Summary of Research

Arijit Khan

Graphs are widely used in many application domains, including social networks, knowledge graphs, biological networks, software collaboration, geo-spatial road networks, interactive gaming, among many others. One major challenge for graph querying and mining is that non-professional users are not familiar with the complex schema and variational information descriptions. It becomes hard for users to formulate a query (e.g., SPARQL or exact subgraph pattern) that can be properly processed by the existing systems. As an example, Freebase (Google Knowledge Graph was powered in part by Freebase) alone has over 22 million entities and 350 million relationships in about 100 domains. The Microsoft Concept Graph has more than 5.4 million concepts; the Knowledge and Action Graph of Microsoft has 21 billion facts, 18 billion action links, and over five billion relationships between more than one billion people, places, and things — and these graph datasets are growing. The Linking Open Data (LOD) community had interlinked over 52 billion RDF triples spanning over several hundred datasets. Before users (including biologists, chemists, data journalists, and social scientists) and developers can do anything meaningful with the graph data, they are overwhelmed by the daunting task of attempting to even digest and understand it (**ACM-SIGMOD-Blog17**).

I research on data management for emerging problems in large networks, with a focus on user-friendly, efficient, and approximate querying and pattern mining in social and information networks, using scalable algorithms, machine learning techniques, and distributed systems. My research is driven by a strong desire to make complex big-graphs comprehensible, actionable, and useful for critical, real-world applications including online querying of knowledge graphs and streams, supporting human-in-the-loop network exploration, web search, user modeling, entity resolution, data cleaning, malware detection, co-occurring patterns discovery and root cause analysis, social and professional recommendations, online advertisements, social networks analysis, uncertainty and reliability estimation, and viral marketing (**ICDE12**).

I design effective, approximate, efficient, and user-friendly algorithms and large-scale systems for answering emerging, complex queries over big graphs. I summarize in the following my past and current works.

User-friendly Online Querying of Heterogeneous Networks

• **Subgraph-Matching-Based Online Graph Querying.** To query networks, it is often necessary to identify the matches of a given query graph in a typically large network. Due to noise and lack of schema, structured methods such as SPARQL – which require an underlying schema to formulate a query – are often too restrictive. However, the user can still come up with some reasonable graph representation of the query. Such graphical representation may not be unique, and there might not be an exact match of the query graph in the data graph. Therefore, instead of finding the exact matches for a given query, it is more appealing to find the top-k approximate matches. Unfortunately, state-of-the-art graph similarity measures such as graph edit distance, maximum common subgraph, and counts of missing edges are not appropriate for capturing the semantics of the user's query. This motivates us to investigate fast subgraph matching techniques suitable for

query answering, which can relax the rigid structural constraints required for subgraph isomorphism and other traditional graph similarity metrics.

Our proposed graph similarity measure is based on the following observation: *if two nodes are close in a query graph, the corresponding nodes in the result graph must also be close* (**SIGMOD11**). Based on this notion of structural proximity, we propose a *neighborhood vectorization* model that converts the neighborhood of a node in the form of a multi-dimensional vector, yet it preserves the proximity information among the neighboring nodes up to a certain number of hops. Our neighborhood vectorization model allows very fast subgraph matching since all the operations can be performed algebraically over a vector space instead of an expensive graph space.

Though our problem of identifying the top-k graph matches using the neighborhood vectorization model remains **NP**-hard and **APX**-hard, we find it similar to the *max-sum inference* problem in graphical models. Therefore, we propose an iterative inference algorithm NeMa (**PVLDB13**), following the *loopy belief propagation* method used for inferencing in Random Markov fields. The inference method in NeMa iteratively boosts the score of more promising candidate nodes, considering both node-label and structural similarity. The time complexity of our algorithm is polynomial in the number of query nodes and their potential matches in the data graph. Based on a detailed empirical evaluation, our proposed method outperforms state-of-the-art graph querying and keyword search techniques such as BLINKS, SAGA, IsoRank, and gStore in terms of both effectiveness and efficiency.

• **Graph Query By Example.** A relatively new paradigm for user-friendly graph querying is graph-query-by-example. Query-by-example (QBE) has a positive history in relational databases, HTML tables, and entity sets. Our work, GQBE (**TKDE15**) adapted similar ideas over knowledge graphs. In particular, the user may find it difficult how to precisely write her query, but she might know a few answers to her query. The GQBE system allows her to input the answer tuple as a query, and directly returns other similar tuples that are present in the target graph. The underlying system follows a two-step approach. Given the input example tuple(s), it first identifies the query graph that captures the user's query intent. Then, it evaluates the query graph to find other relevant answer tuples.

