

# A Survey on Graph Reduction Methods and Applications

Mostafa Aref<sup>1</sup>, Ibrahim Fathy Moawad<sup>2</sup>, Marwa Mahmoud<sup>3</sup>

*Department of Computer Science,  
Faculty of Computer and Information Sciences,  
Ain-Shams University, Cairo, Egypt.*

<sup>1</sup>mostafa.m.aref@gmail.com

<sup>2</sup>ibrahim\_moawad@cis.asu.edu.eg

<sup>3</sup>marwa\_mahmoud100@hotmail.com

**Abstract:** *One of the important Natural Language Processing applications is Text Summarization, which helps users to manage the vast amount of information available, by condensing documents' content and extracting the most relevant facts or topics included. Text Summarization can be classified according to the type of summary: extractive, and abstractive. Extractive summary is the procedure of identifying important sections of the text and producing them verbatim while abstractive summary aims to produce important material in a new generalized form. One of recent approaches to achieve the abstractive summary is converting the original text to a graph, reducing this graph, and then converting the reduced graph into summarized text. In this paper, several approaches for graph reduction are presented. These approaches have different applications in different fields. Using a working example, the paper describes in progress research to apply graph reduction techniques in abstractive text summarization.*

**Key words:** *Natural Language Processing, Text Summarization, Graph Reduction*

## 1 INTRODUCTION

Informally, a graph is set of nodes, pairs of which might be connected by edges. In a wide array of disciplines, data can be intuitively cast into this format. For example, computer networks consist of routers/computers (nodes) and the links (edges) between them. Social networks consist of individuals and their interconnections (which could be business relationships or kinship or trust, etc). Protein interaction networks link proteins which must work together to perform some particular biological function. Ecological food webs link species with predator-prey relationships. In these and many other fields, graphs are seemingly ubiquitous [1]. The reduction [2], or simplification, of graph-based models is critical to the analysis, simulation and control design for systems arising in many diverse areas such as network routing, image processing, statistical learning, and in distributed control of networked dynamical systems, to name a few. The reduction is carried out through node and edge aggregation, where the simpler graph is representative of the original large graph.

The primary aim of our research is to systematically select and review published work and provide an overview of graph reduction theory, their methods, applications and focuses. This paper proceeds as follows. We provide a description of relevant reviewed papers and classify them into appropriate categories according to topics. Typically, we only provide the main ideas and approaches; the interested reader can read the relevant references for details. In all of these, we attempt to collate information from several fields of research. Our conclusions are presented in the last section.

## 2 SELECTED STUDIES

We identified five main categories for graph reduction applications: workflow management system, computer vision, networks, semantic graphs and other application.

### A. Workflow Management System

Workflow technology has been a new hotspot in the area of computer application since 1990. Nowadays workflow management systems are widely used in improving the effectivity and efficiency of business processes. The production workflows, a subclass of workflows, support well-defined procedures for repetitive processes and provide a means for automated coordination of activities that may span over several heterogeneous and mission-critical information systems of an organization [3]. Production workflow applications are built upon business processes that are generally quite complex and involve a large number of activities and associated coordination constraints. Using a generic process modeling language for workflows, we show how a structural specification may contain deadlock and lack of synchronization conflicts that could compromise the correct execution of workflows. It is essential that a process model is properly defined, analyzed, verified, and refined before being deployed in a workflows management system. Sadiq

and Orlowska in [3] presented an effective graph reduction algorithm that can detect the existence of structural conflicts in workflow graphs. The basic idea behind verification approach is to remove some nodes and/or transitions from the workflow graph  $G$ , and keeping the reduction process as long it can reduce the size of process graph. The algorithm reduces a workflow graph without structural conflicts to an empty graph. It means that the original process graph  $G$  before reduction does not contain any structural conflicts. Otherwise, it contains deadlock or lack of synchronization structural conflicts that would be clearly visible from the reduced graph. The algorithm in [4] is based on a set of graph reduction rules to identify the deadlock and lack of synchronization conflicts that could compromise the correct execution of a workflow. A complete set of graph reduction rules is presented to reduce the workflow graph. The graph reduction algorithm can remove all nodes from workflow graphs that are definitely correct. In [5] Lu and Liu proposed a combined graph-reduction and graph search algorithm which is called "CWRS" to verify workflow graphs. This algorithm reduces a cyclic workflow graph to an acyclic workflow graph, and then verifies this acyclic workflow graph. The computational complexity of our algorithm behaves a good performance. So it should be valuable in practical graph-based workflow verification.

