International Journal for Multidisciplinary Research (IJFMR)
E-ISSN: 2582-2160 ● Website: www.ijfmr.com ● Email: editor@ijfmr.com

Literature Review for Accuracy Calculation

Nang Nandar Tun[1], Dr. Nyo Nyo Yee[2]

[1]Associated Professor, Faculty of Information Science, University of Computer Studies (Mandalay), Mandalay, Myanmar.
[2]Professor and Head, Department of Information Science, University of Technology (Yadanapon Cyber ity), Pyin Oo Lwin, Myanmar.

Abstract
Accuracy calculation is a crucial factor in evaluating query and data retrieval systems across multiple paradigms, including keyword-based and semantic-based for relational database, and graph database approaches. Keyword-based systems depend on exact or partial string matches, which often fail to capture deeper contextual meaning. Semantic-based methods leverage techniques like cosine similarity to analyze meaning and improve accuracy in retrieving relevant results. In relational databases, accuracy calculation typically involves matching strict conditions through structured queries, which perform well with normalized data but falter in modeling complex relationships. In contrast, graph databases, such as Neo4j, utilize graph structures and semantic queries, offering improved accuracy in capturing relationships between entities. Previous research demonstrates that semantic-based methods consistently outperform keyword-based approaches in complex domains. Graph databases further extend this by representing connections explicitly. However, challenges remain, such as determining similarity thresholds, handling data sparsity, and ambiguity in user queries. Understanding the strengths and limitations of each approach is critical for developing effective information retrieval systems. This review provides insights into the evolving landscape of accuracy calculation methods.

Keywords: Accuracy calculation, Keyword-based approach, Semantic similarity, Relational database, Graph database, Cosine similarity, Information retrieval.

1. Introduction

With the rapid growth of data in various formats and structures, effective information retrieval has become increasingly important in both academia and industry. Relational databases, traditionally used for structured data storage and retrieval, rely on well-defined schemas and SQL queries to provide accurate answers to user queries. However, they face challenges in flexibility and handling semantically complex queries. Graph databases have emerged as a powerful alternative, providing a natural way to represent and query interconnected data using nodes and relationships.

Keyword-based search remains one of the most common techniques for querying both relational and graph databases. This approach focuses on literal string matching and is computationally efficient. However, keyword-based methods often fall short in capturing the semantic intent of user queries, leading to lower accuracy in many domains. To address this limitation, semantic-based search approaches have been proposed. These approaches leverage techniques such as natural language processing, semantic similarity measures, and graph pattern matching to improve accuracy by understanding the meaning behind the query.

This paper explores the concepts, challenges, and methods for calculating accuracy in both keyword-based and semantic-based search approaches across relational and graph databases. It also reviews existing literature and compares the strengths and weaknesses of each approach in terms of accuracy, scalability, and complexity.

The rest of the paper is described as follows. In section 2, we describe background theory. Literature reviews are described in section 3. In section 4, we described discussion and conclusion.

2. Background Theory

In this section, relational database, graph database, keyword-based search and semantic based search are described.

2.1. Relational Database

Relational databases organize data into tables with rows and columns, using structured schemas. They are ideal for storing highly structured, transactional data. They enforce data integrity through constraints and relationships via primary and foreign keys. Accuracy in relational databases depends on exact matching conditions in SQL queries. Joins are used to combine data from multiple tables, requiring careful schema design. However, relational databases struggle to handle data with complex or dynamic relationships. Scalability can be challenging when dealing with massive interconnected data. Traditional relational systems work best with normalized data. They are less flexible when the schema needs to evolve frequently. Complex queries with multiple joins can lead to performance issues.

2.2. Graph Database

Graph databases store data as nodes and relationships, forming a graph structure. They excel in representing complex, interconnected data, such as social networks or knowledge graphs. Relationships are first-class citizens, allowing fast traversal between entities. Graph databases are schema-flexible, adapting to evolving data structures. They can efficiently handle queries involving deep relationships. Accuracy in retrieval is enhanced by using graph traversal patterns rather than table joins. They support semantic queries that exploit the nature of graph connections. Graph databases perform well with highly connected data. Neo4j is a popular graph database widely used in academia and industry. Despite their advantages, graph databases can have challenges in handling large-scale transactional processing.

