Cognitive Retrieve: Empowering Document Retrieval with Semantics and Domain-Specific Knowledge Graph

Apurva Kulkarni[1,†], Chandrashekar Ramanathan[2] and Vinu E Venugopal[3]

[1,2,3] *International Institute of Information Technology, Bangalore, India*

Abstract

As the data landscape continues to expand, the task of identifying relevant documents becomes increasingly complex, especially when dealing with diverse and varied data sources. Traditional keyword-based search systems struggle to capture the subtle contextual meaning of search queries. Semantic-based search, leveraging open data knowledge graphs, offers a solution by understanding contextual meaning. However, its effectiveness relies heavily on the quality and completeness of the underlying data used to define these semantics. However, incomplete data can lead to spurious results and a lack of relevance in the retrieved documents. To bridge this gap between user search interest and retrieval outcomes, we propose integrating domain-specific alignment into the search process. Our research aims to achieve this through the development of a semantic-driven data processing pipeline, laying the foundation for seamless semantic-oriented retrieval. This approach includes metadata extraction, considering domain-specific keywords and structural metadata from heterogeneous data sources. We enhance metadata by identifying latent terms using language models. Furthermore, we incorporate latent concepts and domain-specific information gathered from domain experts into a special knowledge graph construct- a 'concept graph'. Our primary focus is on identifying relevant concepts from this graph, aligning with semantic and contextual aspects of the specified search intent. Our proposed document retrieval system, which combines the concept graph with semantics, is implemented using data from the Government of Karnataka, India. This approach addresses the administrative need to extract relevant documents from data silos, offering an alternative approach to traditional methods. Extensive evaluations demonstrate the proposed system's superior performance in terms of true positive results compared to baseline systems like Lucene, Elasticsearch, and Doc2Vec.

Keywords

Knowledge-based Document Retrieval, Semantic Linking, Semantic Concept Graph, Group Steiner Tree

1. Introduction

In the field of document retrieval, it is essential to have a deep understanding of the context behind search queries in order to effectively retrieve relevant documents [1]. Traditional methods of document retrieval have evolved from simple pattern matching to incorporating artificial intelligence (AI) (including knowledge graphs, and machine learning (ML) algorithms that facilitate to consideration of cognitive aspects of search intent). The conventional document retrieval process involves three main tasks: *indexing*, *searching*, and *ranking*. The accuracy of a document retrieval (DR) system primarily relies on the searching ability to retrieve relevant concepts from the document collection.

In the realm of data retrieval, traditional keyword-based search methods have limitations in capturing context and semantics. The proposed research explores concept-based and semantic-based search approaches, shedding light on their potential to improvise document retrieval by incorporating domain-specific concepts and semantic understanding.

The existing literature exploring document retrieval methods predominantly stems from the following key approaches: keyword-based, concept-based, and semantic-based techniques. Keyword-based search employs string-matching techniques to retrieve documents, but this approach limits the system's ability to search beyond the exact search words and capture semantic relationships, thus missing out on contextually relevant documents [2, 3]. Concept-based searching enables the inclusion of domain-specific aspects during the search process. Instead of indexing documents based on individual terms, documents are indexed based on concepts, allowing for the retrieval of contextually relevant documents [4]. This approach relies on the accurate representation and comprehensive coverage of domain knowledge in the form of concepts. The semantic-based search approach [5, 6], on the other hand, focuses on understanding the underlying meaning of the search intent. It typically utilizes a common knowledge base to achieve this understanding [7]. In general, by leveraging up-to-date and in-depth knowledge, a DR system is better equipped to comprehend semantics, leading to the retrieval of more relevant documents.

