Journal of Discrete Mathematical Sciences & Cryptography
ISSN 0972-0529 (Print), ISSN 2169-0065 (Online)
Vol. 29 (2026), No. 10-A, pp. 3765–3775
DOI : 10.47974/JDMSC-3007

Topological simplification methods for large-scale knowledge graphs

Mohanad Alfiras *
Department of Electrical & Electronic Engineering
College of Engineering
Gulf University
Sanad 26489
Kingdom of Bahrain

Mohammed Hasan Aldulaimi †
Department of Computer Techniques Engineering
College of Engineering
Al-Mustaqbal University
Hillah
Babylon 51001
Iraq

Mrunmai Mandar Ranade §
Department of Engineering Sciences
Vishwakarma University
Pune 411048
Maharashtra
India

Gagan Tiwari ‡
Department of Computer Sciences
Noida International University
Greater Noida 203201
Uttar Pradesh
India

* *E-mail:* president@gulfuniversity.edu.bh (Corresponding Author)
† *E-mail:* mohammed.hassan@uomus.edu.iq
§ *E-mail:* mrunmai.ranade@vupune.ac.in
‡ *E-mail:* gagan.tiwari@niu.edu.in

© Taru Publications

M. Vinitha [@]
Department of Computer Science
Meenakshi College of Arts and Science
Meenakshi Academy of Higher Education & Research (Deemed to be University)
Chennai 600078
Tamil Nadu
India

Eshniyozov Asadbek [#]
Department of Economics
Termez University of Economics and Service
Termez 190111
Uzbekistan

Saodatkhon Ibragimova [$]
Department of Physics
Tashkent Institute of Irrigation and Agricultural Mechanization Engineers
National Research University
Kori Niyoziy 39
Tashkent 100000
Uzbekistan

and

Faculty of Information Technologies
University of Tashkent for Applied Sciences
Str. Gavhar 1
Tashkent 100149
Uzbekistan

Abstract

Topological simplification of large-scale knowledge graphs seeks to simplify the structure of knowledge graphs with minimal semantic loss and query time. The paper gives a detailed methodology on reducing the graph, incorporating mathematical equations, evaluation criteria, and analysis of the taxonomy. Graph theoretic operators, such as node contraction and edge pruning along with node motif preservation under constraint optimization are used to formalize the problem. A single set of quality standards are suggested: information retention, connectivity preservation, computational efficiency and downstream task accuracy. The simplification techniques identified in the paper are classified into three paradigms which include structural, heuristic and learning-driven. Graph neural networks and embedding-guided pruning are used in machine learning methods to reduce adaptively using centrality methods, clustering methods, and sparsification methods, unlike heuristic methods that do not use graph neural

[@] *E-mail:* **vinitham@maher.ac.in**
[#] *E-mail:* **asadbek_eshniyozov@tues.uz**
[$] *E-mail:* **saodatoy.ibragimova@gmail.com**

networks. Experimental results indicate that there are trade-offs between compression ratio and semantics fidelity.

Subject Classification: *05C62, 05C90.*

Keywords: *Knowledge graphs, Topological simplification, Graph neural networks, Graph sparsification, Embedding-based pruning, Semantic preservation.*

1. Introduction

The booming of the large scale knowledge graph has boosted significantly the applications of the data in areas like semantic search, recommendation system, biomedical analytics and intelligent decision support [1, 2]. The growing size, heterogeneity, and connectivity of these graphs, however, present significant difficulties in storage, computation, visualization and real-time reasoning [3, 4, 5]. Knowledge graphs can be large, with millions or even billions of nodes and edges, but because of their structural complexity, are frequently redundant and noisy and slow to process queries. Topological simplification, in its turn, is emerging as an important method to simplify graph representations of large scale and to control them [6, 7]. Topological simplification tries to reduce the size of the graph to the minimum possible amount, without destroying important structural and semantical information. Figure 1 reveals that it is a multilayer framework which makes it possible to simplify knowledge graph. It is unlike naive pruning in the sense that it should be sensitive to the connectivity patterns, node and relationship semantics importance to ensure that it is capable of retaining meaningful information [8, 9, 10].