• **Stream Graphs Summarization and Querying.** Many graphs such as those formed by the activity on social networks, communication networks, and telephone networks are defined dynamically as rapid edge streams on a massive domain of nodes. In these rapid and massive graph streams, it is often not possible to estimate the frequency of individual items (e.g., edges, nodes) with complete accuracy. Sketch-based stream summaries such as Count-Min can preserve frequency information of high-frequency items with a reasonable accuracy. However, these sketches lose the underlying graph structure unless one keeps information about start and end nodes of all edges, which is prohibitively expensive. For example, the existing methods can identify the high-frequency nodes and edges, but they are unable to answer more complex structural queries such as reachability defined by high-frequency edges. To this end, we designed a 3-dimensional sketch, gMatrix that summarizes massive graph streams in real-time with *theoretical performance guarantees*, while also retaining information about the structural behavior of the underlying graph dataset (**SNAM17**). Our experimental results using large-scale graph streams attest that gMatrix is capable of answering both frequency-based and structural queries with high accuracy and efficiency.

Uncertain Graphs Processing

Uncertain graphs, i.e., graphs whose edges are assigned a probability of existence, have recently attracted a great deal of attention (**PVLDB15**), due to their rich expressiveness and given that

uncertainty is inherent in the data in a wide range of applications, including noisy measurements, inference and prediction models, and explicit manipulation, e.g., for privacy purposes.

• **Reliability Queries over Uncertain Graphs.** A fundamental problem on uncertain graphs is computing the *reliable set* – the set of all nodes that are reachable from a query set of nodes with probability larger than a given threshold. State-of-the-art techniques usually resort to sampling methods, which are inefficient in large graphs. In (**EDBT14**), we proposed *RQ-tree*, a novel *index* for efficiently estimating reliability queries, which is based on a *hierarchical clustering* of the nodes, and restricts the expensive sampling process only inside a small neighborhood of the query node.

Indeed many approaches and problem variants for reliability estimation have been considered in the literature, majority of them assuming that edge-existence probabilities are fixed. In real-world graphs, edge probabilities typically depend on external conditions. In metabolic networks, a protein can be converted into another protein with some probability depending on the presence of certain enzymes. In social influence networks, the probability that a tweet of some user will be re-tweeted by her followers depends on whether the tweet contains specific hashtags. In our recent (**TKDE18**) paper, we overcome this limitation and focus on conditional reliability, that is, assessing reliability when edge-existence probabilities depend on a set of conditions. In particular, we study the novel problem of determining the top-k conditions that maximize the reliability between two nodes. We deeply characterize our problem and show that, even employing polynomial-time reliability-estimation methods, it is **NP**-hard, does not admit any **PTAS**, and the underlying objective function is non-submodular. We then devise a practical method that targets both accuracy and efficiency. An extensive empirical evaluation on several large, real-life graphs demonstrates effectiveness and scalability of our methods.

• **Influence Maximization in Social Networks.** A central characteristic of social networks is that they facilitate rapid dissemination of information between large groups of individuals. The classic influence maximization problem identifies the top-k seed users in a social network such that the expected number of influenced users in the network, starting from those seed users and following some influence cascading model, is maximized.

In (**ICDE16**), we investigated the novel problem of *revenue maximization of a social network host* that sells viral marketing campaigns to multiple client campaigners. We proved that the objective function is **NP**-hard, and neither monotonic, nor sub-modular. We next developed approximate algorithms with performance guarantees under additional constraints.

In our recent work (**SIGMOD18**), we studied the novel problem of jointly finding the top-k seed users and the top-r relevant topics for targeted influence maximization in a social network. Our solution is useful for *feature engineering in influence maximization*. In earlier works, we proposed a novel model for influence diffusion (**SDM11**), and investigated the novel problem of maximizing the time-discounted influence spread (**CIKM16**). We explored *influence graph embedding* in (**ICDE18**).

Distributed Graph Processing Systems

Graphs with millions of nodes and billions of edges are ubiquitous to represent highly interconnected structures. To support online search and query services (possibly from many clients) with low latency and high throughput, data centers and cloud operators consider scale-out solutions, in which the graph and its data are partitioned horizontally across cheap commodity servers. In (**SIGMOD12**), we developed *complementary graph partitioning*, which enables processing h-hop queries ($h = 2, 3$) almost locally and without much network I/O. To bypass the problem of dynamically updating the

existing graph partitions (due to new workloads and graph updates) and for improving scalability, in our recent work (**ATC18**), we *decoupled graph storage from query processors*, and developed *smart query routing strategies* with graph landmarks and embedding. Since a query processor is no longer assigned any fixed part of the graph, it is equally capable of handling any request, thus facilitating load balancing and fault tolerance. Moreover, due to our smart routing strategies, query processors can effectively leverage their cache, reducing the impact of how the graph is partitioned across storage servers. Our experiments with several real-world, large graphs demonstrate that the proposed framework gRouting, even with simple hash partitioning, achieves up to an order of magnitude better query throughput compared to existing distributed graph systems that employ expensive graph partitioning and re-partitioning strategies.