Consider the following scenario: Let D denote the

Workshop on Enterprise Knowledge Graphs using Large Language Models, Oct 21, 2023, Birmingham, UK

[†] Corresponding author.

apurva.kulkarni@iiitb.ac.in (A. Kulkarni); rc@iiitb.ac.in (C. Ramanathan); vinu.ev@iiitb.ac.in (V. E. Venugopal)
0000-0002-9215-2049 (A. Kulkarni); 0000-0002-3330-8365 (C. Ramanathan); 0000-0003-4429-9932 (V. E. Venugopal)

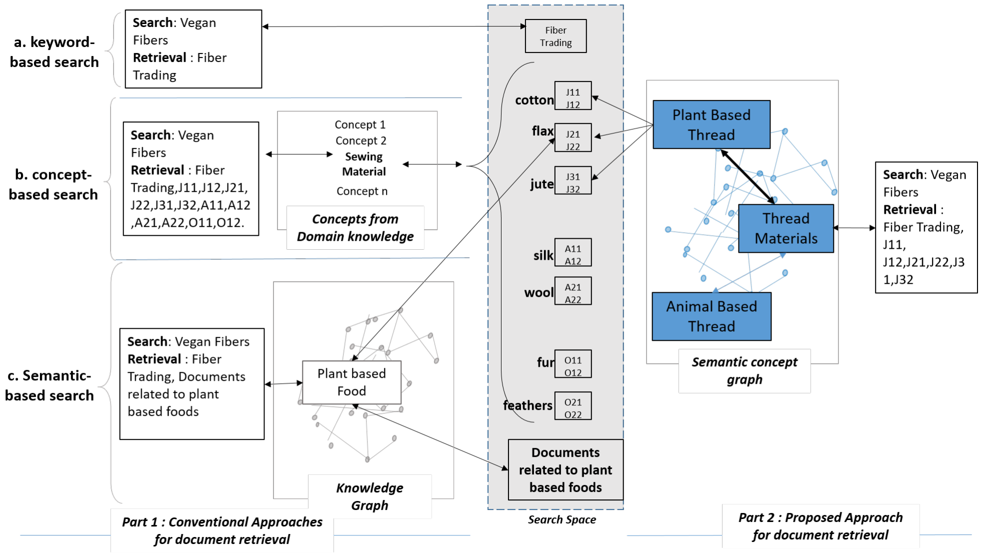

Figure 1: Motivation: Leveraging domain knowledge and semantics for document retrieval

set of documents as $d_1, d_2, \ldots, d_n$, where each document d_i is a subset of a universal set of words W. Similarly, let S represent the search string, which is a set of specific words contained within W. Keyword-based search retrieves documents D_K, defined as $D_K = \{d_i \in D | S \cap d_i \neq \emptyset\}$, where the relevance of documents in D_K is determined by the frequency and proximity of search words within each document d_i.

Concept-Based Search retrieves documents D_C, defined as $D_C = \{d_i \in D | Concept(S) \cap Concept(d_i) \neq \emptyset\}$. Here, $Concept(d_i)$ refers to the set of concepts associated with document d_i, and $Concept(S)$ represents concepts related to the search string. Relevance in D_C relies on the extent of overlap between conceptually related words within documents and the search query.

Semantic-Based Search identifies documents D_S, with $D_S = \{d_i \in D | \exists words_d \in d_i, \exists words_s \in S : Semantic\_Similarity(words_d, words_s)\}$. These are documents within D containing words that are semantically related to those in the search string. Relevance in D_S considers the degree of overlap between semantically related words within documents and the search query, taking into account the intended semantics and context of the query.

In Fig. 1, Part 1 provides an illustrative example of the document retrieval process for the search string 'Vegan Fiber' using different approaches. When employing keyword-based methods for searching, only the file Fiber Trading is retrieved, as it contains the specific term 'Fiber'. All the other files related to different fibers remain unidentified. In the *concept-based* retrieval approach, the concept of Sewing Materials is closest concept associated with search term *'Vegan Fiber'* and hence it retrieves Jute (J11, J12), Flax (J21, J22), Cotton (J31, J32), Silk (A11, A12), Wool (A21, A22), Fur (O11, O12), Feathers (O21, O22), and Fiber Trading. On the other hand, the *semantic-based* search, the most relevant entity to the search string is vegan food is used to fetch the documents- Flax (J21, J22).

