Summarizing Static and Dynamic Big Graphs

Arijit Khan
Nanyang Technological University, Singapore
arijit.khan@ntu.edu.sg

Sourav S. Bhowmick
Nanyang Technological University, Singapore
assourav@ntu.edu.sg

Francesco Bonchi
ISI Foundation, Italy
francesco.bonchi@isi.it

ABSTRACT

Large-scale, highly-interconnected networks pervade our society and the natural world around us, including the World Wide Web, social networks, knowledge graphs, genome and scientific databases, medical and government records. The massive scale of graph data often surpasses the available computation and storage resources. Besides, users get overwhelmed by the daunting task of understanding and using such graphs due to their sheer volume and complexity. Hence, there is a critical need to summarize large graphs into concise forms that can be more easily visualized, processed, and managed. Graph summarization has indeed attracted a lot of interests from various research communities, such as sociology, physics, chemistry, bioinformatics, and computer science. Different ways of summarizing graphs have been invented that are often complementary to each other. In this tutorial, we discuss algorithmic advances on graph summarization in the context of both classical (e.g., static graphs) and emerging (e.g., dynamic and stream graphs) applications. We emphasize the current challenges and highlight some future research directions.

1. INTRODUCTION

Graph data management and mining has become a hot topic in the database research community in recent years, influenced by the growth of knowledge bases and varieties of networks on the Web, as well as with the improvements in technology that has resulted in untapped sources of information. Querying and reasoning about the interconnections between entities in a graph dataset can lead to interesting and deep insights into a variety of phenomena. However, due to sheer volume, complexity, and temporal characteristics, a starting point to analyze these graphs is often a concise representation (i.e., summary) that helps to understand these datasets as well as to formulate queries in a meaningful way.

A summary is a concise representation (either lossless or lossy) of the original graph, whose objectives can greatly vary, e.g., from reducing the number of bits needed for encoding the original graph [4, 5], to more complex database-style operations that summarize graphs where the resolution could be scaled-up or scaled-down interactively [10, 47]. With the advent of dynamic graphs and streams, there is a demand for analyzing the time-evolving properties of such graphs, and once again graph synopsis construction has found increasing interests [45, 54].

Graph summarization is beneficial to a wide range of applications as follows.

- **Interactive and Exploratory Analysis.** Knowledge graphs are complex with many attributes on nodes and edges, hence an important task is to summarize these graphs by grouping nodes and edges that share similar structures and contents for better understanding and query formulation [10, 47]. In protein interaction networks, summarization at multiple resolutions provides high level views of its functional landscape and brings the opportunity to investigate higher level organization and modularity [40].
- **Processing in Modern Hardware.** In order to process fast streaming data, a growing number of applications relies on devices such as network interface cards, routers, switches, cell processors, FPGAs, and GPUs [39]; and usually these devices have very small on-chip memory. Therefore, efficient processing of rapid and massive graph stream data in specialized hardware requires creation of a succinct synopsis, e.g., GSketch [54] and GMatrix [21, 22].
- **Approximate Query Processing.** Query processing over graph summaries can significantly improve the efficiency, often at the cost of tolerable errors. Examples include reachability and pattern matching queries [16]. The summary size, in fact, can be varied to trade-off between accuracy and efficiency [54].
- **Visualization.** Due to large size of social networks, RDFs, webgraphs, and biological datasets, it is difficult to visualize them in a meaningful way. To overcome the visual complexity of very large networks, a semantic abstraction is often necessary [43].
- **Data-driven Visual Graph Query Interface Construction.** Graph queries are more intuitive to draw than to compose them in textual format. Consequently, visual query interfaces (a.k.a GUI) that can enable an end user to draw a graph query interactively have gained increasing attention in recent times. Recently, graph summarization is exploited to generate content of such GUI automatically from the underlying graph dataset [53].
- **Distributed Graph Systems.** Large amount of graph data shuffling and access over networks is a concern for distributed graph systems. In this context, building an effective graph summary, e.g., based on the locality of data access [18], is often critical. GBASE [20], which is a distributed

Proceedings of the VLDB Endowment, Vol. 10, No. 7

graph system and follows the principle of matrix-based operations, developed a novel block compression method to efficiently store homogeneous regions of graphs. This leads up to 50× less storage and faster running time.