### *B. Computer Vision*

Graph cuts had a growing impact in shape optimization. In particular, they are commonly used in applications of shape optimization such as image processing, computer vision and computer graphics. Solving problems with a large number of variables remains computationally expensive and requires a high memory usage since underlying graphs sometimes involve billion of nodes and even more edges. Examples for computer vision problems include segmentation, image restoration, dense stereo estimation and shape matching. An efficient algorithm for image segmentation using GraphCuts which can be used to efficiently solve labeling problems on high resolution images or resource-limited systems is presented in [6]. A Slim Graph is constructed by merging nodes that are connected by simple edges. A proof is given that the value of the maximum flow on the Slim Graph is equal to the maximum flow of the original graph. The nodes connected by a simple edge will have the same label in the final segmentation and can be merged into a single node. Thus the original graph is simplified to a Slim Graph without changing the energy-minimization problem and the value of the global minimum. It was shown that the proposed method required much less memory allowing segmentation of images of reasonable sizes even on mobile devices. A further reduction of computation time can be achieved by using parallel hardware architecture. In [7] Malgouyres and Lerne proposed a method to improve graph-cuts in this regards. A formal statement is given which expresses that a simple and local test performed on every node before its construction permits to avoid the construction of useless nodes for the graphs typically encountered in image processing and vision. A useless node is such that the value of the maximum Flow in the graph does not change when removing the node from the graph. Such a test therefore permits to limit the construction of the graph to a band of useful nodes surrounding the final cut. Energy-minimizing active contour models (snakes) have been proposed for solving many computer vision problems such as object segmentation, surface reconstruction, and object tracking. Energy-minimizing active contour models (snakes) have been proposed for solving many computer vision problems such as object segmentation, surface reconstruction, and object tracking [8]. Dynamic programming which allows natural enforcement of constraints is an effective method for computing the global minima of energy functions. However, this method is only limited to snake problems with one dimensional (1D) topology (i.e., a contour) and cannot handle problems with two-dimensional (2D) topology. Yan et al. [8] presented an algorithm to minimize the energy function associated with 2D snakes. 2D snakes represent 2D surfaces with connected deformable graphs controlled by vertices and edges. A set of reduction operations are defined and used to simplify the graph of the 2D snake into one single vertex while retaining the minimal energy of the snake.

### *C. Networks*

Dynamic, QoS-based routing received considerable attention in recent years, especially considering the difficulty in predicting Internet traffic patterns and the consequent impossibility to properly plan and dimension the network. The core of any QoS-based routing algorithm is a network status-dependent cost function that is used to find the optimal (or at least a suitable) route across the network by solving an optimization problem. Routing can be formalized as the problem of finding a suitable set of edges connecting two nodes in a directed graph. Casetti et al. [9] introduced a new approach to QoS routing, the approach can be summarized as NGR (Network Graph Reduction) i.e., a modification of the graph describing the network before the routing path is computed, in order to exclude from the path selection over-congested portions of the network. This solution leads to a class of two-step routing algorithms, where both steps are simple, hence allowing efficient implementation. Chekuri and Korula [10] gave two applications of a graph reduction step to connectivity and network design problems. This step preserves the global and local connectivity of the graph. The first, is a polylogarithmic approximation for the problem of packing element-disjoint Steiner forests in general graphs, and an  $O(1)$ -approximation in planar graphs. The second is a very short and intuitive proof of a spider-decomposition theorem

of Chuzhoy and Khanna [11] in the context of the single-sink  $k$ -vertex-connectivity problem. The results highlight the effectiveness of the element-connectivity reduction step.