It has been observed that relying solely on domain knowledge or semantics is not sufficient for retrieving relevant documents. Therefore, in this research, the aim is to combine domain knowledge, represented as knowledge graphs capturing the concepts from domain knowledge as concept graphs, with semantics to enhance document retrieval. By leveraging semantics, the research seeks to enrich the domain knowledge and improve the identification of relevant concepts, which in turn leads to more accurate document retrieval[8]. In Figure 1, Part 2, the diagram portrays the rationale behind the proposed research that combines domain knowledge and semantic analysis for document retrieval. For instance, searching for 'Vegan Fibers' yields the concept 'Plant Based Threads,' resulting in the retrieval of relevant documents like 'Jute,' 'Flax,' and 'Cotton.' The proposed approach, Semantic-based Document Retrieval (SemDR), optimizes retrieval by blending domain expertise with semantics.

2. Research Contribution

The proposed cognitive retrieval of documents consists of three main steps: *Semantic Pipeline, Constructing Semantic Concept Graph,* and *Semantically Retrieving the*

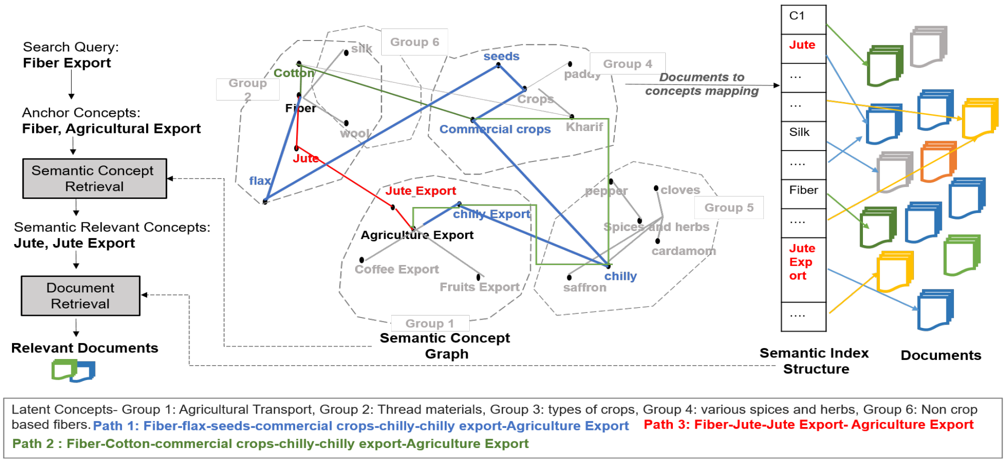
Search Query:
Fiber Export
Anchor Concepts:
Fiber, Agricultural Export
Semantic Concept Retrieval
Semantic Relevant Concepts:
Jute, Jute Export
Document Retrieval
Relevant Documents
Group 6
silk
Cotton
Group 2
Fiber
wool
Jute
flax
seeds
paddy
Group 4
Crops
Commercial crops
Kharif
Documents to concepts mapping
Jute Export
chilly Export
Agriculture Export
Coffee Export
Fruits Export
Group 1
Semantic Concept Graph
pepper
cloves
Group 5
Spices and herbs
cardamom
chilly
saffron
C1
Jute
…
…
Silk
….
Fiber
….
Jute Exp ort
….
Semantic Index Structure
Documents
Latent Concepts- Group 1: Agricultural Transport, Group 2: Thread materials, Group 3: types of crops, Group 4: various spices and herbs, Group 6: Non crop based fibers.
Path 1: Fiber-flax-seeds-commercial crops-chilly-chilly export-Agriculture Export
Path 3: Fiber-Jute-Jute Export- Agriculture Export
Path 2 : Fiber-Cotton-commercial crops-chilly-chilly export-Agriculture Export

Figure 2: Proposed Approach: The use of Group Steiner Tree to identify optimal path with relevant concepts in the document retrieval process

documents and linking them to perform exploratory analytics. The Algorithm1 elaborates the overall process with the functional flow of the proposed system and Figure 2 depicts the architectural flow with an illustration.