While graph summarization is an active area of research with a wide variety of applications, a limited effort has been devoted to survey the research developed [9, 24, 26, 29, 48], and this effort has often targeted specific subproblems (e.g., interactive [11, 48] and mining-based [9] graph summarization), or specific sub-areas (e.g., static networks [29]). This tutorial gives a comprehensive introduction to the topic of graph summarization, discussing state of the art in the industry and in the academic world. A brief overview of the scope of the tutorial is as follows.

- **Graph summarization categories:** We classify and review different categories of graph summarization techniques that have been proposed in the literature. In this context, we also highlight a number of evaluation metrics used to measure the quality of various categories of summary.
- **Summarizing static graphs:** We cover various strategies for summarizing graphs that are assumed to be static (i.e., they do not evolve with time).
- **Summarizing dynamic and stream graphs:** With the prevalence of large-scale dynamic graphs and streams, there is an increasing demand to effectively summarize them. In this part, we review efforts in this direction.
- **Future research directions:** Finally, we discuss open problems on the topic of graph summarization and providing possible directions for future work.

Target Audience and Prerequisites. This tutorial is intended to benefit researchers, system designers, and developers in the broad area of graphs querying, mining, and storage that include but not limited to RDF query, Web search, Ontology and Semantic Web, linked data, streams, social/information networks, genomic, and machine learning. This tutorial does not require any in-depth knowledge on complex graph algorithms and summary techniques.

What We Shall Not Cover in this Tutorial. We shall not discuss various other related graph analytics problems such as sampling, sparsification, community detection, graph embedding, clustering, partitioning, and dense subgraph mining.

2. TUTORIAL OUTLINE

Our presentation follows a top-down approach, starting from motivation for summarizing graphs, proceeding to categories of graph summarization, static and dynamic graph summarization techniques, and concluding with future research directions in this arena. Table 1 shows the key papers discussed in this tutorial.

Summarization Category	Papers
Aggregation-based (static graphs)	[10, 34, 38, 47]
Attribute-based (static graphs)	[40, 53]
Dynamic graphs	[41, 42, 46]
Stream graphs	[22, 45, 54]

Table 1: Representative papers

2.1 Categories of Graph Summary

Graph summarization techniques can be classified into several ways based on their objectives and characteristics of the specific algorithms employed.

Lossless vs. Lossy Summarization. In lossless compression, one can exactly recover the original graph from the summary. On the contrary, for lossy compression, we may not fully recover the original graph; however, such techniques often result in a better compression ratio.

Non-overlapping vs. Overlapping. In overlapping summaries, a node may belong to multiple summarized components, adding more flexibility to summarization techniques; however, a summary with many highly overlapping components could be less intuitive and more complicated than their non-overlapping counterparts.

Summary for Different Graph Categories. The goal of homogeneous graphs summarization is to summarize the topology information. In case of heterogenous graphs, nodes and edges have diverse types and several attributes; therefore, the summarization happens at both structural and semantic levels by considering relationships across attributes and types.

Summarization Techniques.

- *Aggregation-based* techniques create a summary with super-nodes and super-edges, and are useful in understanding and visualizing complex graphs, as well as in efficient storage and query processing.
- *Attribute-based* techniques create a summary that leverage both topology as well as attributes associated with nodes/edges of the graph to generate a high-level summary of the underlying graphs.
- *Compression* methods aim at reducing the space required for encoding the original graph, primarily based on the structural information.
- *Application-oriented* summarization techniques aim for efficient query answering over summary graphs, often with theoretical approximation guarantees.

Evaluation Metrics.

- *Space requirement.* Commonly used metrics are as follows: Reduction of graph size in number of nodes and edges, total data size in bytes, number of nonempty blocks in the graph adjacency matrix, bits per edge.
- *Efficiency.* This is measured by the time required for summarization (i.e., pre-processing time) and query processing time on the summaries (i.e., on-line efficiency).
- *Accuracy.* Reconstruction errors, entropy, quality of answers (e.g., degree, centrality, connectivity), etc.
- *Interestingness.* Visualization quality, user study, diversity, coverage, conciseness.

2.2 Summarizing Static Graphs

We emphasize on four different summarization techniques for static graphs.

Aggregation-based Summary. Notable techniques under this category are pattern mining and community based summarization [7], OLAP [10, 55], set-based aggregation using locality sensitive hashing [23], super-node and edge-correction [34], super-node and reconstruction-error [27, 38], SNAP [47, 52], and distributed graph summarization [30], among many others. In this tutorial, we shall discuss [10, 34, 38, 47]. The main idea of [34] is to merge similar nodes into a super-node, then add a super-edge between two super-nodes conditionally, as well as keep edge-corrections to support lossless summarization. As opposed to this, [38] merges similar nodes into a super-node, together with a theoretically-bounded reconstruction error. Clearly, these techniques are suitable for structural summarization over homogeneous graphs only. SNAP [47] and OLAP [10], on the other hand, allow interactive summarization at various resolutions over heterogeneous networks.