#### *D. Semantic Graphs*

The increasing availability of online information has necessitated intensive research in the area of automatic text summarization within the Natural Language Processing (NLP) community. Automatic text summarization can be done using a semantic graph reducing technique. The reader thus obtains an overview of the content, without having to read through the text. In building a compact semantic graph, an important step is grouping similar concepts under the same label and connecting them to external repositories. Few papers address the problem of reduction and enhancement of semantic graphs. A new method [12] is proposed for reducing large directed graphs to simpler graphs with fewer nodes. The reduction is carried out through node and edge aggregation based on the maximum entropy principle. As a special case, this method applies to the Markov chain model-reduction problem, providing a soft-clustering approach that enables better aggregation of state-transition matrices than existing methods. Graph reduction is utilized for efficient and effective indexing and retrieval. In [13] the authors focus on the role of semantic graph in web page content visualization and the role of graph in displaying semantic relations. The semantic graph is generated in the form of subject, object and verb where subject and object are represented by nodes and verb defines the relationship between them. It also includes the overall process of a system in creating a smaller semantic graph from the given web page html document. More compact semantic graphs can be generated [14] by identifying the triplet elements that share the same meaning and can be therefore merged together under the same label. Additionally, linking semantic graph nodes to external resources, such as WordNet thus helping in better understanding the graph content. Tamil Document Summarization using sub graph [15] presents a method for extracting sentences from an individual document to serve as a document summary or a pre-cursor to creating a generic document abstract. Semantic features of the text are identified using Logical Form (LF) Parser. The semantic graph constructed using the LF parser is then used to select important parts of the document for summary generation. The selection of important sentences from the graph is based on three types of attributes, Linguistic attributes, Semantic Graph attributes and Document Discourse Structure attributes considered as features for learning. This rich set of features serves as input to Support Vector Machine (SVM) classifier which classifies the sentences as important or unimportant for inclusion into the summary. In [16] a novel approach is presented to create an abstractive summary for a single document using a rich semantic graph reducing technique. A model of heuristic rules is applied to reduce the graph by replacing, deleting, or consolidating the graph nodes using the WordNet relations.

#### *E. Other Applications*

Reduction is matching invariant, so it can be used as a speed-up in matching algorithms. Bartha and Kresz [17] presented a linear-time algorithm to shrink a graph  $G$  recursively along its 2-star subgraphs called redexes. The starting point was a greedy algorithm, which works in linear time only if  $G$  does not contain recursively incurring (implied) redexes. Implied redexes have been detected and reduced during a single bottom-up sweep of the depth-first tree of  $G$ , and the resulting graph was transferred to the greedy algorithm to construct the desired graph  $r(G)$ .

Pointer manipulation is notoriously dangerous in languages like C where there is nothing to prevent: the creation and dereferencing of dangling pointers; the dereferencing of nil pointers or structural changes that break the assumptions of a program, such as turning a list into a cycle. Graph reduction specifications (GRSs) are a powerful new method for specifying classes of pointer data structures (shapes). They cover important shapes, like various forms of balanced trees that cannot be handled by existing methods. Bakewell et al. [18] showed how to improve the safety of pointer programs by providing (1) means for programmers to specify pointer data structure shapes by graph reduction specifications (GRSs), which are the dual of graph grammars in that graphs in a language are reduced to an accepting graph rather than generated from a start graph, and (2) algorithms to check statically whether programs preserve the specified shapes.

Natural Language Understanding (NLU) technology is a fundamental component of dialog-based automatic speech understanding systems. Such systems are typically implemented on telephony platforms and are used to automate the communication process between humans and machines through natural speech. An important component of such systems is the semantic parser whose purpose is to recognize structures in the sentence, with the goal to facilitate meaning extraction. In [19] Huertu and Lubensb introduced a method to represent the semantic parser domain into a single directed graph showing the parser's labels and their immediate inter-relationships as they exist in the annotated development corpora. The authors describe how the graph representation method can be utilized in the reduction of the

complexity of the parser by identification and removal of nodes, edges and structures of the domain graph without major impact in the parser's accuracy.