Semantic Pipeline. The primary task of the semantic pipeline is to link documents to the concepts identified from domain knowledge. The of linking is based on the semantics and leverages a knowledge graph built over domain knowledge. The first step in data processing is handling the heterogeneity associated with data sources. The pipeline considers a set of Documents D and generates the metadata for each $d_i \in D$. Along with capturing structural metadata like type, format, size etc.; the emphasis is on capturing the words for each d_i that describes the document. Further, the lexical database WordNet is used to extend the captured words to enrich the $Metadata(d_i)$.

The other key input to the system is the domain knowledge describing nuances in the form of concepts, the description of each concept and the relationships among concepts. A similar process is applied by the semantic pipeline to capture metadata for each concept $Meatadata(c_i)$. To understand the semantic bonding and link the document d_i with respective concepts c_i, the proposed approach uses the large open data corpus. The intuition behind using open data is to capture underlying, abstract relationships beyond direct semantic similarity, enhancing the depth of mapping.

The corpus from the Wikipedia document dump is leveraged for clustering the content from Wikipedia documents and extracting top keywords that characterize each cluster[9]. Subsequently, the metadata of both documents and concepts are associated with a common reference point - cluster.

Semantic Similarity: To understand the semantic similarity between two words, the proposed system employs the Wu-Palmer score [10, 11]. Wu-Palmer considers the depth of common ancestors in a taxonomy (like WordNet), making it better for capturing nuanced semantic relationships in word pairs compared to simpler methods like path-based similarity.

The semantic similarity function serves a technical role in linking documents with concepts and concurrently constructing a semantic concept graph. During the linking operation, it identifies the cluster with the highest semantic similarity score, which is then utilized to establish mappings between documents and concepts. This similarity assessment is based on the comparison of semantic vectors derived from the top words within the cluster and the metadata of documents or concepts. The linking process can be defined as,

$(Metadata(d_i) \mapsto x) \wedge (Metadata(c_i) \mapsto x) \Rightarrow d_i \mapsto c_i$

where d_i represents the document, c_i represents the concept, x represents the cluster, $\mapsto$ denotes the mapping relationship established using semantic similarity score.

Constructing Semantic Concept Graph. To enable search based on domain knowledge and semantics, the essential steps involve creating a semantic concept graph specific to the domain and implementing a traversal algorithm to identify the most pertinent concepts from the graph. The previous section elaborates on concepts and metadata for each concept. The concepts given as input (as a part of the domain information) serve as the nodes in the concept graph. The relationships between these concepts are captured in two ways- the relationship among concepts mentioned in the domain information and another way by using a semantic similarity score.

Concepts with higher semantic similarity (Wu-Palmer scores) indicate a closer semantic relationship. Based on the semantic similarities, the system groups together concepts that share common contexts. These logical entities, referred to as latent concepts, represent cohesive groups of semantically related concepts within the concept graph.

Let $G = (C, R)$ represent a semantic concept graph, where: C is a finite set of concepts, which can be partitioned into two subsets: C_{direct}, comprising concepts derived directly from domain knowledge, and C_{latent}, representing logical nodes denoting groups of similar concepts. Mathematically, $C = C_{direct} \cup C_{latent}$. R is a finite set of undirected edges between pairs of concepts. These relations consist of two distinct subsets: $R_{contextual}$, responsible for capturing hierarchical connections and dependencies among various concepts within the domain, and $R_{semantic}$, linking concepts exhibiting substantial semantic similarity. Formally, $R = R_{contextual} \cup R_{semantic}$. The semantic concept graph, incorporating the latent concepts and their connections, is utilized to identify concepts that are relevant to the search interest. By traversing the graph, the system can determine the most relevant concepts related to the search query, which in turn helps in retrieving pertinent documents [12].