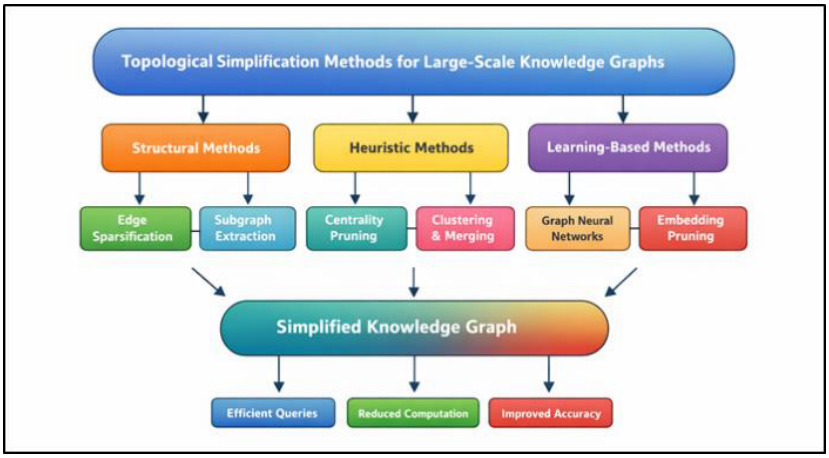


Figure 1
Multilayer Topological Simplification Framework for Large-Scale Knowledge Graphs

It may be performed through edge sparsification, node aggregation, subgraph recovery and motif conservation which are guided by formal mathematical models and optimization theory [11]. Recently, various forms of simplification have been suggested such as methods using heuristic information, such as centrality-based pruning and clustering, and more sophisticated machine learning based methods, such as graph neural networks and embedding-guided reduction [12].

2. Problem Formulation for Topological Simplification

In this section, it is proposed that graph simplification be defined as minimization of nodes and edges, with the required structure, semantics and functionality, to allow efficient computation, and the loss of information at small scale in large scale knowledge graph representations [13], [14].

2.1. *Mathematical Representation of Graph Reduction*

Graph reduction, this is the process of simplifying a given graph by combining nodes, deleting edges and subgraphs such that the connectivity is maintained and significant structural relationships are maintained [15].

$$G = (V, E, W) \tag{1}$$

The knowledge graph is represented by equation (1) to define the knowledge graph: it is a weighted structure (i.e. a graph) of nodes, edges and weights associated to them, as a formal graph-based analysis representation of entities and their relationships.

$$V = \{v1, v2, v3, \ldots, vn\} \tag{2}$$

The equation (2) defines the list of the vertices that are the objects in the graph, and each node in the graph is the unique concept, object, or instance that is part of the overall knowledge representation structure.

$$E \subseteq V \times V \tag{3}$$

The edge set is defined by equation (3) as ordered pairs of vertices, which means that there are relationships between entities, it describes the patterns of connectivity and allows the graph to have a structured interaction and flow of information.

$$\varphi(vi) = \int [0 \to T] \Sigma (j = 1 \to n) A_{ij(t)} \cdot \psi j(t) dt \tag{4}$$

The importance of nodes in equation (4) is modeled by summing up the interactions of the node over time with its neighbors to combine adjacency relationships with influence functions through time to measure the structural and semantic importance of each node.

$$E' = \{ (i, j) \in E \mid w_ij \geq \lambda \cdot \sqrt{(\varphi(vi) \cdot \varphi(vj))} \} \tag{5}$$

Equation (5) specifies the edge pruning condition to only retain the edges with large enough weight relative to the significance of the node to ensure that the weak or redundant edges are eliminated without important structure being lost.

$$\min(P) \int [0 \to T] \,||\, A \,-\, P \,\cdot\, A' \,\cdot\, P^{\mathrm{T}} \,||_F^2 \, dt \,+\, \beta \,\cdot\, \sqrt{(\Sigma(i)\, \varphi(vi))} \tag{6}$$

The optimization problem in (6) is the optimization of the graph reduction, where the error in reconstruction between the original and reduced graph is minimized and regularization is added to ensure that the key nodes are contributing and the entire structure is not destroyed.

