Curriculum Vitae

Karam Gouda

• General information

– **Current Position:** Professor & Dean, Faculty of Computers & Artificial Intelligence, Benha University, Egypt.

– **Citizenship:** Egyptian.

– **Name:** Karam Abdelghany Abdelrahman Gouda.

– **Website:** http://bu.edu.eg/staff/karamabdelrahman7

– **Google Scholar:** http://scholar.google.com/citations?user=FwX7Ud8AAAAJ&hl=en

– **Scopus:** https://www.scopus.com/authid/detail.uri?authorId=9337580200

– **Contact Information:** +201064896417, karam.gouda@fci.bu.edu.eg – **Education:**

1. Kyushu University, Fukuoka, Japan.
   Ph.D. Computer Science, March 2002.
   Subject Descriptors: Data Mining.
   Thesis Title: *On Mining Frequent Patterns and Automatic Knowledge Discovery in Very Large Databases.*
2. Benha University, Benha, Egypt.
   M.Sc. Mathematics (Computer Science), March 1995.
   Subject Descriptors: Computer Algorithms (Approximation Algorithms). **Thesis Title:** *Coping With Combinatorial Problems.*
3. Benha University, Benha, Egypt.B.Sc. Mathematics, July 1989.

– **Language skills:**

∗ Arabic: Mother language.

∗ English: Very Good (speaking and writing) –

Work Experience:

1. **Professor:** Sept. 2017 – present.
   Information Systems Department, Benha University, Egypt.
2. **Researcher:** Jan. 2016 – Jul. 2018.
   Computer & Decision Engineering Dept., Universit´e Libre de Bruxelles, Belgium.
3. **Associate Professor:** Jul. 2012 – Sept. 2017
   Information Systems Department, Benha University, Egypt.
4. **Assistant Professor:** Sept. 2010 – Jul. 2012.
   Information Systems Department, Benha University, Egypt.
5. **Researcher:** Dec. 2007 – June. 2009.
   Database Lab., Kyungpook National University, Daegu, South Korea.
6. **Assistant Professor:** Jun 2002 – Sept. 2010.
   Department of Mathematical Sciences, Faculty of Science, Benha University, Egypt.
7. **Lecturer:** Jun 1995 - May 2002.(May 1997 – April 2002 Ph.D. study in Japan) Department of Mathematical Sciences, Faculty of Science, Benha University, Egypt.
8. **Assistant Lecturer:** October 1989 – May 1995
   Department of Mathematical Sciences, Faculty of Science, Benha University, Egypt.

• **Scholarships & Fellowships:**

- Egypt Government grant to study Ph.D in JAPAN: 12/05/1997 – 05/04/2002
- FY2020 JSPS Invitational Fellowships for Research in Japan (Long-term)

• Administrative and other duties (since 2010)

- **Dean:** Faculty of Computers & Artificial Intelligence. Benha University. Jun. 2021 - present.
- **Vice Dean:** Postgraduate Studies and Researches' Affairs. Faculty of Computers & Artificial Intelligence. Benha University. Oct. 2020 - Jun. 2021.
- **Vice Dean:** Environmental Affairs and Community Services. Faculty of Computers & Artificial Intelligence. Benha University. Sept. 2012 – Jan. 2016, Jul. 2018 Sept. 2020.
- **Board member:** Undergraduate/postgraduate courses and program development committee. 2010 – Jan. 2016, Jul. 2018 - present.
- **Member:** Student Projects evaluating committee. 2010 – 2015.
- **Coordinator & Member:** The preparatory team for the accreditation of the faculty of computers & Artificial Intelligence, Benha University. 2013 – Jan. 2016

