

Cost-Preserving Distributed Approximate Pattern Mining for Scalable RDF Graph Summarization

Mussab Zneika¹

¹*Independent Researcher, Paris, France*

Abstract

RDF graph summarization based on approximate pattern mining has been shown to produce compact, high-quality summaries by capturing the structural regularities of RDF graphs while naturally accommodating the heterogeneous property descriptions of RDF resources. However, existing cost-based approaches are inherently memory-bound, limiting their applicability to large RDF graphs. Although distributed pattern mining techniques improve scalability through data partitioning, partitioning prevents the correct evaluation of the global reconstruction cost when patterns extracted from different partitions are merged, making it impossible to preserve the original optimization objective during distributed execution. To address this limitation, we propose a distributed framework for approximate RDF graph summarization that combines parallel pattern extraction with a cost-preserving merging strategy. The proposed Missing-Neighbors Index retains the information required to evaluate merge costs locally without rescanning the original dataset, enabling distributed optimization while preserving the original reconstruction-cost objective. Implemented using Hadoop MapReduce and evaluated on several real-world RDF datasets, the framework achieves execution-time improvements of up to $2.34\times$ while producing summaries whose quality remains nearly identical to the sequential SemSum+ baseline. Although developed for RDF graph summarization, the proposed framework is applicable to a broader class of cost-based approximate pattern mining methods based on binary representations and global cost functions.

Keywords

RDF Graph Summarization, Knowledge Graphs, Approximate Pattern Mining, Distributed Data Mining, MapReduce, Cost-Based Optimization, Scalability

1. Introduction

RDF graph summarization aims to construct compact representations of knowledge graphs while preserving their essential structural and semantic characteristics. Among the existing summarization approaches, methods based on approximate pattern mining are particularly well suited to RDF data because they naturally accommodate the heterogeneous property descriptions of RDF resources while explicitly controlling the trade-off between compression and reconstruction quality. SemSum+ [1, 2, 3] follows this principle by formulating RDF graph summarization as a cost-based approximate pattern mining problem using the PaNDa+ optimization algorithm [4]. By minimizing a global reconstruction cost, it produces high-quality summaries while providing explicit control over approximation quality. However, because the optimization is performed on the complete binary representation of the RDF graph, SemSum+ is inherently memory-bound and cannot efficiently process knowledge graphs that exceed the memory capacity of a single machine.

Several distributed pattern mining approaches [5, 6, 7] improve scalability by partitioning the input data, extracting patterns independently on each partition, and combining the local results. This strategy is effective for exact pattern mining algorithms such as FP-Growth [8] and Apriori [9], where locally discovered patterns remain globally valid after partitioning. In contrast, cost-based approximate pattern mining relies on a global reconstruction-cost optimization. Once the RDF graph is partitioned, each worker has access only to a partial view of the original binary representation, making it impossible to correctly evaluate the reconstruction cost of patterns originating from different partitions. Consequently, independently combining local patterns introduces approximation errors that are not reflected by the original optimization objective.

Posters, Demos, Blue Sky, and Tutorials at SEMANTiCS 2026, Sep 2026, Ghent, Belgium

✉ mussab.zneika@cyu.fr (M. Zneika)



© 2026 Copyright for this paper by its authors. Use permitted under Creative Commons License Attribution 4.0 International (CC BY 4.0).