2.2. *Evaluation Criteria for Simplification Quality*

The metrics used to measure the quality of simplification are as follows: retained information, similarity in structure, retained connectivity, computational efficiency and influence on subsequent processes as query accuracy and reasoning performance.

$$I \,= \frac{(\Sigma(i,j \in E')w_{ij})}{(\Sigma(i,j \in E)w_{ij})} \tag{7}$$

Retention of the information is measured by equation (7) which measures the total weights of the edges in the simplified graph against that of the original graph, which means that how much relational information is retained after the reduction process.

$$C \,= \frac{\int [0 \to T] \kappa\left(G'(t)\right)}{\kappa(G(t))} dt \tag{8}$$

The preservation of connectivity is assessed in equation (8) with a combination of ratio of connected components of simplified and original graph over time, such that the structural integrity and reachability of the graph remains unchanged.

$$\Delta\lambda \,=\, \sqrt{\Sigma(k = 1 \to r)(\lambda k \,-\, \lambda' k)^2} \tag{9}$$

To obtain spectral consistency, equation (9) measures the differences between the eigenvalues of the original and simplified graphs, which is a measure of how much the topological and structural properties of the graph are preserved during the simplification process.

$$L \,=\, \alpha(1 \,-\, I) + \beta\Delta\lambda \,+\, \gamma D \,+\, \delta \int [0 \to T](1 \,-\, C)dt \tag{10}$$

The total quality goal (as in Equation (10)) is a weighted sum of information loss, spectral deviation, downstream task error and connectivity degradation, which allows simplification performance across multiple criteria to be optimized.

3. Taxonomy of Topological Simplification Methods

Topological simplification approaches are classified into structural, heuristic and learning-based techniques as a result of the reduction strategy, information preservation methods, scalability and appropriateness to various types of large-scale knowledge graph usage.

$$G \,=\, (V, E, W), A \,\in\, R^{n \times n}, \Phi \,=\, \{\varphi^1, \varphi^2, \ldots, \varphi n\} \tag{11}$$

$$\Omega \,=\, [0 \to T]\Sigma(i = 1 \to n)\varphi \int_i^2 (t)\, dt \tag{12}$$

The graph in equation (11) is characterized by adjacency matrix and feature set, which depict the nodes, weighted edges and node importance features of taxonomy mapping. The feature space, as expressed in equation (12) is a combination of squared values of node importance as time progresses, reflecting the general distribution of features in a graph and the dynamic structural aspect of a graph.

$$C(G) = argmax_k \int [0 \to T] \Sigma (i = 1 \to n) \theta_{k(i,t)} \cdot \varphi_{i(t)dt} \tag{13}$$

In equation (13), the classification is decided on the basis of selecting the category that maximizes the sum of weighted importance of nodes, therefore enabling the simplification methods to be allocated based on structural patterns that are predominant.

$$Gs = (Vs, Es) \tag{14}$$

The graph with the least structure and the smallest set of nodes and edges that represents the original graph and does not distort the significant connectivity information is equation (14).

$$\min[0 \to T] \left|\left|A - As\right|\right| \int_F^2 dt + \lambda \int [0 \to T] \sqrt{trace\big(Ls(t)\big)} dt \tag{15}$$

The reconstruction error is minimized in equation (15) and smoothing is made using Laplacian trace, which ensures stability and structural fidelity in the simplified graph representation with time.

$$\frac{\max \int [0 \to T] \Sigma (i, j \in Eh) w_{ij(t)}}{\sqrt{\deg(i) \cdot \deg(j)}} dt \tag{16}$$

The weighted degree-normalized importance of the edge is maximized in equation (16) and it is applied in the simplification operations of a graph through the heuristic approach where the weighted degree-normalized importance of an edge is therefore favored to be preserved.

$$T* = argmin_T \int [0 \to T] L(G, GT) dt + \mu \int [0 \to T] \left|\left|\nabla GT\right|\right|^2 dt \tag{17}$$

It can be optimal to obtain the taxonomy assignment by minimizing the loss and regularization terms using equation (17) in which smoothness, consistency and appropriate classification of simplified graph structures are all guaranteed.