Attribute-based. Nodes and edges of many real-world networks are annotated with attributes. Thus, it is important to consider not only topology, but also semantics of the node and edge attributes in order to generate meaningful summaries. To this end, we shall discuss FUSE [40], a functional summarization technique for protein interaction networks. We shall present its role in comprehending high-level functional relationships in disease-related PPI networks such as Alzheimer's disease network. We shall also present topology and attribute-based summarization of a large collection of small graphs (e.g., chemical compounds) and its application in constructing data-driven visual graph query interfaces [53].

Compression. Due to the prevalence of large-scale social networks and web graphs, their compression techniques have received much attention. Boldi and Vigna [5] showed that web graphs are compressible down to almost two bits per edge. Chierichetti et al. [12] extended the framework using shingle ordering instead of lexicographical ordering of web pages, in order to tackle social networks. Finding an order of nodes, which captures the "regularity" of the network, is indeed a challenging problem. Very recently, Boldi et al. [4] introduced a layered label propagation algorithm for reordering very large graphs. Other interesting works include [6, 13, 19, 25, 37].

Application-Oriented Summary. These are graph summarization techniques for efficient query answering and pattern mining, such as reachability, shortest path, and pattern matching queries [16, 49, 56], keyword search [51], distributed graph computation [20], graph mining [8, 14, 33], eigenvector centrality [27], neighborhood query [32], information cascade and influential node discovery [35, 36, 44], etc.

2.3 Summarizing Dynamic Graphs and Streams

The sources of networked data have increased dramatically due to advances in devices and networking technologies, internet-of-things (IoT). Examples span smart phones and sensors to emerging applications that capture user actions such as edits to documents and source code modifications. This results in temporal graphs which can be viewed as graphs that change over time. With the prevalence of large-scale dynamic graphs and streams, there is an increasing demand to effectively summarize them.

Summary for Dynamic Graphs. Shah et. al. developed Time-Crunch [42] for constructing concise summaries of large, real-world dynamic graphs in order to better understand their underlying behavior. In particular, they employed the MDL (Minimum Description Length) principle to appropriately describe graphs over time using a lexicon of temporal phrases which describe temporal connectivity pattern. DiffNet [41] was designed specifically for biological networks with the goal to automatically construct a high-quality *differential* summary of two snapshots of *epistatic miniarray profile* (E-MAP) networks [3] under contrasting environmental conditions. This enables us to understand functional modules that are differentially effected by the DNA-damaging agent. Tsalouchidou et. al. extended the idea of graph summarization with reconstruction error from the domain of static graphs to a series of dynamic graphs, via an approach, called Tensor Streaming [46]. We shall discuss [41, 42, 46] during the presentation.

Summary for Graph Streams. Due to the availability of massive streams, the problem of graph stream synopsis construction has found increasing interest, e.g., spanners, sparsifiers, and sketches [31]. The main challenge here is that the summary needs to be constructed in one pass (or, a limited number of passes) over the stream, and must be updated incrementally with every incoming item in the stream. gSketch [54] was proposed to estimate edge frequencies. The method in [15] constructs synopsis of graph streams for estimating the degree distributions of the nodes. A method in [17] constructs synopsis structures that are useful for the case of distance-based computations. Ahn et. al. [1, 2] studied graph sketch for answering structural queries such as connectivity, minimum-cost spanning tree, maximum weighted matching, and subgraph pattern matching. Very recently, more advanced graph sketches were proposed in TCM [45] and GMatrix [21, 22], addressing a combination of structural and frequency estimation queries. In this tutorial, we shall discuss [22, 45, 54], and how they estimate statistics combining both structure and frequency.

2.4 The Road Ahead

Lastly, we expose potential research issues and future directions in summarizing big graphs, such as:

- Summarizing networks with additional information, such as uncertainty, spatial and textual data, multi-layer and multi-view networks.
- Advanced applications, e.g., brain networks alignment, database schema matching and entity resolution, documents and activity summarization, latent and deep node representations learned from the context encoded in the graph [50], finding similarities and differences across a set of large graphs.

