*Journal of Information & Optimization Sciences*
ISSN 0252-2667 (Print), ISSN 2169-0103 (Online)
Vol. 45 (2024), No. 8, pp. 2179–2192
DOI : 10.47974/JIOS-1781

# Part of speech-based semantic similarity for RDF Predicate Selection : A benchmarking approach

Aatif Ahmad Khan [§]
Sanjay Kumar Malik [†]
*University School of Information, Communication and Technology*
*Guru Gobind Singh Indraprastha University*
*New Delhi*
*India*

Vanita Jain*
*University of Delhi*
*New Delhi*
*India*

**Abstract**

RDF predicate selection and RDF subject identification are two key steps for effectively processing the natural language queries by the semantic question answering systems. For subject identification, techniques such as Named Entity Recognition are used. For the relatively complex task of predicate selection, Natural Language Processing techniques can be utilized along with knowledge bases to select the semantically matching relationships. Semantic features can be extracted from queries utilizing phrase embeddings of the transformer models. In this paper firstly, a novel Predicate Selection approach on RDF Knowledge Bases has been proposed and implemented utilizing BERT-based phrase embeddings and custom logic for key part-of-speech token combinations. Secondly, the implemented prototype is benchmarked over combinations of key part of speech tokens and stop words for predicate selection accuracy over YAGO Date Facts, demonstrating the effectiveness of our approach. Benchmarking results also highlight the importance of verb stop words for the task of predicate selection.

***Subject Classification:*** *(2010) 68T30, 68T35, 68T50.*

***Keywords:*** *RDF predicate selection, Knowledge base, Semantic similarity, Part of speech tagging, Stop word removal, Question answering.*

[§] *E-mail:* **aatif1992@gmail.com**
[†] *E-mail:* **sdmalik@hotmail.com**
* *E-mail:* **vjain@electronics.du.ac.in (Corresponding Author)**

## 1. Introduction

In the era of generative AI, question answering systems are being augmented with semantic features to process the queries effectively. These systems are also benefitted by utilizing the knowledge present in the structured repositories or knowledge graphs. Such repositories contain facts about real-world concepts and their relationships which are machine understandable. The end goal of the query processing systems is to respond with highly precise results while maintaining a high recall. Advancements in the answer assessment techniques further benefit these systems [1].

Semantic web is the next generation of web, capable of representing and interconnecting information in a machine understandable manner thus forming the basis for knowledge graphs. It uses the Resource Description Framework (RDF) model to represent interconnected knowledge in the open world. RDF uses triple notation (subject, predicate, object) to represent concepts and their relations with Universal Resource Identifiers (URI). Ontology is another key technology of the semantic web stack that define interoperable vocabularies to model systems and provide restrictions on entities and their relationships. The structured data in RDF format can be queried effectively using SPARQL queries.

To utilize factual knowledge contained in the knowledge bases, query processing systems use Natural Language Processing (NLP) techniques to transform the natural language query, identify the subject entity and the relationship [2]. In terms of semantics, the identification of the RDF subject and the RDF predicate or relation are two key steps to effectively process the query. Once these two are identified, the RDF objects from the knowledge graph can be easily fetched. For subject identification, NLP techniques such as the Named Entity Recognition (NER) plays a crucial role. And, for the relatively complex task of predicate selection, query needs to be processed both syntactically and semantically to identify the relationship being queried. NLP pipelines with part of speech tagging and dependency parsing combined with the semantic similarity between query phrases play a crucial role in predicate selection [3]. Semantic similarity can also be computed using concepts hierarchy of the ontology [4, 5].

Word embeddings are large vectors representing semantics of words in a contextualized manner [6]. That is the words embeddings for semantically similar words will be somewhat identical. Applying pairwise similarity metrics such as the Cosine Similarity will be closer to 1.0 for

similar words. Word embedding can be derived from pre-trained models such as BERT, which belongs to a class of Transformer models and phrase embeddings generated from it are used across numerous fields. While GPT-based models such as ChatGPT belong to a class of Large Language Models (LLMs) and are sometimes referred as generative AI models. Also, these models may be further trained on a specific dataset pertaining to a specific field of study, this process is often referred as fine tuning of the model and it can be highly effective in that field of discourse [7].

