nelson, H. Xiao, C. Eckert, F. Roli, Support vector machines under adversarial label contamination. J. Neuro Comput., Spec. Issue Adv. Learn. Label Noise (2014 in press)