Available online at sjuoz.uoz.edu.krd

Science Journal of University of Zakho
Vol. 11, No. 3, pp. 396– 402, July-September, 2023

journals.uoz.edu.krd

p-ISSN: 2663-628X
e-ISSN: 2663-6298

EVALUATING OF EFFICACY SEMANTIC SIMILARITY METHODS FOR COMPARISON OF ACADEMIC THESIS AND DISSERTATION TEXTS

Ramadan Thkri Hassan [a] , Nawzat Sadiq Ahmed [b]

[a] Information Technology, Technical College of Informatics, Duhok Polytechnic University, Kurdistan Region - Iraq
ramadan.hassan@dpu.edu.krd
b Information Technology Management, Technical College of Administration, Duhok Polytechnic University, Kurdistan Region,- Iraq
nawzat.ahmed@dpu.edu.krd

Received: 13 Feb., 2023 / Accepted: 5 Apr., 2023 / Published: 3 July 2023 *https://doi.org/10.25271/sjuoz.2023.11.3.1120*

ABSTRACT:

Detecting semantic similarity between documents is vital in natural language processing applications. One widely used method for measuring the semantic similarity of text documents is embedding, which involves converting texts into numerical vectors using various NLP methods. This paper presents a comparative analysis of four embedding methods for detecting semantic similarity in theses and dissertations , namely Term Frequency–Inverse Document Frequency, Document to Vector, Sentence Bidirectional Encoder Representations from Transformers, and Bidirectional Encoder Representations from Transformers with cosine similarity. The study used two datasets consisting of 27 documents from Duhok Polytechnic University and 100 documents from ProQuest.com. The texts from these documents were pre-processed to make them suitable for semantic similarity analysis. The evaluation of the methods was based on several metrics, including accuracy, precision, Recall, F1 score, and processing time. The results showed that the traditional method, TF-IDF, outperformed modern methods in embedding and detecting actual semantic similarity between documents, with processing time not exceeding a few seconds.

KEYWORDS: TF-IDF, BERT, SBERT, Doc2Vec, Semantic Similarity, Cosine Similarity, NLP

1. INTRODUCTION

In recent years, natural language processing (NLP) has gained increasing attention as a field of computer science that deals with how computers can analyze, understand, and generate human language. One of the critical tasks in NLP is measuring the semantic similarity between different texts, such as documents or terms. This task involves assessing the degree of similarity in meaning between texts rather than their lexical similarity. Semantic similarity is an important technique in various NLP applications, including semantic search, document classification, sentiment analysis, information retrieval, semantic plagiarism detection, and question answering (Chawla et al., 2022).

One common approach for measuring the similarity of text documents is embedding them into a vector space, where the vectors can be compared using similarity measures, such as cosine similarity. Embedding methods convert text documents into vectors that capture their semantic contents. The choice of the embedding method is crucial in ensuring accurate semantic similarity measurement (Park et al., 2020).

Long text documents, such as theses and dissertations, pose a unique challenge when measuring semantic similarity. This study aims to identify the most effective methods for measuring the semantic similarity of long text documents. To achieve this objective, the performance and efficeny of traditional and modern approaches are compared to measure semantic similarity, including TF-IDF, Doc2Vec, BERT, and SBERT.

These methods were evaluated based on several metrics, including accuracy, precision, Recall, F1 score, and the time required for implementation. By comparing the results of these methods, we aim to determine which method is the most effective for measuring the semantic similarity of long text documents. The findings of this study will provide valuable insights into the optimal approaches for measuring semantic similarity in the context of long text documents, with potential implications for a wide range of NLP applications.

This study is divided into seven sections. Section 1 provides an introduction to the research topic and the purpose of the study. Section 2 discusses related works on semantic similarity to provide context for the study. Section 3 presents an overview of natural language processing (NLP), which is the foundation of the proposed methodology. Section 4 focuses on semantic similarity, including its definition and significance in NLP.

Section 5 provides a detailed explanation of the methodology used in this study, which includes four different methods for finding semantic similarity: TF-IDF, Doc2Vec, BERT, and SBERT. Section 6 presents the results of the present study , including comparing each method's accuracy, precision, Recall, and F1 score. In addition, this section includes a discussion of the results, including their implications and limitations. Finally, in Section 7, the conclusion of the study is presented.