In this paper firstly, a novel predicate selection approach on RDF knowledge bases has been proposed and implemented. Semantic similarity between query and predicates is computed using BERT-based phrase embeddings and custom logic for key part-of-speech tokens. Secondly, a dataset of semantically matching queries is generated using ChatGPT for benchmarking the predicate selection accuracy. Thirdly, the implemented prototype is benchmarked on YAGO Date Facts with combinations of key part of speech tokens and stop words, demonstrating the effectiveness of our approach.

Section 2 presents the literature survey in this direction. Section 3 presents our proposed RDF predicate selection approach along with procedural steps. A key criterion for prominent part of speech tokens is discussed in sub-section 3.1. Section 4 details the implementation setup and the details of the novel dataset that is used. Results are discussed in Section 5 along with the Benchmarking approach. It is followed by the conclusions and future scope.

### 1.1 *Contributions of the paper*

- A novel approach for predicate selection on RDF knowledge bases has been proposed and implemented utilizing phrase embeddings and part of speech based semantic similarity.
- Combination of various part of speech token and stop words is benchmarked over a novel dataset of semantically matching queries to find the set of important part of speech tokens.
- Benchmarking results also highlight the importance of retaining verb stop words for predicate selection processing.

## 2. Literature Survey

This section surveys the recent literature on Part of Speech (POS) based semantic similarity, word embeddings, question answering (QA)

over knowledge base (KB) [2, 8]. Various state-of-the-art word embedding algorithms considering the semantic features are surveyed in [6].

To increase the accuracy of text sentiment classification, Yin et al. have proposed contextual POS chunks to construct the sentiment lexicons [9]. They considered POS of the words to compute sentiment and the polarity. Their approach was evaluated on a domain-specific text corpus for sentiment classification resulting in an 82% accuracy.

For computing semantic similarity for the task of a related field of video retrieval, Wray et al. have utilized POS tags and the SYN sets present in the parsed video captions as similarity metrics [10]. They have highlighted that matching tokens without considering the POS may result in computation of incorrect semantic similarities. Also, the POS and SYN sets are particularly useful for actions which are verb in nature thus, increasing the importance of verb tokens.

Vulić et al. have proposed Multi-SimLex, for large-scale lexical benchmarking covering 12 languages [11]. It allows analysis over POS classes, relation types, similarity metrics etc. They have considered four important POS classes viz. noun, verb, adjective and adverb. Around 1900 semantic similarity-based concept pairs are annotated for each language. They have further evaluated the results against famous word embedding approaches such as BERT and FastText. Other approaches consider only the nouns, the verbs and the leading adjectives as important POS [12]. Word classification considering syntactic, semantic and morphologic features is an ongoing aspect in linguistics research and is vital for similarity computations [13].

Xu et al. have proposed AutoQA, a semantic text parsing toolkit for QA over databases [14]. A key feature is to find alternative phrase expressions over different POS thus achieving paraphrasing for input sentences. Considering the POS of the original phrase, it generates questions asking for objects using paraphrases from a neural model. The paraphrases are parsed and verified with a list of grammar rules and finally converted into annotations.

Reilly et al. have proposed a bigram semantic model for semantic processing of continuous natural language [15]. This could be highly useful in deriving phrase-level semantics and identifying relation patterns. For multi-word similarity, Yalcin et al. have proposed an n-gram plagiarism detector using Word2vec similarity and POS tags of the input and target sentences [16].

## 3. Proposed Approach & Key Aspects

This section presents the proposed RDF predicate selection approach using phrase embeddings and part of speech tokens based semantic similarity. Figure 1 depicts the key modules, aspects and process flow. The user provided natural language query Q is processed at token level to extract out key part of speech tokens which are later transformed into phrase embeddings. The embeddings generated from key query tokens are semantically compared against the embeddings generated from RDF KB predicates. RDF predicate embeddings are also generated using similar set of part of speech tokens. Finally, the RDF predicate is selected based on maximum semantic similarity score subject to a threshold value for minimum similarity score which denotes significant matching.

The part of speech combination (i.e., Nouns, Verbs, Adjectives, Adverbs excluding the subject noun and non-verb stop words) has been finalized after benchmarking the combinations over KB using a novel dataset of query variations as detailed in Section 5.