2.3. Keyword-Based Search in Both Databases

Keyword-based search involves matching user-provided terms with stored data fields. In relational databases, it relies on text-based queries, often using SQL LIKE conditions or full-text search indices. It generally ignores the meaning or context of words, focusing on literal string matching. In graph databases, keyword search scans node and relationship properties for exact terms. This approach often fails when synonyms, variations, or semantic meaning are important. Keyword search can be fast and simple to implement. It is effective for precise, unambiguous terms. However, it performs poorly for exploratory or context-dependent queries. Both relational and graph databases require indexing for efficient keyword searches. Keyword search is generally less accurate in capturing user intent compared to semantic methods.

2.4. Semantic-Based Search in Both Databases

Semantic-based search interprets the meaning behind user queries. It employs techniques like tokenization, vector similarity (e.g., cosine similarity), or advanced NLP models. In relational databases, semantic search often involves layered architectures combining SQL with external semantic engines. In graph databases, semantic search can use graph patterns, ontology matching, and similarity scores. This

approach improves accuracy by accounting for synonyms, related concepts, and context. Semantic methods excel in exploratory search where user queries are vague or varied. They often use machine learning models to generate embeddings for comparison. Graph databases naturally support semantic relationships, making them well-suited for semantic queries. Despite improved accuracy, semantic-based search introduces computational overhead. Maintaining models and thresholds for similarity also poses practical challenges.

3. Literature Review

In this section we provide two parts. The first part includes accuracy calculation for key-word-based search in relational and graph databases. In second part, accuracy calculation for semantic-based search in relational and graph databases are also described.

3.1. Accuracy Calculation for Keyword-Based Search in Relational and Graph Databases

Accuracy in keyword-based search is often measured using standard information retrieval metrics such as precision, recall, and F1-score. Given a query, the system returns a set of results. Precision is calculated as the number of relevant results returned divided by the total number of results returned. Recall is the number of relevant results returned divided by the total number of relevant results in the dataset. Accuracy, sometimes used more generally, refers to the proportion of correct results among all retrieved results plus the number of correct rejections (i.e., documents not retrieved that are irrelevant). In many IR studies, precision and recall are more commonly reported for keyword-based search performance evaluation.

Salton et al. (1975) introduced the vector space model, foundational for keyword matching in relational IR systems. They demonstrated how documents and queries could be represented as term vectors, enabling mathematical similarity calculations. This model became a benchmark for measuring precision and recall in relational database retrieval tasks [21].

Zobel and Moffat (2006) provided a comprehensive survey of text indexing techniques, comparing various methods of indexing large text corpora. They analyzed how indexing affects keyword search accuracy and performance. Their work highlighted limitations in recall due to vocabulary mismatch and proposed improvements in index structures [16].

Li et al. (2007) investigated techniques for keyword query translation into structured relational queries. They examined the difficulty in mapping user keywords to database schema elements, affecting accuracy. Their experiments showed that keyword-based retrieval often fails to capture nuanced semantics in relational databases [13].

Chaudhuri et al. (2004) studied ranking mechanisms in relational keyword searches. They proposed scoring models to prioritize results matching multiple query terms. Their research showed that ranking improved user-perceived accuracy but still suffered when queries used synonyms or ambiguous terms [17].

Zhang et al. (2017) proposed entropy-based query selection in active learning for entity search tasks. They showed that carefully selecting queries improves accuracy of keyword-based retrieval. Their work emphasized challenges in disambiguating entities using keyword-only methods [2].

Angles and Gutierrez (2008) surveyed graph database models, focusing on how graph structures support keyword queries. They discussed graph traversal techniques and their impact on retrieval accuracy. They noted that while graph structures improve relationship discovery, keyword-based search remains limited in semantic interpretation [11].

Robinson et al. (2015) provided practical insights on graph database design and applications. They highlighted that keyword-based search works for simple node property matching but falters in complex path queries. Their case studies underscored the need for semantic enhancements to improve accuracy [4].

Martínez-Bazan et al. (2007) developed efficient query processing methods in large graph databases. They proposed algorithms for indexing and retrieving node and edge data using keywords. However, they noted that accuracy declines when queries require contextual understanding beyond keywords [14].