We surveyed state-of-the-art *vertex-centric graph processing systems* in (**PVLDB14**), and analyzed their efficiency bottlenecks in (**EDBT17**).

Complex Graph Mining

My research has spanned several areas of complex graph mining; for example, given an intrusion network, how can we efficiently find a set of intrusions that happen closely together (**CIKM12**)? What graph features one should extract in order to build an accurate and efficient classifier over malware callgraphs (**SIGMOD10**)? For crowdsourced entity resolution (ER) with human errors, how to select the next batch of crowdsourcing questions such that it increases the ER accuracy as much as possible, at the expenses of as few next crowdsourcing questions as possible (**CIKM17**; **PVLDB18**)? In this context, I shall discuss two of my works as follows.

• **Proximity Pattern Mining.** Mining of graph patterns in large information and social networks is critical to a variety of applications such as malware detection and biological module discovery. However, frequent subgraphs are often ineffective at capturing associations existing in these applications due to the complexity of isomorphism testing and inelastic pattern definition. In (**SIGMOD10**), we introduced the *proximity pattern*, which is defined as a set of labels that co-occur frequently in neighborhoods. Proximity patterns relax the rigid structure constraints of frequent subgraphs, while also introducing connectivity to frequent itemsets. Empirical results on real-life social and intrusion networks show that our technique not only finds interesting patterns that are ignored by existing approaches, but also achieves high performance in large-scale graphs.

• **Crowdsourced Entity Resolution with Human Errors.** Crowdsourcing is becoming increasingly important in entity resolution tasks due to their inherent complexity such as clustering of images and natural language processing. Nevertheless, human workers can make mistakes due to lack of domain expertise or seriousness, ambiguity, or even due to malicious intents. We mitigate the above challenges by considering an uncertain graph model, where the edge probability between two records A and B denotes the ratio of crowd workers who voted YES on the question if A and B are same entity. To reflect independence across different crowdsourcing tasks, we apply the notion of possible worlds, and develop parameter-free algorithms for both next crowdsourcing and entity resolution tasks. In particular, *for next crowdsourcing, we identify the record pair that maximally increases the reliability of the current clustering.* Since reliability takes into account the connected-ness inside and across all clusters, this metric is more effective in deciding next questions, in comparison with state-of-the-art works, which consider local features, such as individual paths (e.g., MinMax), nodes (e.g., PC-Pivot), or the set of either positive or negative edges (e.g., DENSE) to select next crowdsourcing questions. Based on detailed empirical analysis over real-world datasets, we find that our proposed solution, PERC (probabilistic entity resolution with imperfect crowd) improves the

quality by 15% and reduces the overall cost by 50% for the crowdsourcing-based entity resolution (**CIKM17**; **PVLDB18**).

Other Big-Data Management

I also worked in the broader area of big-data querying and introduced the *scalar-product query* (**SIGMOD14**), which is a parameterized query with multiple unknown query parameters. Scalar product queries naturally arise in a wide range of data-analytic applications including evaluation of complex SQL functions, time series prediction, scientific simulation, active learning, finding moving object intersection, top-k queries with ranking function, half-space range searching, nearest neighbor queries, and linear constraint queries. We proposed a *lightweight, yet scalable, dynamic, and generalized indexing scheme*, called the *Planar index*, for answering scalar product queries efficiently and in an accurate manner. We empirically demonstrate the usefulness of planar index in moving object intersection finding under various complex motions. For example, when the objects are moving in a circular motion or with an acceleration (e.g., air turbulence, airplanes), none of the existing spatio-temporal indexes work; whereas our proposed planar index is up to 75 times faster in finding their intersections as compared to an expensive sequential scan over all moving object pairs.

In the domain of data streams, pre-filtering and early-aggregation of high-frequency items from a skewed stream usually improves the efficiency and accuracy of stream processing. We investigated the impact of such pre-filtering strategy over the Count-Min sketch in (**SIGMOD16**). The main challenge here is to dynamically identify the high-frequency items with the incoming data stream and without any apriori knowledge about the underlying data distribution. We designed an adaptive algorithm that dynamically exchanges data items between the filter and the count-min sketch data structure; and hence, one always stores and early-aggregates in the filter the most frequent items considering up to the current input stream. We also explored how one can further improve the efficiency of such pre-filtering strategy using modern hardware and pipeline parallelism techniques.