In this paper, we address this limitation by proposing a distributed framework for approximate RDF graph summarization that preserves the original reconstruction-cost optimization during distributed execution. Unlike existing distributed pattern mining approaches, the proposed framework enables global cost-preserving optimization without rescanning the original dataset. The framework consists of two phases: distributed pattern extraction followed by cost-driven pattern merging. During the first phase, the binary representation is partitioned by RDF resources (transactions), while each partition retains the complete set of RDF properties (items). Every worker independently extracts a local set of approximate patterns, after which patterns sharing the same set of RDF properties are consolidated by combining their covered resources. This step reconstructs patterns that were fragmented during partitioning but is not sufficient to recover the globally optimal summary. For example, one partition may independently extract the pattern $\{p_1, p_2, p_3\}$, while another extracts $\{p_1, p_2, p_3, p_4\}$. Consolidating only identical property sets reconstructs the fragmented pattern but cannot determine whether merging these overlapping patterns would further reduce the global reconstruction cost. Since the information required to evaluate such candidate merges is no longer available after partitioning, the second phase introduces the *Missing-Neighbors Index*, which stores compact local statistics that enable merge-cost evaluation without rescanning the original dataset while preserving the original optimization objective. The proposed framework was implemented using Hadoop MapReduce and evaluated on several real-world RDF datasets. Experimental results demonstrate consistent reductions in execution time while maintaining summary quality comparable to the sequential SemSum+ baseline. Although instantiated for RDF graph summarization, the framework is applicable to a broader class of cost-based approximate pattern mining methods relying on binary representations and global cost optimization.

The main contributions of this work are: (i) identifying the limitations of existing distributed pattern mining approaches when applied to cost-based approximate pattern mining; (ii) proposing a distributed framework that preserves the original global reconstruction-cost optimization during distributed execution; and (iii) introducing the *Missing-Neighbors Index*, which enables local merge-cost evaluation without rescanning the original dataset and improves scalability while preserving summary quality.

2. Related Work

RDF graph summarization approaches can be broadly classified into equivalence-based [10, 11, 12], schema-driven [13, 14], and pattern-based methods [15, 3, 1]. More recent work has also investigated structural summarization and schema profiling for large knowledge graphs [16, 17, 18, 19], highlighting the growing need for scalable knowledge graph analytics.

Among pattern-based approaches, SemSum+ [20] formulates RDF graph summarization as a cost-based approximate pattern mining problem using the PaNDa+ optimization algorithm [4]. Although it produces high-quality summaries, both SemSum+ and PaNDa+ are inherently memory-bound, limiting their applicability to large RDF graphs. Several distributed pattern mining approaches [5, 7] improve scalability by partitioning the input data and combining locally extracted patterns. While this strategy is effective for exact pattern mining algorithms such as FP-Growth [8] and Apriori [9], it does not preserve the global reconstruction-cost optimization required by cost-based approximate pattern mining. Unlike existing work, the proposed framework enables distributed execution while preserving the original reconstruction-cost optimization. This is achieved through a cost-driven merging strategy supported by the proposed *Missing-Neighbors Index*, which enables merge-cost evaluation without rescanning the original dataset.

3. Method

This section briefly reviews the approximate pattern mining model underlying SemSum+ before presenting the proposed distributed framework.

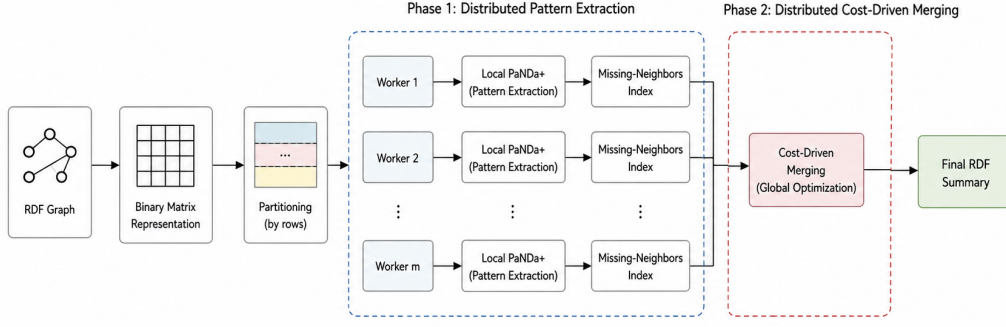


Figure 1: Overview of the proposed distributed framework.

3.1. Approximate Pattern Mining Background

Following SemSum+ [20], an RDF graph is transformed into a binary matrix $\mathcal{D} \in \{0, 1\}^{N \times M}$, where rows represent RDF resources (transactions) and columns correspond to RDF properties (items). An entry $\mathcal{D}(i, j) = 1$ indicates that property j is associated with resource i . Details of the RDF-to-matrix transformation are given in [20].