Identifying Relevant Concepts using Group Steiner Tree: The concepts that hold the highest similarity score with the search string are used as *anchor nodes* (aka *terminal nodes*). The anchor nodes serve as starting points for identifying the most relevant concepts. To determine the most relevant concepts, the system analyzes the paths within the semantic concept graph that connect these anchor concepts. The goal is to find an optimal path that covers the maximum possible (ideally all) anchor concepts [13, 14]. To achieve this, the research work leverages a group Steiner tree approach. This approach takes into account the relaxed anchor concepts, which are replaced by the corresponding latent concepts (semantic groups).

By leveraging semantic groups, the system enhances its ability to identify semantically relevant concepts and establish a path that covers all the anchor concepts. The group Steiner tree algorithm employed in the system traverses the semantic groups within the semantic concept graph. It aims to identify the path that includes all the anchor concepts, ensuring that no anchor concept is left out. By traversing the semantic groups and establishing the optimal path, the algorithm identifies a set of concepts that are considered the most relevant. These concepts play a crucial role in addressing the user's search intent and retrieving pertinent documents. The retrieved documents are linked semantically and used for further analyses.

The proposed approach is illustrated in Fig. 2 through a case study. The search term 'Fiber Export' is closely associated with the concepts 'Fiber' and 'Agricultural Export', which are considered as anchor concepts. The latent concepts *'Group 1' - 'Agricultural Transport'* and *'Group 2' - 'Thread Materials'* are taken into account to identify the Group Steiner Tree that encompasses all anchor concepts. Multiple paths exist that cover the anchor concepts, but the optimal path is chosen based on the inclusion of concepts from the latent concepts. This is because these concepts are semantically related to the anchor concepts and indirectly linked to the search term. In the case study, path 3 is chosen as an optimal path and that gives relevant concepts 'Jute' and 'Jute' Export to fetch the relevant documents.

Algorithm 1 Generic Framework

```
Input: Domain Knowledge $D_K$, Search String $S$, Documents $D = \{d_1, d_2, \ldots, d_n\}$
Output: Relevant_Documents $R_D$
$Metadata(D) \leftarrow Semantic\_Pipeline(Documents)$
$Metadata(C) \leftarrow Semantic\_Pipeline(Concepts)$
$C\_linked\_D \leftarrow Semantic\_Linking(Metadata(C), Metadata(D))$
$CG \leftarrow Semantic\_Concept\_Graph(D_K, Metadata(C))$
$Relevant\_Concepts \leftarrow Group\_Steiner\_Tree(CG, S)$
$R_D \leftarrow Document\_Retrieval(C\_linked\_D, Relevant\_Concepts)$
Return Relevant Documents $R_D$
```

3. Experimental Setup

The system is developed using authentic data supplied by the Government of Karnataka, with the aim of acquiring semantically relevant documents to support administrative decision-making[15]. Table 3. (b) illustrates how incorporating knowledge enhances the outcome of document retrieval. The initial version (v1) lacks any knowledge component, while the subsequent version (v2) improves by including basic domain knowledge, encompassing concepts and relations. The following version (v3) takes into account relations and incorporates new facts into the knowledge graph. Finally, the ultimate

Table 1
System Evolution: The impact of knowledge component in document retrieval

Version	Description of the System	Average Documents Retrieved over 170 queries
V1	No semantic component	12
V2	Addition of concepts and relationships	40
V3	Knowledge updated using fact generation	50
V4	New facts added based on semantic knowledge	62

version (v4) further enriches the knowledge by considering semantic relevance. The figures in the table depict a trend highlighting the progressive impact of knowledge on document retrieval.