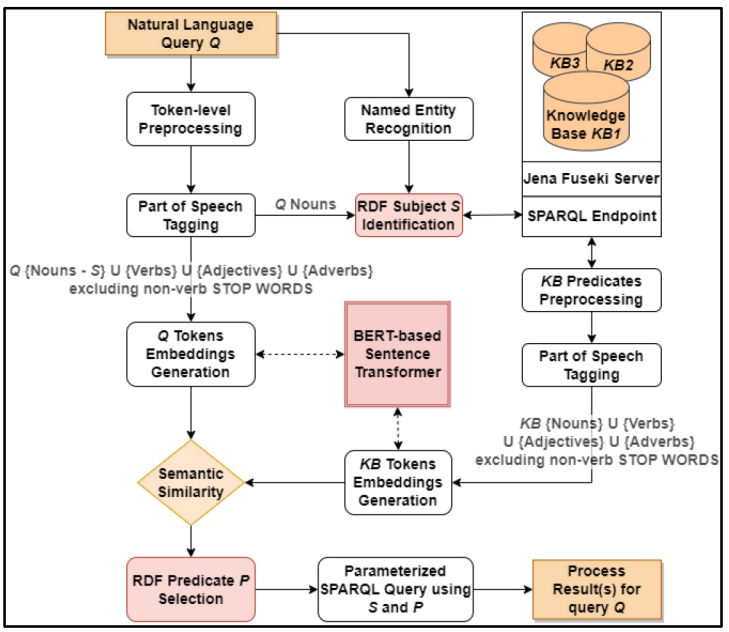


**Figure 1**

**Proposed QA system over KB: RDF predicate selection using BERT phrase embeddings and semantic similarity of key part of speech tokens**

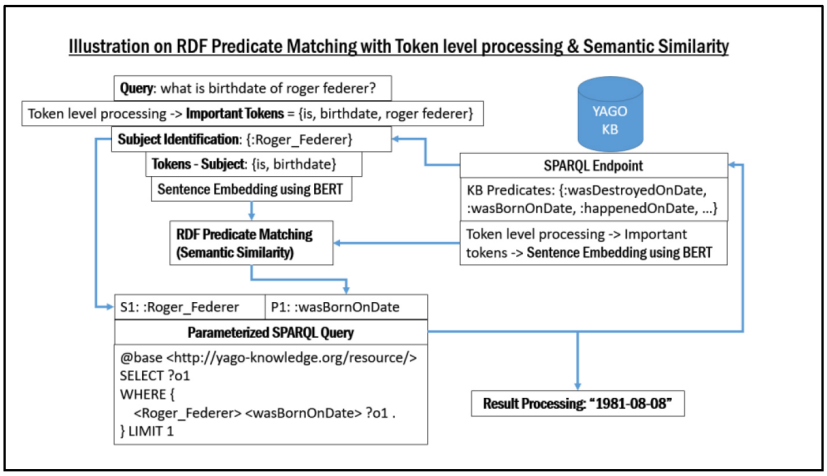


**Figure 2**

**Illustration on complete workflow for the implemented system**

Finally, the identified RDF subject (*S*) and the selected RDF predicate (*P*) are used to generate the SPARQL query which is executed on the SPARQL endpoint of the KB. The fetched RDF object is formatted and returned as the answer for the query. Figure 2 depicts the end-to-end workflow summarized above with a sample query and its response.

### 3.1 *Criteria for Prominent Parts of Speech Tokens*

Literature suggests that the nouns, the verbs and the leading adjectives are the most prominent keywords for understanding the participating entities and their relationships [12]. In general, it can be noticed that entities or concepts belong to the noun part of speech. Relationships belong to the verb part of speech in general but, some other parts of speech such as adverbs may also be used along with verbs. It is also noticed that many sentences use pronouns to replace the subject entity. Entity co-referencing techniques such as neural co-reference resolve the pronouns to their respective subject entities [17, 18]. So, in the analysis, different combinations of key parts of speech are benchmarked while performing the token-level processing.