The summarization problem is formulated as the Approximate Top- k Pattern Discovery problem, which seeks a set of at most k patterns minimizing a global reconstruction cost. Given a pattern set Π , reconstruction errors are represented by the noise matrix

$$\mathcal{N} = \mathcal{D} \oplus \bigvee_{P \in \Pi} (P_T P_I^T), \quad (1)$$

where $\vee$ and $\oplus$ denote the logical OR and XOR operators, respectively. The optimal pattern set is obtained by minimizing

$$\Pi_k = \arg \min_{|\Pi| \leq k} J(\Pi, \mathcal{N}). \quad (2)$$

SemSum+ solves this optimization using the PaNda+ algorithm [4], which minimizes

$$J^+(\Pi_k, \mathcal{N}) = \gamma_{\mathcal{N}}(\mathcal{N}) + \rho \sum_{P \in \Pi_k} \gamma_P(P), \quad (3)$$

where $\gamma_{\mathcal{N}}$ denotes the encoding cost of the reconstruction error, γ_P the description cost of the extracted patterns, and ρ a regularization parameter. Since this optimization is performed on the complete binary matrix, the entire dataset must reside in memory, limiting scalability for large RDF graphs.

3.2. Distributed RDF Graph Summarization Framework

To overcome this limitation, we propose a distributed framework implemented on Hadoop MapReduce. The framework distributes both pattern extraction and cost-preserving optimization while maintaining the original reconstruction-cost semantics. Although instantiated using SemSum+ and PaNda+, the proposed approach is applicable to approximate pattern mining methods relying on binary representations and global cost optimization. Figure 1 presents an overview of the proposed framework.

3.2.1. Phase 1: Parallel Pattern Extraction

The binary matrix is partitioned by RDF resources (transactions), while every partition retains the complete set of RDF properties. Each worker independently applies SemSum+ to its assigned partition

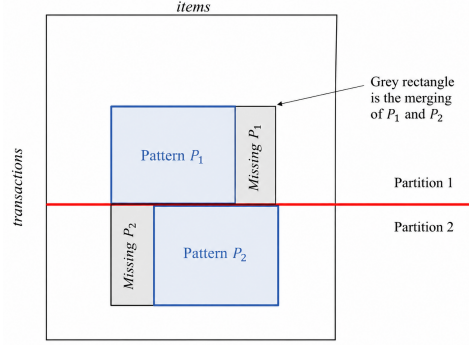


Figure 2: Illustration of the proposed cost-preserving merging strategy using the Missing-Neighbors Index.

and extracts a local set of approximate patterns. Patterns extracted from different partitions that share the same property set are then consolidated by combining their covered resources into a single intermediate representation.

This consolidation reconstructs patterns that were fragmented during partitioning but does not recover globally optimal patterns. For example, one partition may produce the pattern $\{p_1, p_2, p_3\}$, whereas another extracts $\{p_1, p_2, p_3, p_4\}$. Consolidating only identical property sets preserves the first pattern but cannot determine whether merging these overlapping patterns decreases the global reconstruction cost.

3.2.2. Phase 2: Cost-Preserving Pattern Merging

The second phase addresses this limitation by evaluating candidate merges directly in the compressed pattern space, avoiding repeated scans of the original binary matrix.

The main challenge is that each local pattern contains only the information available within its partition. Consequently, the information required to evaluate the reconstruction cost of merging patterns extracted from different partitions is no longer directly available.

To preserve the original optimization objective, we introduce the *Missing-Neighbors Index*, which records the frequencies of non-pattern RDF properties associated with the resources covered by each pattern. These statistics provide sufficient information to estimate the reconstruction noise and evaluate merge costs without accessing the original binary matrix.

To improve scalability, only candidate pattern pairs whose property or resource overlap exceeds a threshold θ are considered for merging. Figure 2 illustrates the proposed cost-preserving merging strategy.