2. RELATED WORKS

Many researchers have evaluated different semantic similarity approaches for short and long text documents using text embedding produced by both traditional and modern approaches. The following works were focused on different text documents in terms of short text.

Malmberg et al.(Malmberg, 2021) conducted a study to evaluate the performance of semantic similarity searches using sentence embedding generated by both the traditional and modern methods. Experiments were conducted to compare the various techniques for creating sentence embedding. Since specific datasets for these experiments were not-available, commonly used datasets were adopted. The results showed that the TF-IDF algorithm outperformed the neural network-based approaches (Sentence-BERT, BERT) in nearly all experiments.

Gumel, Nigeria et al. (Dept. of Computer Science, Jigawa State Colledge of Education, Gumel, Nigeria et al., 2022) conducted a study on using machine learning techniques for text vectorization or word embedding, which is a crucial step in natural language processing tasks as most of machine learning algorithms require numerical input. The process of text vectorization involves mapping words or documents in a corpus to numerical vectors. There are various approaches to document/text representation in the literature. However, this study focuses on three commonly used methods , namely 1-Bag of Words, 2-TF-IDF, 3-(Word2Vec, and Doc2Vec) . The study also aims to identify the reasons behind their use and offer recommendations to researchers . The review of this study revealed that TF-IDF feature vector representations generally outperformed the other two vectorization methods, Word2Vec and Doc2Vec.

Mandal et al. (Mandal et al., 2021) carried out a study to evaluate the performance of 56 different approaches for computing textual similarity, including both traditional and advanced context-aware methods, for computing textual similarity in court case statements from the Supreme Court of India. The study found that traditional unsupervised approaches, such as LDA and TF-IDF, which rely on bag-of-words representation, outperformed more advanced ones, such as Law2Vec which is unsupervised and BERT, which is supervised approaches in computing document-level similarity.

Singh and Shashi (Singh & Shashi, 2019) proposed a framework for grouping news articles related to popular topics on social media. The aim was to sort the articles into similar groups based on their meanings. The study used three techniques for capturing the semantic similarity of articles: TF-IDF, Word2Vec, and Doc2Vec. The k-means algorithm was applied to cluster the vectorized articles. The results from experiments with the DUC 2004 benchmark dataset indicated that the TF-IDF vectorization method was the most effective in creating pure clusters for static datasets.

Zhu et al. (Zhu et al., 2021) designed a scholarly recommendation system that suggests research papers relevant to public datasets from Gene Expression Omnibus (GEO) from PubMed. They evaluated various techniques for representing textual data. They found that term-frequency-based methods such as BM25 and TF-IDF performed better compared to other techniques, including popular NLP embedding models like Doc2Vec, ELMo, and BERT.

Shahmirzadi et al. (Shahmirzadi et al., 2018) evaluated the effectiveness of text vectorization methods for determining the similarity between patents. They compared a basic TF-IDF approach to more advanced methods, such as extensions of TF-IDF, Latent Semantic Indexing (LSI) topic modeling, and Doc2Vec neural modeling. They tested the models on short and long texts for easier and more difficult similarity detection tasks. They found that for their particular application, the simple TF-IDF method was a suitable choice, considering its performance and cost. The use of more complex embedding methods like LSI and Doc2Vec would only be justified if the text was very concise and the similarity detection task was relatively simple.

Vrbanec and Meštrović (Vrbanec & Meštrović, 2020) conducted a comprehensive evaluation of corpus-based models for detecting semantic similarity in texts, which is critical for paraphrase detection. They assessed various pre-processing techniques, such as hyper-parameters, distance measures, and thresholds for semantic similarity and paraphrase detection by testing different text representation models. The performance of six (6) deep-learning methods, namely USE, Glove, Word2Vec, ELMO, Fast-Text and Doc2Vec and two of the traditional methods (LSI and TF-IDF) was compared using three public corpora (Webis Crowd Paraphrase Corpus 2011, Clough and Stevenson, and Microsoft Research Paraphrase Corpus). The results indicated that the traditional TF-IDF model exhibited superior performance in comparison to the other models, including Word2Vec, Doc2Vec, Glove, Fast-Text, and ELMO, in terms of accuracy, precision, Recall, and F1 measure.