• Reference Information

1. Mohammed Javeed Zaki, Professor
   Department of Computer Science, Rensselaer Polytechnic Institute, NY, USA.
   Tel: (518) 276-6340.
   Fax: (518) 276-4033.
   Email: zaki@cs.rpi.edu
2. Yoshiharu Ishikawa, Professor
   Graduate School of Informatics Nagoya University, Furo-cho, Chikusa-ku, Nagoya 464-8601, Japan.
   Tel: +81-52-789-3306.
   Fax: +81-52-789-3306.
   Email: ishikawa@i.nagoya-u.ac.jp
3. Toon Calders, Professor
   Department of Mathematics - Computer Sciences, University of Antwerp, Belgium.
   Email: toon.calders@uantwerp.be
4. Maher Zayed, Professor
   Department of Mathematical Sciences,
   Faculty of Science, Benha University, Benha, EGYPT.
   Email: maherzayed@hotmail.com Tel: 00201004976574

• Professional development

– Professional Activities Reviewer:

1. Engineering applications of Artificial Intelligence: IF:ă4.201 Elsevier J.
2. Information Processing & Management : IF:ă4.787 Elsevier J.
3. Data Mining & Knowledge Discovery : IF:4.431 Springer Journal
4. Very Large Databases : IF:2.64 Springer Journal
5. ACM Transactions on Internet Technology : IF:ă3.75 ACM Journal

Invited International Lectures:

1. **HUNGARY** 2024: University of Pannonia, Erasmus+ (Staff Mobility for teaching prog.).

2. **FRANCE** 2020: GRYEC, Université de Caen, Dépt. Mathématiques et informatique.
3. **CHINA** 2018: One Thousand Talent Competition & Ningbo University
4. **ESTONIA** 2013: University of Tartu, Computer Science Inst. http://www.uttv.ee/naita?id=1

– Training Competency

1. Workshop & training 2019: **"The Role of higher Education in Providing the Relevant Skills Needed for the digital Age (Life Skills, Digital Skills & Leadership Skills."** Arab Academy of Science, Technology & Maritime Transport (AASTMT), Alexandria, Egypt.
2. **Big Data Analysis training course:** Del EMC^2 Egypt, Nov. 2015.
3. **Cooperative Learning training course,** By Prof. Peter Saunders, Oregon State Universy Feb. 2006.
4. **Students Evaluation Systems training course,** By Prof. Peter Saunders, Oregon State Universy Feb. 2006.

- **Supervision**: Many M. Sc. & Ph.D. students since 2003.
- **Skills:**

1. **Programming Languages:** C/C++, perl, XML, HTML.
2. **Operating Systems:** UNIX, LINUX, XP.

• Attended Conferences

1. **INFOS 2012, ICCTA 2022, 2023,** EGYPT
2. **SISAP 2016**: October 2016, Tokyo, JAPAN.
3. **ICDE 2016**: May 2016, Helsinki, FINLAND.
4. **EDBT 2013**: March. 2013, Genoa, ITALY.
5. **ICDM 2007**: October 2007, Omaha, Nebraska, USA.
6. **KDD 2003**: Aug. 2003, Washington D.C., USA.
7. **ICDM 2001**: December 2001, San Jose, California, USA.
8. **EIS 2000**: June 2000, University of Paisley, Scotland, UNITED KINGDOM.
9. **IEEE-SMC 1999**: October 1999, Tokyo, JAPAN.
10. **IEEE-SMC 1998**: October 1998, San Diego, USA.
11. **DEWS 1998**: March 1998, Gunma University, JAPAN.

• Significant Publications

– Journals

1. **Karam Gouda** and Mosab hassaan. "A Novel Edge-centric Approach for Graph Edit Similarity Computation." *Inf. Syst.* 80: 91-106 (2019)
2. **Karam Gouda**, Mona Arafa and Toon Calders. "A novel hierarchical-based framework for upper-bound computation of graph edit distance." Pattern Recognition 80 (2018) 210–224.
3. **Karam Gouda** and Metwally Rashad. "An Efficient String Edit Similarity Join Algorithm." *Computing & Informatics*, 36(3):683-704 (2017)

4. **Karam Gouda** and Mona Arafa. ”An improved global lower bound for graph edit similarity Search.” *Pattern Recognition Letters* 58, 8-14 (2015).
5. **Karam Gouda**, Mosab hassaan, and Mohammed J. Zaki. ”Prism: An Effective Approach for Frequent Sequence Mining via Prime-Block Encoding.” *Journal of Computer and Systems Sciences*, 76(1): 88-102 (2010).
6. **Karam Gouda** and Mohammed Javeed Zaki, ”GenMax: An Efficient Algorithm for Mining Maximal Frequent Itemsets.” *Data Mining and Knowledge Discovery: An International Journal*, 11(3): 223-242 (2005).