Another key aspect is the decision to retain or eliminate the stop words from the query. Stop words are often regarded as less important tokens of a natural language query. But, for specific cases retention of these stop words become crucial especially if the query uses more stop words. In the analysis, different cases with stop word retention and

elimination are addressed. Set of standard English language stop words is used. Benchmarking results demonstrate the importance of certain verb stop words (such as “is”, “was”, and “did” etc.) for predicate selection. Table 1 shows the list of stop words belonging to verb part of speech. The benchmarking results are presented in Section 5. Following combinations for parts of speech and stop words are addressed:

- NVAdjAdvPrn: Noun(s) + Verb(s) + Adjective(s) + Adverb(s) + Pronoun(s)
- NVAdjPrn: Noun(s) + Verb(s) + Adjective(s) + Pronoun(s)
- NVAdj: Noun(s) + Verb(s) + Adjective(s)
- NV: Noun(s) + Verb(s)
- NVAdjAdvPrn – All Stop Words
- NVAdjAdvPrn – Non-verb Stop Words

**Table 1**

**Stop Words belonging to Verb Part of Speech**

| Stop Words Tagged as Verbs | | | | |
|---|---|---|---|---|
| am | being | had | is | ‘ve |
| are | did | has | ‘s | was |
| be | do | have | ‘ll | were |
| been | doing | having | ‘m | ‘re |

### 3.2 *Limitations*

Following are the limitations of the proposed approach:

- If RDF subject is unavailable in the KB, predicate selection will be skipped. This situation may arise due to failed NER step or due to the knowledge incompleteness. Such queries can’t be answered with the current implementation.
- Predicate selection considers the scenario for knowledge incompleteness i.e., if the semantic similarity between the embeddings for the query and KB predicates is less than a minimum threshold value of relevancy. It will help in responding precise answers only, skipping the ones having less confidence.

## 4. Implementation Setup

This section presents the implementation details of the predicate selection approach along with the details of queries dataset and the KB used for benchmarking. For part of speech tagging, "spacy" python library along with a spacy model "en_core_web_lg" is used. And for generating the phrase embeddings BERT model "bert-base-cased" was utilized. Cosine similarity is computed using the "sklearn" python library.

Figure 3 depicts a sample code snapshot for illustration on BERT embedding and the computation of semantic similarity using cosine similarity metric. Four sample phrases are used which are semantically matching. Phrase 3 and 4 are subset of phrase 1 and 2 by filtering out the preposition "in" and the verb stop word "has" respectively. Firstly, their embedding vectors are generated using the BERT model. Secondly, the pairwise cosine similarity is computed for embedding vector pairs. In this particular example, it can be noticed that upon filtering the preposition "in" from phase 1, its similarity with phrase 2 has increased from 0.897 to 0.920. Whereas while filtering out the verb stop word "has" from the phrase 2, decreases its similarity with the phrase 1. Also, it can be noticed that in the second phrase "has" is the only verb relation. Thus, for the predicate selection processing, verb stop word may become very useful, if it is the only verb present in the query.

```python
from sentence_transformers import SentenceTransformer
from sklearn.metrics.pairwise import cosine_similarity

bert_model = SentenceTransformer('bert-base-cased')

# Predicates ":locatedIn" and ":hasLocation"
sent_1 = "located in"       # VERB + ADPOSITION
sent_2 = "has location"     # VERB + NOUN
sent_3 = "located"          # VERB
sent_4 = "location"         # NOUN

# Sentence Embeddings
s1_embed = bert_model.encode([sent_1])
s2_embed = bert_model.encode([sent_2])
s3_embed = bert_model.encode([sent_3])
s4_embed = bert_model.encode([sent_4])

print("Similarity {'located', 'in'}::{'has', 'location'}", cosine_similarity(s1_embed, s2_embed))
print("Similarity {'located'}::{'has', 'location'}", cosine_similarity(s3_embed, s2_embed))
print("Similarity {'located', 'in'}::{'location'}", cosine_similarity(s1_embed, s4_embed))
```

```
PROBLEMS  OUTPUT  DEBUG CONSOLE  TERMINAL  PORTS

Similarity {'located', 'in'}::{'has', 'location'} [[0.897038]]
Similarity {'located'}::{'has', 'location'} [[0.9208419]]
Similarity {'located', 'in'}::{'location'} [[0.87804437]]
```