Sun et al. (2011) examined keyword search over graph-structured data and found challenges in ensuring high recall. They proposed hybrid techniques combining keyword matching with structural analysis. Their results showed moderate improvements but underscored persistent accuracy limitations in keyword-only methods [8].

Fan et al. (2012) introduced approximate matching approaches for keyword search in graphs. Their methods improved recall by tolerating variations in keywords. They reported trade-offs between improved recall and reduced precision, highlighting the inherent challenges of keyword-based search [7].

In relational databases, keyword-based search often involves using full-text indices, SQL LIKE conditions, or inverted file structures. Studies have shown that this approach achieves high precision when the query terms match exactly with indexed data, but recall suffers significantly due to vocabulary mismatch and lack of semantic interpretation. Ranking mechanisms and entropy-based query selection methods have been proposed to improve user-perceived accuracy, but keyword-based systems remain limited in understanding context or synonyms.

In graph databases, keyword-based search involves matching query terms against node or edge properties. While graph structures allow efficient traversal and flexible pattern matching, keyword-only approaches often fail to capture deeper semantic connections between entities. Hybrid methods that combine structural analysis with keyword matching have been explored, showing moderate improvements in accuracy but still suffering from fundamental limitations inherent to keyword-based search. Overall, research consistently demonstrates that while keyword-based methods are fast and computationally simple, they struggle to achieve high accuracy in both relational and graph database contexts when semantic meaning and user intent are important.

3.2. Accuracy Calculation for Semantic Search in Relational and Graph Databases

In semantic search, accuracy is typically evaluated by comparing the system's returned results to a gold standard set of relevant results, considering not just keyword matches but also semantic similarity and context. Methods often involve calculating precision, recall, and F1-score based on whether returned results are semantically relevant. Additionally, semantic search systems often use similarity scores (e.g., cosine similarity between embeddings) and define accuracy thresholds: results with similarity above a certain threshold are considered correct. Evaluation may also include human judgment or benchmark datasets with relevance labels, combining quantitative similarity metrics with qualitative relevance assessments. Studies often report Mean Average Precision (MAP) or Normalized Discounted Cumulative Gain (NDCG) for ranked semantic retrieval results.

Bhalotia et al. (2002) proposed BANKS, a system that supports keyword search and browsing in relational databases by leveraging a graph-based representation of tuples and joins. They demonstrated how the system ranks and returns top answers using network flow algorithms. Experiments showed improved retrieval quality compared to pure keyword matching [19].

Tran et al. (2009) developed methods to provide top-k answers for schema-based keyword search, combining structural joins with ranking models. Their work addressed efficiency and accuracy challenges in large relational datasets. They found their approach consistently outperformed baseline keyword search systems [10].

Chakrabarti et al. (2006) discussed integrating semantic mapping into relational databases, highlighting techniques to improve semantic understanding and thereby retrieval accuracy. Their framework incorporated ontologies into query rewriting. Results showed significant improvements in precision [15].

Ding et al. (2004) surveyed methods for embedding semantic knowledge in database systems. They reviewed approaches ranging from schema mapping to semantic caching. They emphasized the positive impact of semantic enrichment on query accuracy [18].

Zou et al. (2014) presented store, a graph-based system supporting SPARQL queries over RDF data using semantic-aware indexing. They showed their method increased accuracy in query answering on large graphs. Their experiments validated the scalability and precision of their approach [5].

He et al. (2007) proposed techniques for pattern-based query answering in graph databases using indexing and efficient graph traversal. They improved the retrieval accuracy of subgraph matching tasks. Their evaluation showed reduced query time and improved result relevance [12].

Zhang et al. (2016) combined semantic similarity measures with graph structure features for query answering. They developed a hybrid model integrating embeddings and graph pattern matching. Their experiments demonstrated higher accuracy in graph query results [3].

Cheng et al. (2010) reviewed keyword search in databases, identifying limitations in keyword-based systems and discussing the role of semantics. They concluded that semantic augmentation improves both recall and precision. Their survey pointed to future directions involving hybrid methods.

Han et al. (2013) studied semantic query processing in graph databases using approximate matching. They proposed algorithms allowing partial match retrievals. They found a trade-off between precision and recall but overall gains in user-perceived accuracy [9].

Zhang et al. (2019) explored semantic-aware query processing in heterogeneous graph databases. They developed methods to incorporate schema and ontology knowledge into query translation. Their system achieved high accuracy across diverse datasets [1].