– **Refereed Conferences**

1. **Karam Gouda**, Mona Arafa and Toon Calders. ”BFST ED: A Novel Upper Bound Computation Framework for the Graph Edit Distance.” In: *SISAP*, pp. 3-19 (2016).
2. **Karam Gouda** and Mosab hassaan. ”CSI GED: An efficient approach for graph edit similarity computation.” In: *ICDE*, pp. 265276 (2016).
3. **Karam Gouda** and Mosab hassaan. ”Compressed Feature-based Filtering and Verification Approach for Subgraph Search.” In: *EDBT*, pp. 287-298 (2013).
4. **Karam Gouda**, Mosab hassaan, and Mohammed J. Zaki. ”PRISM: A PrimeEncoding Approach for Frequent Sequence Mining.” In: *ICDM 2007*, pp. 487-
   492.
5. M. Javeed Zaki and **Karam Gouda**. ”Fast Vertical Mining Using Diffsets.” In: *KDD*, pp. 326-335 (2003).
6. **Karam Gouda** and M. Javeed Zaki. ”Efficiently Mining Maximal Frequent Itemsets.” In: *ICDM*, pp. 163-170 (2001).

• Research and development work

– **Research Areas:** My research interest is in the algorithmic aspects of:

1. **Data Mining**
2. **Graph & String Data query Processing.**
3. **Security Monitoring in Data streams.**

– **Research Keywords:**

Graph matching, string matching, sequence mining, itemset mining, data encoding, data representation, indexing, verification methods, edit distance, lower and upper bounds, edit Similarity search and join, graph databases, string databases, patternbased security monitoring, data streams.

– **Application Domains:**

Bio-informatics, Chem-informatics, Pattern Recognition, Information Retrieval, Social Networks, Semantic Web, Web Mining, Text Mining, Software Engineering and Information Networks, Information Security.

– **Scientific Praises and Acknowledgement:** ”Database Systems Research in the Arab World: A Tradition that Spans Decades.” *Communications of the ACM* Vol. 64 No. 4, 2021

– **Developed Software:**

1. GenMax: Maximal Frequent Itemsets Miner.
2. dEclat: Frequent Itemset Miner in Dense Data.

3. PRISM: Frequent Sequences Miner.
4. pEclat: Frequent Itemset Miner in Sparse Data.
5. PathIndex: Graph Indexing and Sub-graph Query Processor.
6. PreJoin: String Edit Similarity Join Algorithm.
7. CSI GED: Exact Graph Edit Distance Computation Approach.
8. BFST ED: Upper-bound Computation Approach of Graph Edit Distance.

– **General Research objectives:**

1. Exploring new issues in data mining and management.
2. Extending my research on new computation platforms.
3. Exploring new research areas related to my research.

– **Future Research objectives:** In the coming period of time I will deeply investigate two main problems in the area of data analysis and management. These are: 1) the problem of discovering succinct summaries from (complex) event sequences and 2) the problem of (edit) similarity search in graph databases. The huge wealth of (complex) event sequences such as network sensor data, business events, annotated text, and transactional records have motivated the recent interest in discovering summaries to gain useful insights into the data. A summary is a non-redundant, relevant and informative set of patterns that describe the data well. In many application domains, these summaries are used as models for data prediction, data encoding, and anomaly detection. The motivation of discovering succinct summaries comes from the *information overloading* problem which is caused by the huge amount of frequent, closed or maximal patterns. Clearly, returning collections of such magnitude is useless, as they cannot be used or inspected in any meaningful way. Our objective is to develop efficient and scalable algorithms for discovering succinct summaries.