4. Proposed Topological Simplification Framework

The proposed approach is an adaptive semantic-aware topological simplification approach, which aims at simplifying large-scale knowledge graphs with semantic consistency, connectivity and analytical performance based on the properties of the knowledge graphs and their downstream applications. It combines graph-theoretic reduction mechanisms with graph neural network-based semantic preservation and multi-objective optimization to get scalable and efficient graph compression.

4.1. *Framework Overview*

Proposed Adaptive Semantic-Aware Graph Simplification Framework (ASGSF) is processed in a variety of steps including graph acquisition, node importance estimation, edge pruning based on semantics, community-based graph partitioning, graph semantic preservation via graph neural networks, and reconstruction of reduced graph. The goal of the framework is to reduce graph redundancy while preserving critical properties of both the semantic and topology of the graph utilized for reasoning, querying, and knowledge inference.

4.2. *Graph Acquisition and Preprocessing*

The framework starts by loading a large scale knowledge graph data from various semantic repositories where all the semantic resources, ontologies and entity-relation triples are stored. During preprocessing the redundant relations, isolated nodes, duplicate entities and incomplete semantic relations are removed to make the graph more consistent and reduce the computation time. This pre-processing step is to preserve the semantic meaning and structural validity of the initial representation of the graph before simplifications will be made.

The adjacency matrix representation of the graph is expressed as:

$$A_i j = 1, if\left(v_i, v_j\right) \in E$$
$$0, otherwise$$

Where, A_{ij} is 1 if there is a semantic connection between v_i and v_j and 0 otherwise. A value is put in the matrix if an edge exists between two nodes otherwise it is put as 0. Afterwards, feature normalization and embedding initializations are used to learn a graph representation using semantic feature vectors.

4.3. *Node Importance Estimation*

The structurally significant entities are kept in the simplification process by computing node importance scores based on hybrid centrality analysis of the structure. The process combines degree centrality, PageRank influence and semantic embedding relevance to find the most impacting and semantic relevant nodes in the knowledge graph. The higher important nodes are considered first when trimming the graph to ensure connectivity, semantics and analytical performance when going downstream.

An importance score for the node is the proportion of time the node will be under observation.

$$I(v_i) = \lambda_1 C_{d(v_i)} + \lambda_2 PR(v_i) + \lambda_3 S(v_i)$$

$C_{d(v_i)}$ is the degree centrality of node $v_i, PR(v_i)$ is the PageRank based influence score of the node and S(v_i) is the semantic embedding relevance of the node. The parameters λ_1, λ_2 and λ_3 are weights that can be applied to

the three components (structural centrality, global influence and semantic importance). The importance values are used to determine whether to keep nodes in a graph when trying to simplify it; nodes with higher values have a great contribution to graph connectivity, semantic influence and quality of knowledge representation.

5. Result and Discussion

The experimental findings reveal that topological simplification causes an important decrease in the size of graphs without compromising key structural and semantic features of the graph. Heuristic algorithms are faster to execute and have moderate precision, compared to machine learning algorithms, which are more fidelity and downstream task execution. The trade-offs between compression rate and information conservation can be observed especially in dense graphs.

Table 1
Performance Evaluation of Topological Simplification Methods

Metric	Original Graph	Heuristic-Based Method	GNN-Based Simplification
Number of Nodes	10,00,000	6,20,000	5,80,000
Number of Edges	52,00,000	29,00,000	26,00,000
Information Retention (%)	100	85.6	92.3
Query Processing Time (ms)	420	260	210

Table 1 indicates that, topological simplification often causes a substantial decrease in the size of graphs without losing important information. Significant node and edge reduction in heuristic-based approach with moderate information retention at 85.6% enhances the performance of query processing. Figure 2 compares the performance of the graph simplification methods.