Modern embedded systems typically consist of both hardware and software components. Hardware/software (HW/SW) partitioning Hardware/software (HW/SW) partitioning is one of the key processes in an embedded system. It is used to determine which system components are assigned to hardware and which are processed by software. The design's performance is directly determined by the results of HW/SW partitioning. In [20] Hui et al. presented the idea of graph reduction techniques for HW/SW partitioning. The task graph reduction technology, which is proposed for HW/SW partitioning, consists of the sub-graph searching algorithm and the Sub-graph evaluation and selecting algorithm. The purpose of this pre-process for the task graph is to find all the sub-graphs and to reduce each sub-graph to a single task node, which results in a new reduced task graph. The partitioning algorithm for the reduced task graph achieved improvement on partitioning speed and accuracy and also is able to make full use of the hardware area.

Goldman and Ngoko investigated the question of Service Response Time (SRT) prediction of a Web Service Composition using the graph reduction technique [21]. The authors propose a fast algorithm for graph reduction that uses less memory and perform fewer operations. The graph reduction approach proceeds in two phases. The first phase deals with the computation of a reduction order. The idea in such an order is to define an ordered set of decomposable subgraphs for the reduction. The second stage is the reduction with the defined order.

The bug localization techniques based on graph mining are successfully applied in a wide range of practical problems arising in software industry. One of the techniques for automated bug localization is usage of call graph. Since size of the call graph generated is quite large, in [22] a novel algorithm for call graph reduction has been proposed in order to use the respective call graphs for bug localization, the developed technique stores the parent information in the matrix and reduced at each level drastically. Information about each node is retained by using the call frequency by annotating each edge with a numerical weight. Similarly the algorithm used to reduced call graph has various advantages over traditional techniques.

Consider an agent who seeks to traverse the shortest path in a graph having random edge weights. If the agent has no information about the realizations of the edge weights, it should simply take the path of least average length (a deterministic optimization). Generalizations of this framework whereby the agent has access to a limited amount of side information about the edge weights ahead of choosing a path. Rinehart and Dahleh [23] presented A new graph reduction for analyzing the value of side information for shortest path optimization was presented this graph reduction that captures limited but significant information about the geometry of the graph's path polytope. The resulting reduction serves to concentrate side information to its ability to help the agent discern the shortest path in the graph, not simply determine its length.

### 3 A WORKING EXAMPLE

To show how the graph reduction approach works [16], a simulated case study called "Graduate students" is presented in details. Fig. 1 shows the input text, which consists of single paragraph talking about two graduate students (Angle Chris and John Michel). It consists of 7 sentences and contains 53 words. A rich semantic graph is created as shown in Fig. 2. The rich semantic graph nodes represent the instantiated objects of the domain ontology classes for the input text nouns and verbs. It contains 8 noun nodes representing the sentence subjects and objects, and 5 verb nodes (with gray background color) representing the sentence main verbs. By applying the graph reduction on the generated rich semantic graph from the input text, the reduced graph is shown in Fig. 3.

Finally, the abstractive summary is shown in Fig. 4. As shown in the summary, the "*Angle Chris and John Michel are graduate students.*" sentence was composed from both "Student 1" and "Student 2" nodes, and the "*They are specialized in Artificial Intelligence field.*" sentence was composed from the "Student 1", "Student 2", "Specialize 1", and "Field 3" nodes. The final summary consists of 4 sentences and contains 29 words. The final abstractive text represents about 50% of the original text.