  Graph data are also abundant. Graph similarity search, graph clustering and classification are common tasks which require efficient computation of graph edit distance. In the next period of time we extend our current research on graph edit distance. In particular, we target new graph partitioning methodologies to fully exploiting the parallel processing capabilities of contemporary and commodity multi-core hardware to break down the complexity of graph edit distance on large graphs. Although the parallel processing of graph edit distance is not an easy task, as it poses challenges such as recovering the distance accuracy achieved on local graph parts when joining these distances, it in turn allows us to solve the edit similarity search, clustering, and classification problems on big graph data efficiently.

  Related Literature

1. Roel Bertens, Jilles Vreeken and Arno Siebes. ”Keeping it Short and Simple: Summarising Complex Event Sequences with Multivariate Patterns.” In: KDD 2016.
2. Apratim Bhattacharyya, and Jilles Vreeken. ”Efficiently Summarizing EventSequences with Rich Interleaving Patterns.” In: SDM 2017.
3. A. Ibrahim, S. Sastry and P. Sastry. ”Discovering compressing serial episodesfrom event sequences” In: Knowl Inf Syst, pp. 405432 (2016).
4. K. Nakagawa, S. Suzumura, M. Karasuyama, K. Tsuda, and I. Takeuchi. ”SafePattern Pruning: An Efficient Approach for Predictive Pattern Mining.” In: KDD 2016.

– **Previous Research Projects:**

  * **SPICES: Scalable Processing and mIning for SEcurity-AnalyticS (project at ULB-Bruxelles)**

The objective of this project is to develop a powerful software technology that allows organizations (companies) to detect and describe potential security problems in their systems based on certain sequences or certain collections of business events. These can be any type of event (messages sent, RFID scans,...) for which a digital avatar exits. Once detected, certain accompanying business actions need to be executed (message send, SMS send, ...). SPICES proposes to conceive these sequences and collections as instances of abstract patterns that can be described in a complex event processing (or CEP) language. Such a language allows for a declarative description of the patterns without having to resort to low level implementation technology. Given the abstract description of the patterns, detecting and monitoring security problems then boils down to an complex event processing problem. In brief, complex event processing is relevant for security process monitoring since it provides the facility for the online detection of pre-coded opportunities, security problems and attacks. The overal goal of SPICES is to design an open reusable platform centred around the notion of complex event processing that is specifically targeted towards security process monitoring.

I am one of the Data mining (DM) team in this project. We are responsible for developing the mining technology that allows to extract the patterns that will trigger the notifications from existing (big) data bases of events. Since security process monitoring is a continuous online activity, the team is also responsible of the online adaptation of the security patterns, i.e., dealing with the issue of concept drift.

Publications

1. **Karam Gouda** and Toon Calders. ”Pattern-based behavioral modelling for anomaly detection in event sequences”, Technical Report, ULB, Belgium, 2017.
2. **Karam Gouda** and Toon Calders. ”Edit distance-based Concept Drift”, Technical Report, ULB, Belgium, 2017.

* **Graph Similarity Search and Join.**

  One of the most important research topics in graph data processing is graph similarity search and join. It is an essential operation in many application areas including Pattern Recognition, Bio-informatics, Chem-informatics, Social Networks, Semantic Web, Software Engineering, etc. Recently, There has been a great interest in graph similarity search joins with edit distance constraint. Because the edit similarity measure is applicable to all types of data graphs and captures precisely structural differences, it can be used in applications such as graph classification and graph clustering. More interestingly, the accompanying edit sequence provides an explanation for an edit distance value and this is a very valuable feature for the user.

  Existing solutions to graph similarity join problem adopt the filter-and-verify strategy to speed up processing, where lower and upper bounds of graph edit distance are employed as pruning and validation rules in this process. Currently, we are interested in developing efficient methods to tackle the performance issues of the filter-and-verify approaches, such as developing new easy-to-compute and tight lower and upper bounds. Also, new efficient verification methods are the target of our research.