**Figure 3**

**Illustration on semantic similarity computation using BERT embeddings and cosine similarity**

**Table 2**
**RDF Predicates in YAGO Date Facts KB**

| Property | Description |
|---|---|
| startedOnDate | Start date of an event |
| happenedOnDate | Date on which some famous event happened |
| wasBornOnDate | Birth date of popular people |
| wasDestroyedOnDate | Date on which some popular structure or record was destroyed/discontinued |
| hasGloss | In linguistics, a gloss is a brief definition of a word, phrase, or sentence. |

For conducting the benchmarking, a dataset of 110 semantically matching queries was utilized which was earlier generated using ChatGPT prompting [19]. Next, the consolidated dataset was checked for grammatical errors and was further processed for generating various sets based on part of speech combinations.

Benchmarking is conducted on "YAGO Date Facts" knowledge base for querying [20]. This is itself a large dataset containing about 1.8 million triples about date events. It contains facts of events with the date with either of the following properties as listed in Table 2.

These predicates are pre-processed with part of speech token filtering logic finally the phrase embeddings are generated for each processed predicate.

## 5. Results & Discussion

This subsection details the benchmarking approach, results for predicate selection and accuracy of different part of speech combinations.

*Benchmarking Approach:*

/* Preprocessing and Loading models in memory */

**1.** For NLP processing, such as Part of Speech Tagging load spacy model "en_core_web_lg". Load BERT model.

/* Prepare collection of Stop Words */

**2.** Prepare set for stop words: "All Stop Words", "Verb Stop Words", and "Non-verb Stop Words".

/* Process queries dataset */

**3.** For all the queries:

**3.1.** Subject in the query is identified (using NER) and is excluded from the tokens tagged as nouns for further processing.

**3.2.** Prepare sets of part of speech combinations to be benchmarked viz. "NVAdjAdvPrn", "NVAdjPrn", "NVAdj", "NV", "NVAdjAdvPrn – All Stop Words", and "NVAdjAdvPrn – Non-verb Stop Words".

**3.3.** Generate phrase embedding for the respective sets excluding the tokens in respective stop word set collection.

/* Process KB Predicates */

**4.** Extract all RDF predicates from the KB and generate their embedding considering only key parts of speech tokens.

/* Compute semantic similarity and comparison metrics */

**5.** If the RDF subject is identified* (Step **3.1** above):

**5.1.** Compute semantic similarity between embeddings of the KB predicates and embeddings for the query tokens embedding for each set.

**5.2.** Select the RDF predicate with maximum similarity score.

**5.3.** Increment the counter for selected RDF predicate for the respective collection.

/* Evaluate accuracy for correct RDF predicate selection */

**6.** If the selected predicate matches with the known correct RDF predicate:

**6.1.** Frame the SPARQL query with RDF subject and RDF predicate parameters.

**6.2.** Execute the SPARQL query on the endpoint to fetch the response RDF object.

**6.3.** Format and return the RDF object as the query response from the KB.

A master sheet was prepared during the benchmarking, covering following details for analysis: Query variations; Semantic Similarity for predicates (across all combinations of parts of speech); Matched RDF predicate; and the Filtered Adjectives, Adverbs, Pronouns at that stage (depending on the combination).

Results for RDF predicate selection for the 110 semantically matching queries are presented in Figure 4 with counts for predicate matched across all the six part of speech combinations being compared.

For queries related to the birthdate of Roger Federer, the known correct predicate URI is <http://yago-knowledge.org/resource/wasBornOnDate>. Thus, the accuracy of correct RDF predicate selection across the six parts of speech combinations is presented in Figure 5. As per the benchmarking results, category "NVAdjAdvPrn – "Non-verb Stop Words"" has the highest predicate selection accuracy at around 72.73%.