4. Experimental Evaluation

The evaluation assesses two aspects of the proposed framework: (i) its ability to reduce execution time compared with the sequential SemSum+ implementation, and (ii) its ability to preserve the quality of the generated RDF summaries during distributed execution.

4.1. Experimental Setup and Datasets

Experiments were conducted on a Hadoop cluster consisting of one master node and two worker nodes, each equipped with an Intel Core i5 processor and 4 GB RAM. The distributed implementation was developed using Hadoop MapReduce, while the sequential baseline was executed on the master node using identical parameter settings.

Hadoop MapReduce was selected to validate the proposed cost-preserving strategy. The objective was not to evaluate large-scale distributed performance, but to verify that the original reconstruction-cost optimization is preserved during distributed execution. Preliminary experiments showed that

Table 1

Datasets used in the evaluation.

Dataset	Transactions	Items	Domain
JPeel	76,229	49	Multimedia
Jamendo	335,925	50	Music metadata
Sec	460,446	20	Security data
Bank	200,429	34	Financial data
WordNet	647,215	123	Lexical knowledge

Table 2

Execution time comparison.

Dataset	Seq. (s)	Dist. (s)	$\times$
JPeel	16	12	1.33
Jamendo	71	36.13	1.96
Sec	46	26	1.76
Bank	6465	4741	1.36
WordNet	947	403.27	2.34

Table 3Schema-level class quality (F_1^c).

Dataset	Seq.	Dist.
JPeel	0.900	0.894
Jamendo	0.950	0.946
Sec	1.000	0.998
Bank	0.930	0.925
WordNet	0.940	0.936

the sequential implementation exceeded the memory limits of a single 4 GB machine, motivating the distributed implementation. Evaluation on larger clusters and substantially larger RDF graphs is left for future work. The framework was evaluated on five real-world RDF datasets.

4.2. Results

Tables 2 and 3 compare the distributed implementation with the sequential SemSum+ baseline. The proposed framework achieves speed-ups ranging from $1.33\times$ to $2.34\times$ while preserving summary quality, with schema-level differences below 0.01 for all datasets. Although the observed speed-ups are influenced by the modest cluster size and Hadoop MapReduce overhead, the results demonstrate that the proposed framework preserves the original reconstruction-cost optimization while reducing execution time, providing a practical foundation for larger distributed deployments.

5. Conclusion

This paper presented a distributed framework for approximate RDF graph summarization that preserves the global reconstruction-cost optimization of cost-based approximate pattern mining. The proposed Missing-Neighbors Index enables merge-cost evaluation without rescanning the original dataset, allowing distributed execution while maintaining the original optimization objective. Experimental results demonstrate that the proposed approach reduces execution time while preserving summary quality. More broadly, this work shows that cost-based approximate pattern mining can be effectively parallelized without sacrificing its optimization semantics. Beyond RDF graph summarization, the proposed framework provides a foundation for distributed cost-based approximate pattern mining, opening new opportunities for scalable analysis of large knowledge graphs. Future work includes evaluation on larger distributed infrastructures, implementation on Apache Spark, and extension to other approximate pattern mining and knowledge graph analytics tasks.

Declaration on Generative AI

During the preparation of this work, the author used ChatGPT (OpenAI) for language editing, grammar and spelling checking, and rephrasing. After using the tool, the author reviewed and edited the content as needed and takes full responsibility for the content of the publication