3. HISTORY OF THE TUTORIAL

To the best of our knowledge, this tutorial has not been presented in any major database or data mining conference.

4. BIOGRAPHICAL SKETCHES

Arijit Khan is an Assistant Professor in the School of Computer Science and Engineering at Nanyang Technological University, Singapore. He earned his PhD from the Department of Computer Science, University of California, Santa Barbara, USA, and did a post-doc in the Systems group at ETH Zurich, Switzerland. Arijit is the recipient of the IBM PhD Fellowship in 2012-13. He published several papers in premier database and data-mining conferences and journals including SIGMOD, VLDB, TKDE, ICDE, SDM, EDBT, and CIKM. Arijit co-presented tutorials on emerging graph queries, big-graph systems, and uncertain graphs at ICDE (2012) and VLDB (2014, 2015), and served in the program committee of KDD, SIGMOD, VLDB, ICDE, ICDM, EDBT, WWW, and CIKM. Arijit served as the co-chair of Big-O(Q) workshop co-located with VLDB 2015. More information at `http://www.ntu.edu.sg/home/arijit.khan/`.

Sourav S. Bhowmick is an Associate Professor in the School of Computer Science and Engineering (SCSE), Nanyang Technological University. He leads the data management research group (DANTe) in SCSE. His research has appeared in top-tier venues in data management and analytics such as SIGMOD, VLDB, ICDE, VLDB Journal, TKDE, WWW, and KDD. Sourav has been keynote and tutorial speaker for several international conferences. He has received Best Paper Awards at ACM CIKM 2004 and ACM BCB 2011 for papers related to evolution mining and biological network summarization, respectively. His work on influence maximization was nominated for the best paper award in ACM SIGMOD 2015. His research on biological network summarization is recently published as a book entitled "Summarizing Biological Networks" by Springer-Verlag, Computational Biology Series (May

2017). Sourav has served as a PC member of premium data management and data mining conferences (e.g., VLDB, ICDE, KDD, ICDM) and reviewer for various premium journals (e.g., TKDE, VLDB Journal). More information at `http://www.ntu.edu.sg/home/assourav/`.

Francesco Bonchi is Research Leader at the ISI Foundation, Turin, Italy, where he leads the Algorithmic Data Analytics group. He is also (part-time) Principal Scientist for Data Mining at Eurecat Barcelona. He was Director of Research at Yahoo Labs in Barcelona, Spain, leading the Web Mining Research group. He was PC Chair of IEEE ICDM'16 and ACM HT'17 and is a member of the ECML PKDD Steering Committee, and Associate Editor of many journals in the data management and mining area (IEEE TBD, IEEE TKDE, ACM TIST, KAIS, DMKD). He presented tutorials at ACM KDD'14, ACM KDD'16 and at WWW'16. More information at `http://francescobonchi.com/`.

5. ACKNOWLEDGEMENT

The research of Arijit Khan is supported by AcRF MOE Tier-1 M401020000 and NTU M4081678. Sourav S Bhowmick was supported by the Singapore-MOE AcRF Tier-2 Grant MOE2015-T2-1-040. Any opinions, findings, and conclusions in this publication are those of the authors and do not necessarily reflect the views of the funding agencies.

6. REFERENCES

[1] K. J. Ahn, S. Guha, and A. McGregor, *Analyzing Graph Structure via Linear Measurements*, SODA, 2012.

[2] K. J. Ahn, S. Guha, and A. McGregor, *Graph Sketches: Sparsification, Spanners, and Subgraphs*, PODS, 2012.

[3] S. Bandyopadhyay, M. Mehta, D. Kuo, M.-K. Sung, R. Chuang, E. J. Jaehnig, B. Bodenmiller, K. Licon, W. Copeland, M. Shales, D. Fiedler, J. Dutkowski, A. Guénolé, H. van Attikum, K. M. Shokat, R. D. Kolodner, W.-K. Huh, R. Aebersold, M.-C. Keogh, N. J. Krogan, and T. Ideker, *Rewiring of Genetic Networks in Response to DNA Damage.* , Science, **330**(6009), 1385–9.

[4] P. Boldi, M. Rosa, M. Santini, and S. Vigna, *Layered Label Propagation: a Multiresolution Coordinate-free Ordering for Compressing Social Networks*, WWW, 2011.

[5] P. Boldi and S. Vigna, *The Webgraph Framework I: Compression Techniques*, WWW, 2004.

[6] N. R. Brisaboa and S. Ladra and G. Navarro, *k2-Trees for Compact Web Graph Representation*, SPIRE, 2009.