Publications

1. **Karam Gouda** and Mosab hassaan. ”A Novel Edge-centric Approach for Graph Edit Similarity Computation.” *Information systems* 80, 91-106 (2019).

2. **Karam Gouda**, Mona Arafa and Toon Calders. ”A novel hierarchicalbased framework for upper-bound computation of graph edit distance.” *Pattern Recognition* 80 (2018) 210224.
3. **Karam Gouda**, Mona Arafa and Toon Calders. ”BFST ED: A Novel Upper Bound Computation Framework for the Graph Edit Distance.” *In: SISAP 2016, pp.* 3-19 (2016).
4. **Karam Gouda** and Mosab hassaan. ”CSI GED: An efficient approach for graph edit similarity computation.”
In: ICDE 2016, pp. 265276 (2016).
5. **Karam Gouda** and Mona Arafa. ”An improved global lower bound for graph edit similarity Search.”
Pattern Recognition Letters 58, 814 (2015).

∗ **Efficient Management of Sub-structure Search Over Large Graph Databases.**
The problem of substructure search over graph data has recently drawn significant research interest due to its importance in many application areas such as in Bio-informatics, Chem-informatics, Social Network, Software Engineering, World Wide Web, Pattern Recognition, etc. For example, in drug design, efficient techniques are required to query and analyze huge data sets of chemical molecules thus shortening the discovery cycle in drug design and other scientific activities. Managing substructure search is efficiently conducted through the framework of filtering-and-verification, i.e., Indexing. Thus, building compact indices with great pruning capability is the main objective of current research. Feature-based and summary-based indexing techniques are two main streams of methods that are widely used. Though both directions have shown good performance, they suffer from some drawbacks which hinder any of them to be the best choice for effective sub-structure query processing. With feature-based techniques, filtering accuracy is getting worse when graph sizes are increasing. Moreover, building the index is time-consuming due to the overhead of the involved data mining process; and therefore they are poor in handling updates. On the other hand, the major limitation of the summary-based techniques is the high cost of filtering phase. The main goal of our research is to scale up the current indexing approaches either by importing the best of both world, i.e., summary-based and feature-based approaches, into one single indexing approach or by importing one of the successful approaches working in closer areas to be developed in our domain. In this project, in particular, we are going to study each of the following:

1. Scaling up the summary-based approach: Our analysis of the closure treeindex as a summary-based representative has concluded that high cost of filtering is due to large summaries size (graphs closure size). It is difficult to build compact closures on graphs of large size because compactness requires finding optimal graph mapping which is NP-Complete. Summaries of smaller size could be obtained if we partition the data graphs and then cluster similar parts. In this project we are going to test how database partitioning techniques that decompose each data graph into smaller parts or that divide large size data sets into smaller disjoint ones could affect closures size and its construction time. However, the main challenge is the recovery process in order to get the final answer set.
2. Developing a new approach: Gram-based indexing has been successfullyworking in string searching for decades. The concept has also been developed for searching problems on hierarchal Data. We believe that, transforming each data graph into overlapping windows of variable length has the benefit of first preserving the graph structural information and second the possibility of transforming each window to a string and then benefitting form the string processing technology.

Thus, in this project, we will test the possibility of applying a similar concept to the graph searching problems. Variable length grams is recently used for approximate queries on string databases .

3. Scaling up the feature-based approach: Building a new feature-based indexwhich combines paths, trees and subgraphs as features. The idea of combining different types of features was first applied in (Tree +delta). Tree + delta uses trees and on demand add discriminative subgraphs (discriminative means features having different pruning power). we suggest that an easy to extract path features could also be discriminative because pathes alone as features was applied in GraphGrep.