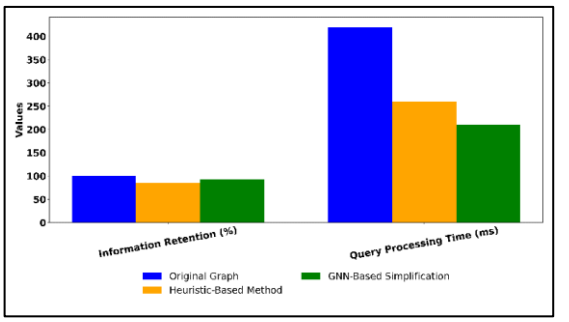


Figure 2
Graph Simplification Performance Comparison

Conversely, the GNN based method can retain more information with 92.3% and can compress and respond to queries faster.

Table 2
Downstream Task Performance after Simplification

Metric	Original Graph	Heuristic-Based	Embedding-Guided Pruning
Link Prediction Accuracy (%)	91.2	86.5	89.8
Node Classification Accuracy (%)	88.7	83.9	87.6
F1-Score (%)	89.4	84.7	88.5
AUC Score	0.93	0.87	0.91

Table 2 indicates that performance of downstream tasks slightly reduces in the wake of simplification, but is still competitive. The heuristic methodology leads to the observable reduction in the accuracy in all measures, which indicates the low level of maintenance of the semantic relationships.

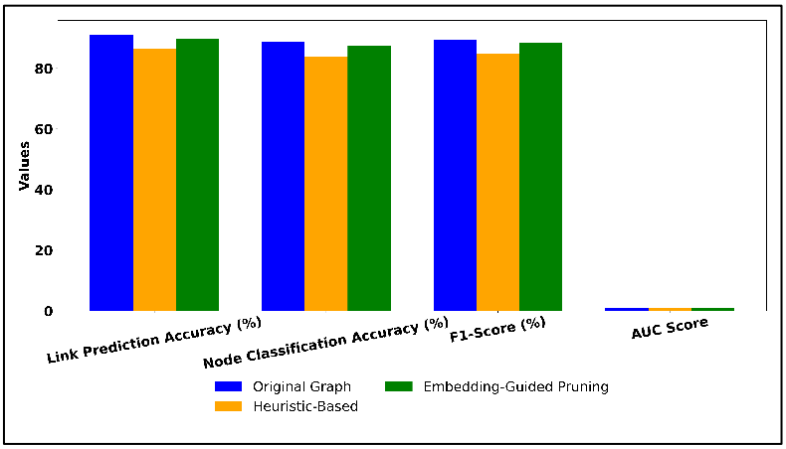


Figure 3
Graph Pruning Technique Performance Comparison

There is a performance comparison of graph pruning techniques as shown in Figure 3. Embedding-guided pruning, in contrast, preserves more performance than the original graph, and is more accurate in link prediction and node classification, F1-score, and AUC. This illustrates that the simplification that is driven by learning is more appropriate to represent latent graph structures and, hence, does not lead to significant degradation (latent) in predictive tasks, yet, simplification of the task reduces complexity and increases computational efficiency in general.

6. Conclusion

This paper provides an in-depth discussion of the topological simplification of large-scale knowledge graphs, and how it is necessary to develop effective reduction strategies that do not degrade the structure or semantics. The formalization of the problem and the classification of the existing methods of addressing the issue into heuristic and machine learning-based ones offers the work a coherent view on the topic of graph simplification. Through experimental results or observations, it can be seen that heuristic methods are computationally efficient, but the accuracy and versatility provided by learning models are much better, especially in the complex and dynamic graph setup. The results indicate the need to balance compression efficiency and information retention to guarantee good downstream performance.