[7] G. Buehrer and K. Chellapilla, *A Scalable Pattern Mining Approach to Web Graph Compression with Communities*, WSDM, 2008.

[8] C. Chen, C. X. Lin, M. Fredrikson, M. Christodorescu, X. Yan, and J. Han, *Mining Graph Patterns Efficiently via Randomized Summaries*, VLDB, 2009.

[9] C. Chen, C. X. Lin, M. Fredrikson, M. Christodorescu, X. Yan, and J. Han, *Mining Large Information Networks by Graph Summarization*. Link Mining: Models, Algorithms, and Applications. (pp. 475-501), Springer New York, 2010.

[10] C. Chen, X. Yan, F. Zhu, J. Han, and P. S. Yu, *Graph OLAP: Towards Online Analytical Processing on Graphs*, ICDM, 2008.

[11] C. Chen, F. Zhu, X. Yan, J. Han, P. S. Yu, and R. Ramakrishnan, *InfoNetOLAP: OLAP and Mining of Information Networks*. Link Mining: Models, Algorithms, and Applications. (pp. 411-438), Springer New York, 2010.

[12] F. Chierichetti, R. Kumar, S. Lattanzi, M. Mitzenmacher, A. Panconesi, and P. Raghavan, *On Compressing Social Networks*, KDD, 2009.

[13] Y. Choi and W. Szpankowski, *Compression of Graphical Structures: Fundamental Limits, Algorithms, and Experiments*, Information Theory, IEEE Transactions on, 58(2):620-638, 2012.

[14] D. J. Cook and L. B. Holder, *Substructure Discovery Using Minimum Description Length and Background Knowledge*, J. Artif. Int. Res., 1(1), 231-255, 1994.

[15] G. Cormode and S. Muthukrishnan, *Space Efficient Mining of Multigraph Streams*, PODS, 2005.

[16] W. Fan, J. Li, X. Wang, and Y. Wu, *Query Preserving Graph Compression*, SIGMOD, 2012.

[17] J. Feigenbaum, S. Kannan, A. McGregor, S. Suri, and J. Zhang, *Graph Distances in the Data Stream Model*, SIAM J. Computing, 38(5), 2005.

[18] W. Han and Y. Miao and K. Li and M. Wu and F. Yang and L. Zhou and V. Prabhakaran and W. Chen and E. Chen, *Chronos: A Graph Engine for Temporal Graph Analysis*, EuroSys, 2014.

[19] U. Kang and C. Faloutsos, *Beyond 'Caveman Communities': Hubs and Spokes for Graph Compression and Mining*, ICDM, 2011.

[20] U. Kang, H. Tong, J. Sun, C. Y. Lin, and C. Faloutsos, *GBASE: A Scalable and General Graph Management System*, KDD, 2011.

[21] A. Khan and C. Aggarwal, *Query-Friendly Compression of Graph Streams*, ASONAM, 2016.

[22] A. Khan and C. Aggarwal, *Towards Query-Friendly Compression of Rapid Graph Streams*, Social Network Analysis and Mining, Springer, 2017.

[23] K. U. Khan, W. Nawaz, and Y.-K. Lee, *Set-based Unified Approach for Summarization of a Multi-Attributed Graph*, WWW, 2016.

[24] D. Koutra, *Summarizing Large-Scale Graph Data*, SDM, 2017.

[25] D. Koutra, U. Kang, J. Vreeken, and C. Faloutsos, *Summarizing and Understanding Large Graphs*, Stat. Anal. Data Min., 8(3), 183-202, 2015.

[26] Y. Liu, A. Dighe, T. Safavi, and D. Koutra, *A Graph Summarization: A Survey*, CoRR, abs/1612.04883, 2016.

[27] K. LeFevre and E. Terzi, *GraSS: Graph Structure Summarization*, SDM, 2010.

[28] C.-T. Li and S.-D. Lin, *Egocentric Information Abstraction for Heterogeneous Social Networks*, ASONAM, 2009.

[29] S.-D. Lin, M.-Y. Yeh, and C.-T. Li, *Sampling and Summarization for Social Networks*, SDM, 2013.

[30] X. Liu, Y. Tian, Q. He, W.-C. Lee, and J. McPherson, *Distributed Graph Summarization*, CIKM, 2014.

[31] A. McGregor, *Graph Stream Algorithms: A Survey*, SIGMOD Rec., 43(1), 2014.