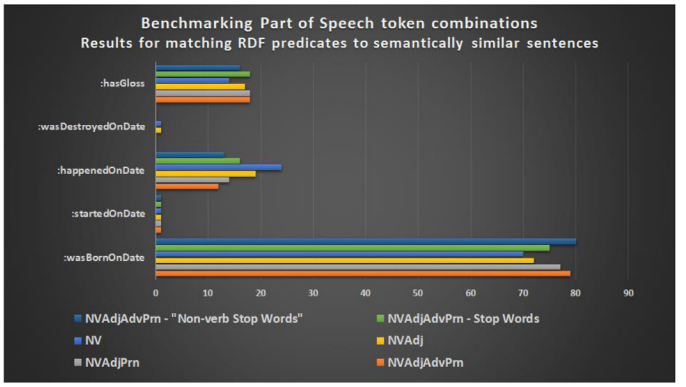


**Figure 4**

**Benchmarking of matching RDF predicates based on semantic similarity scores across Part of Speech combinations. Notation: Noun (N), Verb (V), Adjective (Adj), Adverb (Adv), Pronoun (Prn).**

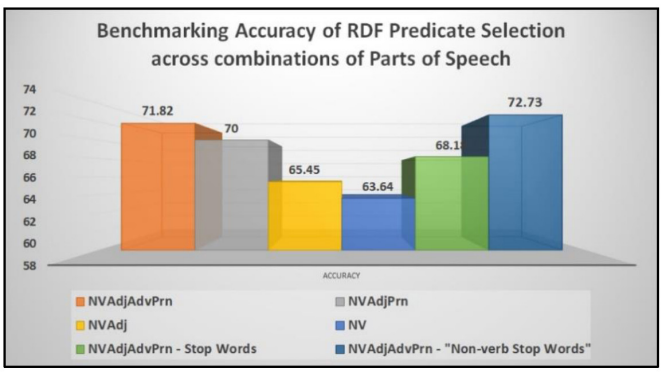


**Figure 5**

**Accuracy for correct RDF predicate selection for the benchmarked part of speech combinations**

Whereas the category for "NV" scored the least at 63.64%. The set containing nouns, verbs and the adjective scored 65.45% accuracy.

Based on the analysis and benchmarking results, key parts of speech tokens which should be given importance while matching and extracting predicate relation from KB are Noun(s), Verb(s), Adjective(s), Adverb(s), and Pronoun(s). Pronouns should be resolved and replaced by respective entity for further query processing. Pipeline with these parts of speech often yield better semantic similarity for predicate selection. Another key aspect from the benchmarking results demonstrates the importance of

specific stop words, which belong to verb part of speech. Specific examples of such stop words are: “is”, “was”, “did” etc. Thus, a custom rule could be applied at the stop words removal step of NLP pipeline to exclude verb stop words from being filtered out. Its benefits may become more obvious for queries containing more stop words or in cases where stop word is the only verb present in the query.

## 6. Conclusion & Future Scope

In this paper, a predicate selection approach is proposed and implemented utilizing the phrase embeddings for the key part of speech tokens in the query. Retaining the best combination of tokens increase the accuracy of correct predicate selection. Benchmarking is conducted over YAGO Date Facts KB using a novel dataset of semantically similar sentences comparing the set of key part of speech token combinations along with stop words.

Key takeaways from our analysis are:

- For predicate selection task, important parts of speech tokens for query processing are: Noun(s), Verb(s), Adjective(s), Adverb(s), and Pronoun(s) excluding the non-verb Stop Words. Pronouns should be resolved and replaced by respective entity for further query processing.
- Benchmarking results highlight the importance of specific stop words, which belong to verb part of speech, during query processing. Specific examples of such stop words are: “is”, “was”, “did” etc. These are listed in Table 1 for reference.
- Stop word removal is often considered as a pre-processing step in NLP pipelines. Specifically, for relation extraction from a query, stop words belonging to verb part of speech should be retained. Also, the Verb Stop Word should not be filtered in case it is the only verb present in the query.

As part of the future work, other semantic similarity metrics may be integrated into the benchmarking approach. Further, complex queries containing conjunctions and prepositions may be analysed for predicate selection task.

**Acknowledgements**

This publication is an outcome of research supported by the Visvesvaraya Ph.D. Scheme, MeitY, Govt. of India. VISPHD-MEITY-2800.

**References**

[1] T. Ahmad, M. Ahamad, S. U. Ahmed and N. Ahmad, "Short question-answers assessment using lexical and semantic similarity-based features", *Journal of Discrete Mathematical Sciences and Cryptography,* vol. 25(7),, pp. 2057–2067 (2022).