Publications

1. Mosab hassaan and **Karam Gouda**. ”New Subgraph Isomorphism Algorithms: Vertex versus Path-at-a-time Matching” *arXiv preprint arXiv:*1904.08819 (2019).
2. **Karam Gouda** and Mosab hassaan. ”Compressed Feature-based Filtering and Verification Approach for Subgraph Search.” *In: EDBT 2013,* 287-298.
3. **Karam Gouda** and Mosab hassaan. ”Fast Algorithm for Subgraph search problem.” *The 8th international conference on informatics and systems (INFOS 12).*

∗ **String Data Processing (Similarity Join)**

One of the most important research topics in string processing is string similarity join. It is an essential operation in many applications, such as data integration and cleaning, near duplicate object detection and elimination, and collaborative filtering. Recently, string similarity joins with edit distance constraint (simply edit string similarity join) has been extensively studied. There exist many algorithms to support edit string similarity join. These algorithms follows the filter and verification paradigm, where q-gram inverted indexes are used to filter many of the unpromising string pairs and generate candidate pairs; then these candidates are verified to output the final result. Other algorithms adopt a trie-based framework. A trie-based framework is verification-free, that is, it generates all similar string pairs without the verification step, and uses a trie structure to share prefixes and utilizes prefix pruning to improve the performance.

Algorithms following the filter-and-verification paradigm have the following disadvantages. Firstly, they are inefficient for the data sets with short strings (the average string length is no larger than 30), since they cannot select high-quality signatures (q-grams) for short strings and thus they may generate a large number of candidate pairs which need to be further verified. Secondly, they cannot support dynamic update of data sets. The dynamic update may change the weights of signatures. Thus the two methods need to reselect signatures, rebuild indexes, and rerun their algorithms from scratch. Thirdly, they involve large index sizes as there could be large numbers of signatures. Existing Triebased Join algorithms, on the other hand, have shown that Trie Indexing is more suitable for Similarity Join on short strings. The main problem with current approaches is that they generate and maintain lots of candidate prefixes called active nodes which need to be further removed. With large edit distance, the number of active nodes becomes quite large.

Our research is focused on devising new algorithms which overcome the performance challenges intimate to current approaches. Recently, we proposed a new Trie-based Join algorithm called PreJoin, which improves upon current Trie-based Join methods. It efficiently finds all similar string pairs using a new active-node set

generation method, and a dynamic preorder traversal of the Trie index. Experiments show that PreJoin is highly efficient for processing short as well as long strings, and outperforms the state-of-the-art Trie-based Join approaches by a factor five. Currently, we are interested in integrating the two previous paradigms into a new string similarity join paradigm which overcomes performance bottleneck.

Publications

1. **Karam Gouda** and Metwally Rashad. ”An Efficient String Edit Similarity Join Algorithm.”
Computing & Informatics 36(3):683-704 (2017).

2. **Karam Gouda** and Metwally Rashad. ”PreJoin: An Efficient Trie-based String Similarity Join Algorithm.”
The 8th international conference on informatics and systems (INFOS 12).

∗ **Frequent Patterns Mining**

Frequent Pattern Mining (FPM) in transaction databases is one of the most interesting data mining tasks. Research has been conducted for many years to solve this problem with the aim of developing scalable algorithms with respect to database size and the number of frequent patterns. Most of this research is classified into two approaches: horizontal and vertical. In the horizontal approach, researchers developed solutions using the horizontal data representation. In this representation, the database consists of a list of transactions, where each transaction has an identifier, called Tid, followed by a list of items in that transaction. In the vertical approach, researchers were using the vertical format. In the vertical database each item is associated with its corresponding *tidset*, the set of all transactions (or Tids) where it appears.

1. **Itemset pattern Mining: PhD Research**

During PhD. research We have been successful in developing the vertical approach to the itemset mining problem (a basic type of FPM). This success has been achieved in two different stages. Firstly, we characterized the limitations of the current vertical approach, which lie in both the excessive memory used to store intermediate tidsets which are required for testing a new candidate itemset pattern for frequency, and the excessive time used for long tidsets intersections. We developed a new vertical data representation, called *diffset*, to work in place of tidset. Diffsets are analytically and experimentally proved to be very short compared to their tidsets counterparts. Thus the doubly difficult issue of time and memory is addressed using this new data structure. Secondly, We developed a new vertical-based algorithm, called *GenMax*, for mining the exact set of maximal frequent itemset patterns. GenMax is an algorithm that utilizes a novel backtracking search, combined with the ability to utilize previously discovered patterns to compute the set of maximal itemset patterns efficiently. GenMax uses a number of optimizations to quickly prune away a large portion of the search space.