[32] H. Maserrat and J. Pei, *Neighbor Query Friendly Compression of Social Networks*, KDD, 2010.

[33] H. Maserrat and J. Pei, *Community Preserving Lossy Compression of Social Networks*, ICDM, 2012.

[34] S. Navlakha, R. Rastogi, and N. Shrivastava, *Graph Summarization with Bounded Error*, SIGMOD, 2008.

[35] M. Purohit, B. A. Prakash, C. Kang, Y. Zhang, and V.S. Subrahmanian, *Fast Influence-Based Coarsening For Large Networks*, KDD, 2014.

[36] Q. Qu, S. Liu, C. S. Jensen, F. Zhu, and C. Faloutsos, *Interestingness-Driven Diffusion Process Summarization in Dynamic Networks*, ECML PKDD, 2014.

[37] S. Raghavan and H. Garcia-Molina, *Representing Web Graphs*, ICDE, 2003.

[38] M. Riondato and D. García-Soriano and F. Bonchi, *Graph Summarization with Quality Guarantees*, Data Min. Knowl. Discov., 31(2), 314-349, 2017.

[39] O. Rottenstreich, Y. Kanizo, and I. Keslassy, *The Variable-Increment Counting Bloom Filter*, IEEE/ACM Trans. Netw., 22(4):10921105, 2014.

[40] B.-S. Seah, S. S. Bhowmick, C. F. Dewey Jr, and H. Yu, *FUSE: A Profit Maximization Approach for Functional Summarization of Biological Networks*, BMC Bioinformatics, 13(Suppl 3):S10, 2012.

[41] B.-S. Seah, S. S. Bhowmick, and C. F. Dewey Jr, *DiffNet: Automatic Differential Functional Summarization of dE-MAP Networks.*, Methods, 69(3), 2014.

[42] N. Shah, D. Koutra, T. Zou, B. Gallagher, and C. Faloutsos, *TimeCrunch: Interpretable Dynamic Graph Summarization*, KDD, 2015.

[43] Z. Shen, K. L. Ma, and T. Eliassi-Rad, *Visual Analysis of Large Heterogeneous Social Networks by Semantic and Structural Abstraction*, IEEE Transactions on Visualization and Computer Graphics, 12(6), 14271439, 2006.

[44] L. Shi, S. Sun, Y. Xuan, Y. Su, H. Tong, S. Ma, and Y. Chen, *TOPIC: TOward Perfect InfluenCe Graph Summarization*, ICDE, 2016.

[45] N. Tang, Q. Chen, and P. Mitra, *Graph Stream Summarization: From Big Bang to Big Crunch*, SIGMOD, 2016.

[46] I. Tsalouchidou, G. D. F. Morales, F. Bonchi, and R. A. Baeza-Yates, *Scalable Dynamic Graph Summarization*, IEEE Big Data, 2016.

[47] Y. Tian, R. A. Hankins, and J. M. Patel, *Efficient Aggregation for Graph Summarization*, SIGMOD, 2008.

[48] Y. Tian and J. M. Patel, *Interactive Graph Summarization*. Link Mining: Models, Algorithms, and Applications. (pp. 389-410), Springer New York, 2010.

[49] H. Toivonen, F. Zhou, A. Hartikainen, and A. Hinkka, *Compression of Weighted Graphs*, KDD, 2011.

[50] D. Wang, P. Cui, and W. Zhu, *Structural Deep Network Embedding*, KDD, 2016.

[51] Y. Wu, S. Yang, M. Srivatsa, A. Iyengar, and X. Yan, *Summarizing Answer Graphs Induced by Keyword Queries*, VLDB, 2013.

[52] N. Zhang, Y. Tian, and J. M. Patel, *Discovery-driven Graph Summarization*, ICDE, 2010.

[53] J. Zhang, S. S. Bhowmick, H. H. Nguyen, B. Choi, F. Zhu. *DAVINCI: Data-driven Visual Interface Construction for Subgraph Search in Graph Databases*, ICDE, 2015.

[54] P. Zhao, C. Aggarwal, and M. Wang, *gSketch: On Query Estimation in Graph Streams*, VLDB, 2012.

[55] P. Zhao, X. Li, D. Xin, and J. Han, *Graph Cube: On Warehousing and OLAP Multidimensional Networks*, SIGMOD, 2011.

[56] F. Zhou, S. Malher, and H. Toivonen, *Network Simplification with Minimal Loss of Connectivity*, ICDM, 2010.