[2] A. Pereira, A. Trifan, R. Lopes and J. Oliveira, "Systematic review of question answering over knowledge bases", *IET Software*, vol. 16(1), pp. 1-13 (2022).

[3] Z. H. Amur, Y. Kwang Hooi, H. Bhanbhro, K. Dahri, and G. M. Soomro, "Short-Text Semantic Similarity (STSS): Techniques, Challenges and Future Perspectives", *Applied Sciences*, vol. 13(6), pp. 3911 (2023).

[4] P. Rathee, and S. K. Malik, "IWD towards Semantic similarity measure in ontology", *Journal of Information and Optimization Sciences, vol.* 41(7), pp. 1561–1577 (2020).

[5] P. Rathee, and S. K. Malik, "Ontology concept semantic similarity matching based on Ant Colony Optimization algorithm", *Journal of Information and Optimization Sciences,* vol. 42(8), pp. 1987–2000 (2021).

[6] P. J. Worth, "Word embeddings and semantic spaces in natural language processing", *International Journal of Intelligence Science, vol.* 13(1), pp. 1-21 (2023).

[7] M. Bilal, and A. A. Almazroi, A. A., "Effectiveness of fine-tuned BERT model in classification of helpful and unhelpful online customer reviews", *Electronic Commerce Research,* vol. 23(4), pp. 2737-2757 (2023).

[8] C. Antoniou and N. Bassiliades, "A survey on semantic question answering systems", *The Knowledge Engineering Review*, vol. 37, pp. 1-42 (2022).

[9] F. Yin, Y. Wang, J. Liu, and L. Lin, "The construction of sentiment lexicon based on context-dependent part-of-speech chunks for semantic disambiguation", *IEEE Access*, vol. 8, pp. 63359-63367 (2020).

[10] M. Wray, H. Doughty, and D. Damen, "On semantic similarity in video retrieval", In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp. 3650-3660 (2021).

[11] I. Vulić, S. Baker, E. M. Ponti, U. Petti, I. Leviant, K. Wing, O. Majewska, E. Bar, M. Malone, T. Poibeau, and R. Reichart, "Multi-simlex: A large-scale evaluation of multilingual and crosslingual lexical semantic similarity", *Computational Linguistics*, vol. 46(4), pp. 847-897 (2020).

[12] F. S. Lievers, M. Bolognesi, and B. Winter, "The linguistic dimensions of concrete and abstract concepts: lexical category, morphological structure, countability, and etymology", *Cognitive Linguistics*, vol. 32(4), pp. 641-670 (2021).

[13] T. Rustamov, X. M. Jumanazarov, U. Almatova, Z. X. Mamaziyayev, and Z. A. Alibekova, "Classification symbols of words", *Asian Journal of Research in Social Sciences and Humanities,* vol. 12(2), pp. 213-219 (2022).

[14] S. Xu, S. Semnani, G. Campagna, and M. Lam, "AutoQA: From databases to QA semantic parsers with only synthetic training data", In Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP) (2020).

[15] J. Reilly, A. M. Finley, C. P. Litovsky, and Y. N. Kenett, "Bigram semantic distance as an index of continuous semantic flow in natural language: Theory, tools, and applications", *Journal of Experimental Psychology: General*, vol. 152(9), pp. 2578 (2023).

[16] K. Yalcin, I. Cicekli, and G. Ercan, "An external plagiarism detection system based on part-of-speech (POS) tag n-grams and word embedding", *Expert Systems with Applications*, vol. 197, pp. 116677 (2022).

[17] X. Zhao, H. Zhang, and Y. Song, "PCR4ALL: A Comprehensive Evaluation Benchmark for Pronoun Coreference Resolution in English", In Proceedings of the Thirteenth Language Resources and Evaluation Conference, pp. 5963-5973 (2022).

[18] R. Liu, R. Mao, A. T. Luu, and E. Cambria, "A brief survey on recent advances in coreference resolution", *Artificial Intelligence Review*, vol. 56(12), pp. 14439-14481 (2023).

[19] OpenAI. (2023). ChatGPT (Feb 13 version) [Large language model]. https://chat.openai.com/chat.

[20] F. Mahdisoltani, J. Biega, and F. M. Suchanek, "Yago3: A knowledge base from multilingual wikipedias", In CIDR (2013).

*Received April, 2024*