Pranjic and Podpecˇan (Pranjic & Podpecˇan, 2020) compared the effectiveness of several link recommendation methods on a news archive from a popular Croatian website, 24sata. The results showed that the TF-IDF weighting applied to the bag-of-words document representation provided better matches with manually selected links by journalists than more advanced methods, like multilingual contextual embedding's BERT and XLM-R, Doc2Vec, and latent semantic indexing.

Based on the literature mentioned above, it is shown that most researchers were focusing on the short text. Also, the TF-IDF method of determining the semantic similarity of short texts outperforms other modern methods. This study compares the performance and efficacy of the traditional TF-IDF method and modern methods in predicting semantic similarity in terms of accuracy, precision, Recall, F1 score, and time using long text documents of academic theses and dissertations of graduate students.

3. NATURAL LANGUAGE PROCESSING (NLP)

NLP is a subfield of artificial intelligence that focuses on teaching machines to understand and generate natural languages. NLP involves techniques for analysing, processing, and understanding human language using statistical and machine-learning methods. NLP is an interdisciplinary field that draws on linguistics, computer science, and cognitive psychology, among other areas. On-going research in NLP has focused on developing more advanced algorithms in order to better understand the complexities of human language(Ofer et al., 2021) (Jones, 1999).

NLP techniques can be used for a variety of applications (Goldberg, 2017), including

- Semantic analysis: a broad term that encompasses various NLP techniques for analysing the meaning of natural language texts
- Sentiment analysis: determining the sentiment (positive, negative, neutral) of a piece of text
- Named entity recognition: identifying and categorizing named entities (such as people, organizations, and locations) in a piece of text
- Text classification: assigning categories or labels to a piece of text
- Machine translation: automatically translating text from one language to another
- Question answering: answering questions posed in natural languages

4. SEMANTIC SIMILARITY

Semantic similarity is a metric that assesses the similarity of meaning between different text documents or terms. It is based on the semantic content of the documents or terms rather than their lexical similarities. It is a numerical representation of the similarity between two items, such as concepts, sentences, or documents. It is calculated by comparing the Information supporting their meaning or describing their nature. Semantic similarity is an important technique in NLP and is utilized in various applications, including semantic search, document classification, sentiment analysis, information retrieval, semantic plagiarism detection, and question answering. Accuracy is a key concern in the process of semantic similarity, and various techniques have been developed to measure the similarity between text documents(P. & Shaji, 2019).

Semantic similarity can be calculated using a variety of techniques(Mikolov et al., 2013), including

- Word embeddings: representing words as vectors in a high-dimensional space based on their contexts in a corpus of text
- Latent semantic analysis: analysing the statistical patterns of co-occurrence between words in a corpus of text to identify underlying concepts or topics
- Knowledge-based methods: using external knowledge sources, such as ontologies or semantic networks to identify semantic relationships between words.

Semantic similarity has a wide range of applications(Mikolov et al., 2013), including

- Information retrieval: retrieving documents or passages of text those are semantically similar to a user's query
- Plagiarism detection: identifying text that is similar in meaning to the previously published texts
- Textual entailment: determining whether one piece of text logically entails another piece of text

5. METHODOLOGY

In this study, the authors conducted an experiment applying four methods to determine which method is the most effective in predicting the semantic similarity between long text documents of theses and dissertations of graduate students. These methods were TF-IDF, the oldest traditional method for finding semantic similarity between documents, and Doc2Vec, BERT, and SBERT, which are modern ways to find semantic similarity between documents. Then, the authors compared the results of these four methods in terms of efficiency and time. The following sub-sections show the steps to collect data and conduct the experimental test.

5.1 Data collection

To perform the experiment, two collections of theses and dissertations were gathered.

- The Duhok Polytechnic University collection (https://www.dpu.edu.krd/page/en/5808/) includes 27 original English theses and dissertations submitted to the University.
- ProQuest collection from proquest.com includes 100 original English theses and dissertations. ProQuest is a company that provides access to various academic resources, including databases, eBooks, and periodicals.