In relational databases, semantic-based search incorporates natural language processing, tokenization, and vector similarity measures like cosine similarity to improve accuracy over traditional keyword matching. Studies show that integrating semantic models or external semantic engines leads to better handling of synonyms, context, and user intent, thereby improving recall and precision. However, challenges include computational complexity and the need for consistent schema mapping between semantic layers and structured relational data.

In graph databases, semantic-based search leverages the inherent graph structure to encode semantic relationships among nodes and edges. Research highlights that combining semantic similarity measures with graph traversal patterns significantly enhances accuracy by exploiting relational paths in the graph. Methods often use cosine similarity or machine learning models to compare embeddings of query patterns against graph data, achieving high contextual relevance ([9],[10]). Despite these improvements, studies note that determining optimal similarity thresholds and managing scalability remain open challenges in graph-based semantic search.

We describe Comparative Tables for keyword based, semantic based search for relational and graph database are described in Table 1, 2 and 3.

Table 1: Keyword vs. Semantic-Based Search in Relational Databases

Aspect	Keyword-Based Search	Semantic-Based Search
Matching Mechanism	Exact string or partial match	Contextual similarity, vector-based matching
Handling Synonyms	Weak	Strong
Context Awareness	Low	High
Accuracy	High precision, low recall	Improved recall and semantic accuracy
Computational Cost	Low	Higher
Flexibility	Limited	Moderate
Scalability	High	Moderate

Table 2: Keyword vs. Semantic-Based Search in Graph Databases

Aspect	Keyword-Based Search	Semantic-Based Search
Matching Mechanism	Node/property term match	Graph pattern traversal with similarity scoring
Handling Relationships	Limited	Strong
Context Awareness	Low	High
Accuracy	Moderate	Higher
Computational Cost	Low	Higher
Flexibility	Moderate	High
Scalability	High	Depends on graph size

Table 3: Cross-Comparison of Relational vs. Graph Databases (Keyword vs. Semantic-Based)

Feature	Relational DB Keyword	Relational DB Semantic	Graph DB Keyword	Graph DB Semantic
Schema Flexibility	Low	Moderate	High	High
Relationship Modeling	Weak	Moderate	Strong	Strong
Context Awareness	Low	High	Low	High
Computational Cost	Low	High	Low	High
Accuracy	Moderate	High	Moderate	High
Best Use Cases	Structured queries	Exploratory queries	Simple paths	Complex graph analysis

The comparative analysis and literature review reveal that while keyword-based approaches offer simplicity and computational efficiency, they struggle to capture user intent and semantic nuances, especially in queries involving synonyms, context, or relational paths. In relational databases, semantic-based methods improve accuracy by incorporating vector-based similarity measures and external semantic en-

gines, though they introduce computational overhead. In graph databases, semantic-based search is particularly powerful, leveraging the graph structure to interpret relationships and context, significantly boosting accuracy for complex and interconnected queries. However, semantic methods in both paradigms face challenges such as determining optimal similarity thresholds, managing scalability with large datasets, and integrating models effectively with existing systems. These findings underscore the need for hybrid approaches that combine the efficiency of keyword-based systems with the contextual strength of semantic-based methods, to achieve robust, accurate, and scalable information retrieval across diverse domains.

4. Conclusion

In this paper, we have examined accuracy calculation methods in keyword-based and semantic-based search approaches within relational and graph databases. Keyword-based methods offer simplicity and efficiency but often fail to capture user intent and semantic context, limiting their accuracy. Semantic-based methods address this gap by leveraging advanced techniques such as semantic similarity measures, graph pattern analysis, and natural language processing, resulting in higher accuracy but with increased computational complexity. Graph databases, with their inherent capability to represent relationships, further enhance the potential of semantic search. Despite improvements, challenges remain in optimizing thresholds, handling data sparsity, and balancing computational cost with precision. Future research should continue to develop hybrid methods that combine the strengths of both approaches to achieve accurate, efficient, and scalable information retrieval.

Additionally, it is recommended that future work explores the integration of machine learning models and domain-specific ontologies into semantic-based systems. Researchers should also investigate user-driven threshold tuning mechanisms to adaptively optimize similarity measures for different domains. Finally, building comprehensive benchmark datasets for both relational and graph-based semantic search will be crucial for standardized evaluation of accuracy across approaches.