Angle Chris is a graduate student. Mrs. Chris is specialized in Machine learning field. John Michel is a graduate student. He is specialized in Intelligent Agents field. During his study, Mr. Michel passed the preparatory courses. Angle Chris published two papers in international conferences. Also, John Michel published two

Figure 1: The original text of graduate students example

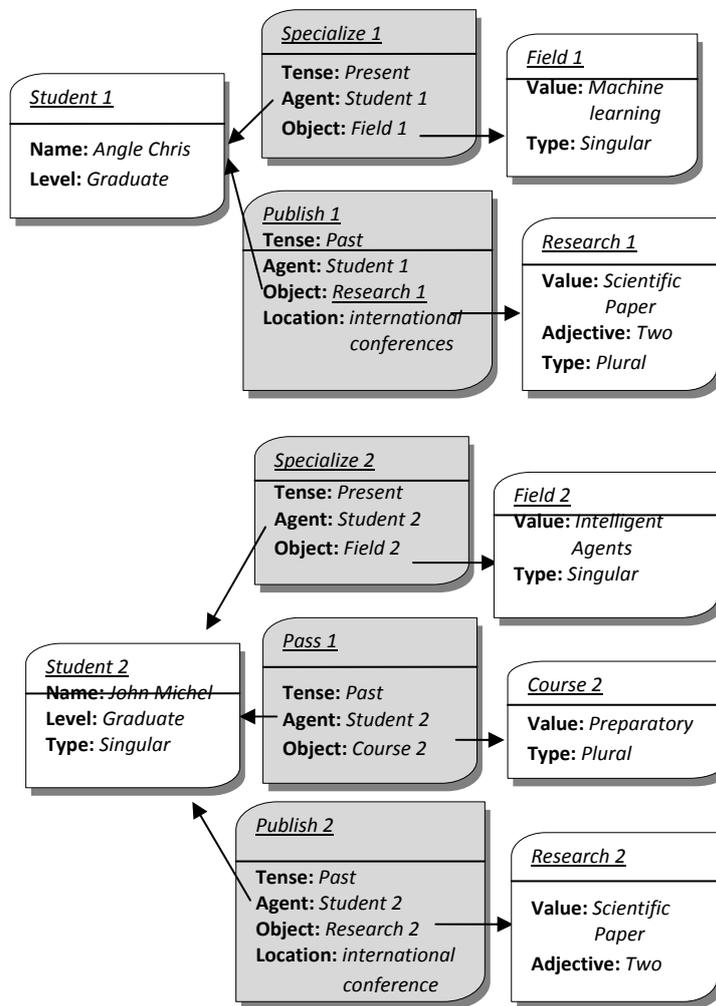
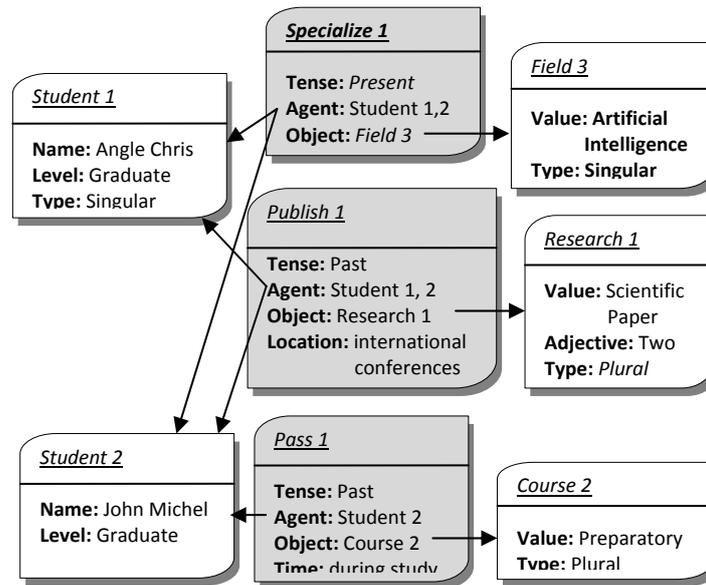


Figure 2: The rich semantic graph of graduate students' original text



**Figure 3: The reduced rich semantic graph**

*Angle Chris and John Michel are graduate students. They are specialized in Artificial Intelligence field. They published two papers in international conferences. During study, John*

**Figure 4: The graduate students summarized text**

## 4 CONCLUSION

This paper presents several graph reduction techniques. These techniques and their applications in different fields are discussed. These techniques include Workflow Management System, Computer Vision, Networks and semantic graphs. The objective of this research is to apply graph reduction techniques in abstractive text summarization.