It uses a novel *progressive focusing* technique to eliminate non-maximal patterns. For fast frequency testing, GenMax uses two optimizations based on the diffset structure. These optimizations are called *diffsets propagation* and *diffsets all the way*. We showed using extensive set of experiments performed on both dense and sparse databases the superiority of the developed vertical

approach over the horizontal one in mining frequent, closed, and maximal itemset patterns.

2. **Sequence pattern Mining:**

After PhD. research, we continued in developing the vertical approach for the problem of pattern mining in sequential data. Our contribution was a novel in-memory data encoding technique called Prime-Encoding which generates compressed data structures to work with instead of tid-lists. An efficient sequence mining algorithm, called PRISM, is developed that benefits from prime structure. Via an extensive evaluation on both synthetic and real datasets, PRISM was shown to outperform popular sequence mining methods like SPADE, Pre fixSpan, and SPAM, by an order of magnitude or more.

Publications

(a) MJ Zaki and **Karam Gouda**. ”Fast vertical mining using diffsets.” *In: KDD 2003*.

(b) **Karam Gouda** and MJ Zaki. ”Efficiently mining maximal frequent itemsets.” *In: ICDM 2001.*

(c) **Karam Gouda** and MJ Zaki. ”Genmax: An efficient algorithm for mining maximal frequent itemsets.” *Data Mining and Knowledge Discovery 11 (3), 223-242.*

(d) **Karam Gouda**, Mosab Hassaan and MJ Zaki. ”Prism: A primalencoding approach for frequent sequence mining.” *In: ICDM 2007.*

(e) **Karam Gouda** and MJ Zaki. ”Prism: An effective approach for frequent sequence mining via prime-block encoding.” *Journal of Computer and System Sciences 76 (1), 88-102.*

(f) **Karam Gouda**, Mosab Hassaan. ”Efficiently Using Prime-Encoding for Mining Frequent Itemsets in Sparse Data.” *Computing & Informatics* 32 (5), 1079-1099 (2013).

∗ **Knowledge Discovery Process: PhD. Research**

The Knowledge Discovery in Databases (KDD) process comprises numerous steps with many decisions being made by human experts in order to achieve its ultimate goal of discovering new and effective knowledge. The great emphasis on human factor though increases the quality of discovered knowledge, it, on the other hand, slows down the KDD process and decreases the number of discovery applications. The only alternative is automated KDD. Automating KDD would produce many orders of magnitude speedup in the process, with the corresponding reduction in cost and increase in the number of viable applications. However, it is a big challenge to replace external intervention by automated search, while maintaining or expanding the scope of knowledge that can be acquired by the discoverer. The key issue is how to increase both autonomy and versatility of the KDD systems.

Relevant domain knowledge construction and incorporation in the KDD process is one of the most viable tasks of human agent. Full automation of the process amounts to full automation of this task.

Our research investigated the problem in details and proposed a computational framework, called DASS, to help in solving this problem. It also argued the difficulties which appear in this framework. We developed an algorithm based on DASS framework to show that construction and incorporation of relevant domain

knowledge can be integrated into one step. We also developed a relevance reasoning method for Horn clause languages.

Publications

1. **Karam Gouda** and Jingde Cheng. ”Using relevant reasoning to solve the relevancy problem in knowledge discovery in databases.” *In: SMC’98*.
2. **Karam Gouda** and Jingde Cheng and K. Ushijima. ”DASS: a discovery agent supporting system.” *In: SMC’99.*
3. **Karam Gouda**, T. Shoudai, and K. Ushijima. ”DASS: A Discovery Agent Supporting System,” *In: Proceeding of the international ICSC Symposia on Engineering of Intelligent Systems (EIS’2000)*, Scotland, UK, June 2000.