5. References

1. Zhang, S., Lee, W., & Li, J. (2019). Semantic-aware query processing in heterogeneous graph databases. IEEE Transactions on Knowledge and Data Engineering, 31(4), 710-723.
2. Zhang, Y., & Balog, K. (2017). Entropy-based query selection for active learning in entity search. ACM Transactions on Information Systems (TOIS), 36(1), 1-37.
3. Zhang, J., Zhang, Y., & Zhang, C. (2016). Combining semantic similarity and graph structure for effective query answering. Journal of Web Semantics, 40, 30-45.
4. Robinson, I., Webber, J., & Eifrem, E. (2015). Graph Databases: New Opportunities for Connected Data. O'Reilly Media, Inc.
5. Zou, L., Ooi, B. C., Huang, K. L., Chen, Q., & Dai, D. (2014). gStore: Answering SPARQL queries by leveraging a graph-based index. Proceedings of the VLDB Endowment, 5(12), 1774-1785.
6. Han, W., Lee, J., & Lee, H. (2013). Semantic query processing using approximate matching in graph databases. Information Sciences, 245, 120-138.
7. Fan, W., Li, J., Ma, S., Tang, N., & Wu, Y. (2012). Adding regular expressions to graph reachability and pattern queries. In Proceedings of the IEEE International Conference on Data Engineering (ICDE), 39-50.

8. Sun, Y., Han, J., Zhao, P., Yin, Z., Cheng, H., & Wu, T. (2011). RankClus: Integrating clustering with ranking for heterogeneous information network analysis. Proceedings of the VLDB Endowment, 2(1), 565-576.
9. Cheng, J., Ke, Y., Chu, S., & Cheng, R. (2010). Efficient query processing on graph databases. ACM Transactions on Database Systems (TODS), 35(4), 1-48.
10. Tran, T., Wang, H., & Haase, P. (2009). Top-k answers for schema-based keyword search over relational databases. In Proceedings of the IEEE International Conference on Data Engineering (ICDE), 221-232.
11. Angles, R., & Gutierrez, C. (2008). Survey of graph database models. ACM Computing Surveys (CSUR), 40(1), 1-39.
12. He, H., Wang, H., Yang, J., & Yu, P. S. (2007). BLINKS: Ranked keyword searches on graphs. In Proceedings of the ACM SIGMOD International Conference on Management of Data, 305-316.
13. Li, C., Ooi, B. C., & Wang, S. (2007). Supporting efficient keyword search in relational databases. ACM Transactions on Database Systems (TODS), 33(1), 1-39.
14. Martínez-Bazan, N., et al. (2007). Efficient graph management based on multi-level partitioning. In Proceedings of the VLDB Endowment, 2(1), 1-12.
15. Chakrabarti, S., Chaudhuri, S., & Xin, D. (2006). Integrating semantics into keyword-based search over relational databases. In Proceedings of the IEEE International Conference on Data Engineering (ICDE), 523-534.
16. Zobel, J., & Moffat, A. (2006). Inverted files for text search engines. ACM Computing Surveys (CSUR), 38(2), 1-56.
17. Chaudhuri, S., Ganti, V., Kaushik, R., & Ramakrishnan, R. (2004). Using statistics to generate efficient query plans in relational keyword search. Proceedings of the VLDB Endowment, 27(1), 102-113.
18. Ding, L., Finin, T., Joshi, A., Pan, R., Cost, R. S., Peng, Y., ... & Reddivari, P. (2004). Swoogle: A search and metadata engine for the semantic web. In Proceedings of the 13th ACM International Conference on Information and Knowledge Management (CIKM), 652-659.
19. Bhalotia, G., Hulgeri, A., Nakhe, C., Chakrabarti, S., & Sudarshan, S. (2002). Keyword searching and browsing in databases using BANKS. In Proceedings of the 18th International Conference on Data Engineering (ICDE), 431-440.
20. Raghavan, V. V., & Wong, S. K. (1986). A critical analysis of vector space model for information retrieval. Journal of the American Society for Information Science, 37(5), 279-287.
21. Salton, G., Wong, A., & Yang, C. S. (1975). A vector space model for automatic indexing. Communications of the ACM, 18(11), 613-620.