To evaluate the effectiveness of the proposed system, a comparative study is conducted considering three baseline systems: Lucene [16], Doc2Vec [17], and Elastic Search [18]. The performance evaluation involves utilizing a benchmark that consists of 170 search queries manually gathered from both domain experts and system users. A subset of these queries is also formulated to represent government use cases. These queries are then employed to conduct tests on real-world data[1], serving the purpose of assessing the system's performance. These queries cater to various user needs and include QS1, which involves straightforward keyword searches; QS2, enhancing queries with geographic tags to find location-specific data; QS3, adding temporal tags to search for time-specific information; QS4, allowing more complex indirect queries with filters and operators; and QS5, a collaborative approach combining domain experts' knowledge with direct and indirect search terms. Rigorous evaluation and verification procedures are employed, involving domain experts, to ensure the validity and accuracy of the obtained results. In Figure 3, the representation showcases the True Positive values across various baseline systems concerning distinct search queries that include location and time tags. Notably, the performance of these systems shows variations based on the query types. SemDR consistently demonstrated the most stable overall performance, successfully retrieving relevant results for queries of all types. LUCENE and Elastic Search also demonstrated better performance, especially when it came to Direct Query searches complemented by additional location and time tags (QS2 and QS3). In contrast, DOC2VEC holds comparatively lower performance levels across all query types.

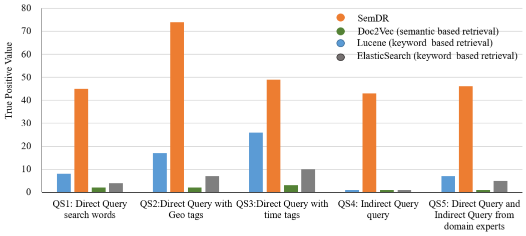

Figure 3: System Performance with respect to different query types

Table 2
Performance Analysis considering Precision, Recall, Accuracy and F1-Score

System	Precision	Recall	Accuracy	F1-Score
SDR	90	88	82	88.98
ES	78	11	16	19.21
LUCENE	80	19	24	30.70
DOC2VEC	34	3	7	5.51

The experimental findings reveal significant advancements in the SemDR system compared to the baseline systems. Table 3.(a) indicates the performance evaluation of four different systems (SemDR, Elastic Search, Lucene, and Doc2Vec) based on four different evaluation metrics (precision, recall, accuracy, and F1-score). In document retrieval, True Positive (TP) counts correctly retrieved documents, while False Positive (FP) signifies irrelevant documents retrieved by the system. False Negative (FN) represents relevant documents missed, and True Negative (TN) denotes documents neither retrieved nor relevant. Type 1 error, or false positive, occurs when irrelevant documents are retrieved. Type 2 error, or false negative, happens when relevant documents are missed. Key performance metrics include

$$Accuracy = (TP + TN)/(TP + TN + FP + FN)$$

$$Precision = TP/(TP + FP)$$

$$Recall/Sensitivity = TP/(TP + FN)$$

$$F1{-}score = 2*((precision*recall)/(precision+recall))$$

which combine to evaluate retrieval system effectiveness. To ensure a standardized comparison scale, all values are expressed as percentages. The findings indicate that SemDR retrieves a high percentage of relevant documents with a good balance of precision and recall. The precision and accuracy of the SemDR system reach impressive levels of 90% and 82% respectively, indicating promising improvements in information retrieval capabilities.

4. Conclusion

The objective of the proposed work is to enhance the document retrieval system by incorporating a deeper understanding of the underlying cognitive aspects of search intent by using concept-based knowledge graphs. The proposed approach addresses the limitations of traditional search methods by integrating domain-specific information and semantic understanding into the document retrieval process, even in heterogeneous environments. The work includes the implementation details and experimentation with real-world search queries to demonstrate the practicality and effectiveness of the proposed system. By leveraging cognitive understanding and domain knowledge, the system aims to bridge the

[1] https://ndap.niti.gov.in/info?tab=sampleusecases

gap between the user's search intent and the documents retrieved in the document retrieval environment.