## REFERENCES

- [1] Deepayan Chakrabarti, Christos Faloutsos: *Graph mining: Laws, generators, and algorithms*. ACM Comput. Surv. 38(1): (2006)
- [2] Y. Xu, S.M. Salapaka, and C.L. Beck, "On reduction of graphs and Markov chain models", in Proc. CDC-ECE, pp.2317-2322, 2011.
- [3] Wasim Sadiq and Maria E. Orlowska," Analyzing Process Models Using Graph Reduction Techniques" Information Systems, 25(2): 117-134, 2000.
- [4] H. Lin, Z. Zhap, H. Li, and Z. Chen, "A novel graph reduction algorithm to identify structural conflicts," in Proc. 35th Hawaii Int. Conf. Syst. Sci., vol. 9, p. 289, 2002.
- [5] Kai Lu; Liu, Qiang, "An Algorithm Combining Graph-Reduction and Graph-Search for Workflow Graphs Verification". En: 11th International Conference on Computer Supported Cooperative Work in Design, CSCWD 2007. Melbourne, Australia: Springer, p. 772-776, 2007.

- [6] B. Scheuermann and B. Rosenhahn, "SlimCuts: GraphCuts for High Resolution Images Using Graph Reduction," Energy Minimization Methods in Computer Vision and Pattern Recognition (EMMCVPR), July 2011.
- [7] F. Malgouyres and N. Lermé, "A non-heuristic reduction method for graph cut optimization," Technical Report hal-00692464, CCSD, April 2012.
- [8] Yan, J., K. Zhang, Z. Zhang, and S.C. Chen, "A Graph Reduction Method for 2D Snake Problems," IEEE Computer Society Conference on Computer Vision and Pattern Recognition, p. 6, 2007.
- [9] Casetti, C.; Lo Cigno, R.; Mellia, M.; Munafo, M., "A new class of QoS routing strategies based on network graph reduction," INFOCOM 2002. Twenty-First Annual Joint Conference of the IEEE Computer and Communications Societies Proceedings, IEEE, vol.2, pp.715-722, 2002.
- [10] C. Chekuri and N. Korula, "A graph reduction step preserving element-connectivity and applications," In ICALP (1), pages 254–265, 2009.
- [11] Chuzhoy, J., Khanna, S., "Algorithms for Single-Source Vertex-Connectivity," In: Proc. of IEEE FOCS, pp. 105–114 (October 2008)
- [12] Y. Xu, S.M. Salapaka, and C.L. Beck, "On reduction of graphs and Markov chain models", in Proc. CDC-ECE, pp.2317-2322, 2011.
- [13] Sushil Shrestha, "Role of Semantic Graph in Web Page Content Visualization". Athmandu University Journal of Science, Engineering And Technology Vol. 8, No. 1, 125- 133, February, 2012.
- [14] Delia Rusu, Blaz Fortuna, Dunja Mladenčić "Improved Semantic Graphs with Word Sense Disambiguation". Jozef Stefan Institute, Semantic Web, kcap09.stanford.edu, 2009.
- [15] Banu, M.; Karthika, C.; Sudarmani, P.; Geetha, T.V., "Tamil Document Summarization Using Semantic Graph Method," Conference on Computational Intelligence and Multimedia Applications, 2007. International Conference on, vol.2, no., pp.128,134, 13-15 Dec. 2007
- [16] Moawad, I.F.; Aref, M., "Semantic graph reduction approach for abstractive Text Summarization," Computer Engineering & Systems (ICCES), 2012 Seventh International Conference on , vol., no., pp.132,138, 27-29 Nov. 2012
- [17] Bartha, M.; Kresz, M., "A Depth-first Algorithm to Reduce Graphs in Linear Time," Symbolic and Numeric Algorithms for Scientific Computing (SYNASC), 2009 11th International Symposium on , vol., no., pp.273,281, 26-29 Sept. 2009.
- [18] A. Bakewell, D. Plump, C. Runciman, "Specifying Pointer Structures by Graph Reduction," in AGTIVE 2003, pp. 30–44. Springer-Verlag, 2004.
- [19] Huerta, J.M.; Lubensky, D., "Graph-based representation and techniques for NLU application development," Acoustics, Speech, and Signal Processing, 2003. Proceedings (ICASSP '03). 2003 IEEE International Conference on, vol.1, no., pp.I-288,I-291 vol.1, 6-10 April 2003.
- [20] Hui Li; Wenju Liu; Honglei Han, "Graph Reduction Algorithm for Hardware/Software Partitioning," Control, Automation and Systems Engineering (CASE), 2011 International Conference on , vol., no., pp.1,4, 30-31 July 2011.
- [21] Goldman, A., Ngoko, Y., "On Graph Reduction for QoS Prediction of Very Large Web Service Compositions," In: IEEE SCC, pp. 258–265 (2012).
- [22] A Novel Technique for Call Graph Reduction for Bug Localization. International Journal of Computer Applications 47(15):1-5, Published by Foundation of Computer Science, New York, USA, June 2012.
- [23] Rinehart, M.; Dahleh, M.A., "A graph reduction for bounding the value of side information in shortest path optimization," American Control Conference (ACC), 2010, vol., no., pp.4078- 4083, June 30 2010-July 2 2010