References

[1] A Lobo-Santos and J. Borrego-Díaz, "Enhancing mathematical knowledge graphs with large language models," *Modelling*, vol. 6, no. 3, Art. no. 53 (2025), doi: 10.3390/modelling6030053.

[2] D. Zhong, L. Wen, L. Bi, and Y. Liu, "An efficient and automatic simplification method for arbitrary complex networks in mine ventilation," *Mathematics*, vol. 12, no. 18, Art. no. 2815 (2024), doi: 10.3390/math12182815.

[3] B. Wang, G. Wu, Q. Zhao, Y. Li, Y. Gao, and J. She, "A topology-preserving simplification method for 3D building models," *ISPRS International Journal of Geo-Information*, vol. 10, no. 6, Art. no. 422 (2021), doi: 10.3390/ijgi10060422.

[4] Y. Wang, Y. Liu, K. Ding, S. Wei, X. Zhang, and Y. Zhao, "Dynamic optimization method of knowledge graph entity relations for smart maintenance of cantilever roadheaders," *Mathematics*, vol. 11, no. 23, Art. no. 4833 (2023), doi: 10.3390/math11234833.

[5] N. Ghanbari, M. A. Dehghanizadeh, and S. Alikhani, "Some new results on total irregularity of graphs," *Journal of Information and Optimization Sciences*, vol. 47, no. 3, pp. 1051–1062 (2026), doi: 10.47974/JIOS-1845.

[6] M. Hofer, D. Obraczka, A. Saeedi, H. Köpcke, and E. Rahm, "Construction of knowledge graphs: Current state and challenges," *Information*, vol. 15, no. 8, Art. no. 509 (2024), doi: 10.3390/info15080509.

[7] L. Yang, H. Guo, Y. Dai, and W. Chen, "A method for complex question-answering over knowledge graph," *Applied Sciences*, vol. 13, no. 8, Art. no. 5055 (2023), doi: 10.3390/app13085055.

[8] R. A. Mustafa, M. S. Ahmed, and A. M. Ali, "Wiener index for a square of some special graphs," *Journal of Information and Optimization Sciences*, vol. 47, no. 3, pp. 1175–1185 (2026), doi: 10.47974/JIOS-2156.

[9] F. Rolli, C. Parretti, R. Barbieri, A. Polidoro, and B. Bindi, "RDF knowledge graphs designed with axiomatic methodology to enhance Industry 4.0," *Machines*, vol. 13, no. 1, Art. no. 58 (2025), doi: 10.3390/machines13010058.

[10] M. Yani and A. A. Krisnadhi, "Challenges, techniques, and trends of simple knowledge graph question answering: A survey," *Information*, vol. 12, no. 7, Art. no. 271 (2021), doi: 10.3390/info12070271.

[11] X. Guo, H. Qian, F. Wu, and J. Liu, "A method for constructing geographical knowledge graph from multisource data," *Sustainability*, vol. 13, no. 19, Art. no. 10602 (2021), doi: 10.3390/su131910602.

[12] P. Prabu, M. Alsulami, D. Alsadie, A. K. J. Saudagar, M. Alkhathami, and R. C. Poonia, "Utilizing behavioral deep learning models to monitor and alert physicians regarding trauma cases," *Journal of Information and Optimization Sciences*, vol. 44, no. 5, pp. 983–995 (2023), doi: 10.47974/JIOS-1461.

[13] M. Zamini, H. Reza, and M. Rabiei, "A review of knowledge graph completion," *Information*, vol. 13, no. 8, Art. no. 396 (2022), doi: 10.3390/info13080396.

[14] Y. Song, W. Li, G. Dai, and X. Shang, "Advancements in complex knowledge graph question answering: A survey," *Electronics*, vol. 12, no. 21, Art. no. 4395 (2023), doi: 10.3390/electronics12214395.

[15] A Lee, "Geometry and topology correction of 3D building models with fragmented and disconnected components," *ISPRS International Journal of Geo-Information*, vol. 14, no. 5, Art. no. 198 (2025), doi: 10.3390/ijgi14050198.

Received April, 2026