5. Acknowledgments

This work was supported by the Government of Karnataka and Karnataka Innovation & Technology Society, Dept. of IT, BT and S&T, Govt. of Karnataka, India, vide GO No. ITD 76 ADM 2017, Bengaluru; Dated 28.02.2018.

References

[1] H. Schutze, C. D. Manning, P. Raghavan, Introduction to information retrieval, volume 39, Cambridge University Press Cambridge, 2008.

[2] M. Mitra, B. Chaudhuri, Information retrieval from documents: A survey, Information retrieval 2 (2000) 141–163.

[3] X. Wang, X. Zhang, D. Cai, Bert-keyword: A language model for keyword-based document retrieval, arXiv preprint arXiv:2101.00316 (2021).

[4] J. F. Sowa, B. Moulin, G. MINEAU, Conceptual graphs for knowledge representation, Lecture Notes in AI 699 (1993).

[5] P. Xiong, X. Li, X. Wang, H. Ji, A semantic-based approach to document retrieval, arXiv preprint arXiv:1708.03445 (2017).

[6] J. Xiao, W. Zhang, T. Liu, S. Ma, Semantic-based document retrieval method using wordnet and ontology, Information Retrieval Journal 21 (2018) 209–232.

[7] R. Reinanda, E. Meij, M. de Rijke, et al., Knowledge graphs: An information retrieval perspective, Foundations and Trends® in Information Retrieval 14 (2020) 289–444.

[8] A. Kulkarni, C. Ramanathan, V. E. Venugopal, Ontology mediated document retrieval for exploratory big data analytics, in: 2023 IEEE 17th International Conference on Semantic Computing (ICSC), IEEE, 2023, pp. 100–103.

[9] A. Kulkarni, C. Ramanathan, V. E. Venugopal, Semantics-aware document retrieval for government administrative data, International Journal of Semantic Computing (2023).

[10] Z. Wu, M. Palmer, Verb semantics and lexical selection, arXiv preprint cmp-lg/9406033 (1994).

[11] T. Wei, Y. Lu, H. Chang, Q. Zhou, X. Bao, A semantic approach for text clustering using wordnet and lexical chains, Expert Systems with Applications 42 (2015) 2264–2275.

[12] J. Han, Y. Wang, S. Zhang, Y. Zhu, C. Shi, P. S. Yu, Group steiner tree for knowledge graph completion, in: Proceedings of the 27th ACM International Conference on Information and Knowledge Management (CIKM), 2018.

[13] X. Lu, S. Pramanik, R. Saha Roy, A. Abujabal, Y. Wang, G. Weikum, Answering complex questions by joining multi-document evidence with quasi knowledge graphs, in: Proceedings of the 42nd International ACM SIGIR Conference on Research and Development in Information Retrieval, 2019, pp. 105–114.

[14] J. Jung, J. Kang, U. Kang, Efficient steiner tree computation for large-scale knowledge graphs, in: Proceedings of the 28th ACM International Conference on Information and Knowledge Management (CIKM), 2019.

[15] A. Kulkarni, P. Bassin, N. S. Parasa, V. E. Venugopal, S. Srinivasa, C. Ramanathan, Ontology augmented data lake system for policy support, in: Big Data Analytics in Astronomy, Science, and Engineering: 10th International Conference on Big Data Analytics, BDA 2022, Aizu, Japan, December 5–7, 2022, Proceedings, Springer, 2023, pp. 3–16.

[16] A. Białecki, R. Muir, G. Ingersoll, L. Imagination, Apache lucene 4, in: SIGIR 2012 workshop on open source information retrieval, 2012, p. 17.

[17] J. H. Lau, T. Baldwin, An empirical evaluation of doc2vec with practical insights into document embedding generation, arXiv preprint arXiv:1607.05368 (2016).

[18] C. Gormley, Z. Tong, Elasticsearch: the definitive guide: a distributed real-time search and analytics engine, O'Reilly Media, Inc., 2015.