References

ACM-SIGMOD-Blog17 A. Khan and Y. Wu. *Graph Pattern Matching Queries – Approximation and User-friendliness*. ACM SIGMOD Blog. 2017.

ATC18 A. Khan, G. Segovia, and D. Kossmann. "On Smart Query Routing: For Distributed Graph Querying with Decoupled Storage". In: *USENIX ATC*. 2018.

CIKM12 N. Li, X. Yan, Z. Wen, and A. Khan. "Density Index and Proximity Search in Large Graphs". In: *CIKM*. 2012.

CIKM16 A. Khan. "Towards Time-Discounted Influence Maximization". In: *CIKM*. 2016.

CIKM17 V. K. Yalavarthi, X. Ke, and A. Khan. "Select Your Questions Wisely: For Entity Resolution With Crowd Errors". In: *CIKM*. 2017.

EDBT14 A. Khan, F. Bonchi, A. Gionis, and F. Gullo. "RQtree: an Index for Reliability Queries". In: *EDBT*. 2014.

EDBT17 A. Khan. "Vertex-Centric Graph Processing: Good, Bad, and the Ugly". In: *EDBT*. 2017.

ICDE12 A. Khan, Y. Wu, and X. Yan. "Emerging Graph Queries in Linked Data". In: *ICDE*. 2012.

ICDE16 A. Khan, B. Zehnder, and D. Kossmann. "Revenue Maximization by Viral Marketing: A Social Network Host's Perspective". In: *ICDE*. 2016.

ICDE18 S. Feng, G. Cong, A. Khan, X. Li, Y. Liu, and Y. M. Chee. "Inf2vec: Latent Representation Model for Social Influence Embedding". In: *ICDE*. 2018.

PVLDB13 A. Khan, Y. Wu, C. Aggarwal, and X. Yan. "NeMa: Fast Graph Search with Label Similarity". In: *Proc. of the VLDB Endowment* 6.3 (2013), pp. 181 –192.

PVLDB14 A. Khan and S. Elnikety. "Systems for Big-Graphs". In: *Proc. of the VLDB Endowment* 7.13 (2014), pp. 1709–1710.

PVLDB15 A. Khan and L. Chen. "On Uncertain Graphs Modeling and Queries". In: *Proc. of the VLDB Endowment* 8.12 (2015), pp. 2042–2053.

PVLDB18 X. Ke, M. Teo, A. Khan, and V. K. Yalavarthi. "A Demonstration of PERC: Probabilistic Entity Resolution With Crowd Errors". In: *Proc. of the VLDB Endowment* (2018).

SDM11 C. Aggarwal, A. Khan, and X. Yan. "On Flow Authority Discovery in Social Networks". In: *SDM*. 2011.

SIGMOD10 A. Khan, X. Yan, and K.-L. Wu. "Towards Proximity Pattern Mining in Large Graphs". In: *SIGMOD*. 2010.

SIGMOD11 A. Khan, N. Li, X. Yan, Z. Guan, S. Chakraborty, and S. Tao. "Neighborhood Based Fast Graph Search in Large Networks". In: *SIGMOD*. 2011.

SIGMOD12 S. Yang, B. Zong, X. Yan, and A. Khan. "Towards Effective Partition Management for Large Graphs". In: *SIGMOD*. 2012.

SIGMOD14 A. Khan, P. Yanki, B. Dimcheva, and D. Kossmann. "Towards Indexing Functions: Answering Scalar Product Queries". In: *SIGMOD*. 2014.

SIGMOD16 P. Roy, A. Khan, and G. Alonso. "Augmented Sketch: Faster and More Accurate Stream Processing". In: *SIGMOD*. 2016.

SIGMOD18 X. Ke, A. Khan, and G. Cong. "Finding Seeds and Relevant Tags Jointly: For Targeted Influence Maximization in Social Networks". In: *SIGMOD*. 2018.

SNAM17 A. Khan and C. Aggarwal. "Toward Query-Friendly Compression of Rapid Graph Streams". In: *Springer Social Network Analysis and Mining* 7.1 (2017), 23:1–23:19.

TKDE15 N. Jayaram, A. Khan, C. Li, X. Yan, and R. Elmasri. "Querying Knowledge Graphs by Example Entity Tuples". In: *IEEE Transactions on Knowledge and Data Engineering* 27.10 (2015), pp. 2797–2811.

TKDE18 A. Khan, F. Bonchi, F. Gullo, and A. Nufer. "Conditional Reliability in Uncertain Graphs". In: *IEEE Transactions on Knowledge and Data Engineering* (2018).