References

- [1] M. Zneika, C. Lucchese, D. Vodislav, D. Kotzinos, Rdf graph summarization based on approximate patterns, in: International Workshop on Information Search, Integration, and Personalization, Springer, 2015, pp. 69–87.
- [2] M. Zneika, D. Vodislav, D. Kotzinos, Quality metrics for RDF graph summarization, *Semantic Web* 10 (2019) 555–584. URL: <https://doi.org/10.3233/SW-190346>. doi:10.3233/SW-190346.
- [3] M. Zneika, C. Lucchese, D. Vodislav, D. Kotzinos, Summarizing linked data rdf graphs using approximate graph pattern mining, in: 19th International Conference on Extending Database Technology, 2016.
- [4] C. Lucchese, S. Orlando, R. Perego, A unifying framework for mining approximate top-k binary patterns, *IEEE TKDE* 26 (2014) 2900–2913.
- [5] M. Riondato, J. A. DeBrabant, R. Fonseca, E. Upfal, PARMA: a parallel randomized algorithm for approximate association rules mining in mapreduce, in: 21st ACM International Conference on Information and Knowledge Management, CIKM’12, Maui, HI, USA, October 29 - November 02, 2012, 2012, pp. 85–94. URL: <https://doi.org/10.1145/2396761.2396776>. doi:10.1145/2396761.2396776.
- [6] Z. Zhang, G. Ji, M. Tang, Mreclat: An algorithm for parallel mining frequent itemsets, in: International Conference on Advanced Cloud and Big Data, CBD 2013, Nanjing, China, December 13-15, 2013, 2013, pp. 177–180. URL: <https://doi.org/10.1109/CBD.2013.22>. doi:10.1109/CBD.2013.22.
- [7] S. Moens, E. Aksehirli, B. Goethals, Frequent itemset mining for big data, in: Proceedings of the 2013 IEEE International Conference on Big Data, 6-9 October 2013, Santa Clara, CA, USA, 2013, pp. 111–118. URL: <https://doi.org/10.1109/BigData.2013.6691742>. doi:10.1109/BigData.2013.6691742.
- [8] J. Han, J. Pei, Y. Yin, R. Mao, Mining frequent patterns without candidate generation: A frequent-pattern tree approach, *Data mining and knowledge discovery* 8 (2004) 53–87.
- [9] R. Agrawal, T. Imieliński, A. Swami, Mining association rules between sets of items in large databases, in: *Acm sigmod record*, volume 22, ACM, 1993, pp. 207–216.
- [10] S. Khatchadourian, M. P. Consens, Exploring rdf usage and interlinking in the linked open data cloud using explod., in: LDOW, 2010.
- [11] S. Campinas, T. E. Perry, D. Ceccarelli, R. Delbru, G. Tummarello, Introducing rdf graph summary with application to assisted sparql formulation, in: Database and Expert Systems Applications (DEXA), 2012 23rd International Workshop on, IEEE, 2012, pp. 261–266.
- [12] S. Khatchadourian, M. P. Consens, Constructing bisimulation summaries on a multi-core graph processing framework, in: Proceedings of the GRADES’15, ACM, 2015, p. 8.
- [13] B. Zhang, Y. He, L. Pintscher, A. M. Peñuela, E. Simperl, Schema generation for large knowledge graphs using large language models, 2025. URL: <https://arxiv.org/abs/2506.04512>. arXiv:2506.04512.
- [14] E. Pürmayr, SHACL Shapes Extraction for Evolving Knowledge Graphs, Ph.D. thesis, 2025.
- [15] A. K. Joshi, P. Hitzler, G. Dong, Towards logical linked data compression, in: Proceedings of the Joint Workshop on Large and Heterogeneous Data and Quantitative Formalization in the Semantic Web, LHD+ SemQuant2012, at the 11th International Semantic Web Conference, ISWC2012, Citeseer, 2012.
- [16] A. Scherp, D. Richerby, T. Blume, M. Cochez, J. Rau, Structural summarization of semantic graphs using quotients, *Transactions on Graph Data and Knowledge* 1 (2023) 12.
- [17] B. Spahiu, M. Palmonari, R. A. Alva Principe, A. Rula, Understanding the structure of knowledge graphs with abstat profiles, *Semantic Web* 15 (2024) 1519–1545.
- [18] M. Lissandrini, K. Rabbani, K. Hose, Mining validating shapes for large knowledge graphs via dynamic reservoir sampling, volume 3741, 2024, pp. 25–34.
- [19] A. Arpaci-Dusseau, Z. Zhou, X. Chen, Accurate and fast approximate graph pattern mining at scale, *arXiv preprint arXiv:2405.03488* (2024).
- [20] M. Zneika, Querying semantic web/linked data graphs using summarization, Ph.D. thesis, Université de Cergy Pontoise, 2019.