## BIOGRAPHY



Mostafa Aref is a professor of Computer Science and Vice Dean for Graduate studies and Research, Ain Shams University, Cairo, Egypt. Ph.D. of Engineering Science in System Theory and Engineering, June 1988, University of Toledo, Toledo, Ohio. M.Sc. of Computer Science, October 1983, University of Saskatchewan, Saskatoon, Sask. Canada. B.Sc. of Electrical Engineering - Computer and Automatic Control section, in June 1979, Electrical Engineering Dept., Ain Shams University, Cairo, EGYPT.



Dr. Ibrahim F. Moawad is an Associate Professor at the Faculty of Computer and Information Sciences, Ain Shams University, Egypt. He is interested in Artificial Intelligence, Information Retrieval, and Software Engineering research areas. He has published more than 50 academic and technical articles in both international journals and conferences.

## مسح شامل على الطرق المختلفة لتقليص التمثيل الرسومي الدلالي وتطبيقاته

مصطفى عارف<sup>1</sup>، إبراهيم فتحى معوض<sup>2</sup>، مروة محمود<sup>3</sup>

كلية الحاسبات والمعلومات – جامعة عين شمس

<sup>1</sup>mostafa.m.aref@gmail.com

<sup>2</sup>ibrahim\_moawad@cis.asu.edu.eg

<sup>3</sup>marwa\_mahmoud100@hotmail.com

### خلاصة:

يمثل تلخيص النصوص واحداً من أهم تطبيقات معالجة اللغات الطبيعية الذى يساعد المستخدمين على إدارة كم هائل من المعلومات. يتم تلخيص النصوص عن طريق تكثيف محتوى الوثائق واستخراج أهم الحقائق ذات الصلة أو الموضوعات المدرجة ودمجها داخل تقرير. يمكن تلخيص النصوص الى نوعين: تلخيص إسترجاعى وتلخيص تجرىدى. يهتم التلخيص الإسترجاعى بتحديد الاجزاء الهامة فى الوثيقة الاصلية وعرضها كما هى، أما التلخيص التجرىدى يهتم بإنتاج محتوى أكثر تعميماً من المحتوى الاصلى. ومن أحدث التقنيات التى ظهرت لتنفيذ التلخيص التجرىدى هى تقليص التمثيل الرسومي الدلالي. نقوم فى هذا البحث بعرض التقنيات والطرق المختلفة لتقليص التمثيل الرسومي الدلالي وتطبيقاته المختلفة. كما نقوم بعرض مثال تفصيلى لإنتاج ملخص نصى تجرىدى بواسطة تقليص التمثيل الرسومي الدلالي.