For each collection, the four methods were applied to compute the semantic similarity between the theses and dissertations, which involves comparing each thesis or dissertation to the other documents using the four methods. As shown in Figure 1, each thesis or dissertation can be treated as a suspected document and compared to all other theses and dissertations in the collection as source documents. The results of these comparisons are used to evaluate each method's accuracy, precision, Recall, and F1 score in finding the semantic similarity in the collection.

5.2 Text extraction

In order to compare the semantic similarity between the source and suspected documents, it is necessary to extract the relevant texts from the documents (see Figure 1). For thesis and dissertations, this involves cutting the text from the beginning of the first chapter to the end of the last chapter, as these are the most important sections for comparison. Pages such as the front page, contents page, dedication page, and list of terms and drawings are not included in the text extraction process.

5.3 Text normalization

To improve the efficiency of semantic similarity comparisons, it is necessary to normalize the text by deleting certain characters and converting the text to lowercase (see Figure 1). This includes removing characters, such as periods, commas, semicolons, parentheses, special characters, non-English characters, white spaces, quotation marks, and numbers(Davoodifard, 2022).

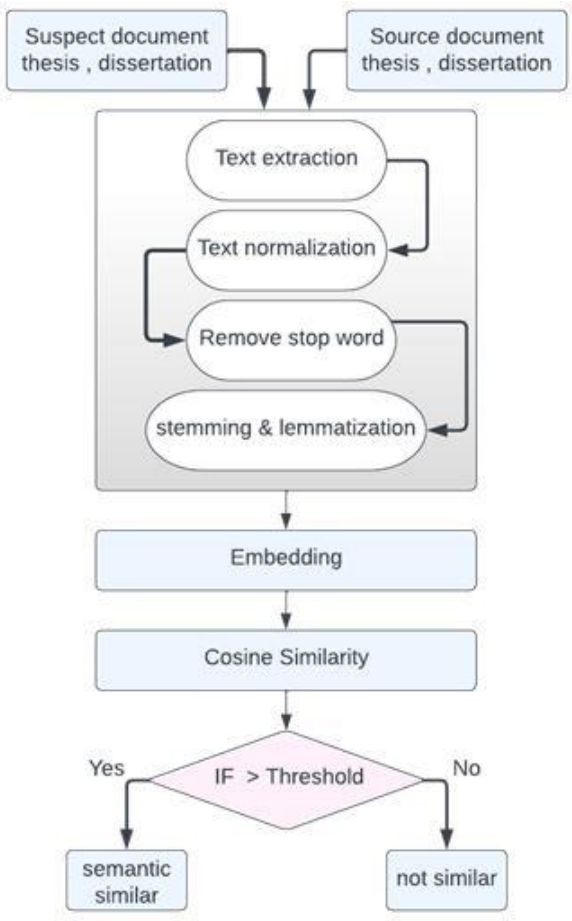

Figure 1: The Model of Experimental Test Steps

5.4 Remove stop word

As shown in Figure 1, removing stop words is a prevalent preliminary processing technique in Natural Language Processing (NLP) applications. The process involves eliminating words ubiquitous in all the documents present in the corpus, such as definite and indefinite articles and pronouns. These words are deemed to lack discriminatory power and hold no importance in tasks such as information retrieval and classification(Davoodifard, 2022).

5.5 Stemming and lemmatization

Stemming and lemmatization (see Figure 1): In order to avoid different inflectional forms of words and reduce the derivational forms of words with common affinities, it is necessary to perform stemming and lemmatization. This involves identifying words with etymological links and similar meanings, such as democracy, democratic, and democratization, and reducing them to their base form (Resta et al., 2021).

5.6 Embedding

Embedding, as shown in Figure 1, is a process of creating a numeric representation of words, sentences, and texts in order to enable computers to understand the context and meaning of natural language texts. These representations are typically vectors, with texts that are closer in vector space expected to be semantically similar. This study will use four methods to generate the embedding of documents to find semantic similarities between them. Each document (suspect or source) will be converted into a vector using one of the four methods shown below(Sitikhu et al., 2019).

TF-IDF

Term Frequency - Inverse Document Frequency (TF-IDF) is a widely employed method for constructing document vectors. The algorithm does not factor in the arrangement of words, instead adopting a bag-of-words approach that assigns a weight

to each term based on its frequency within a document relative to its inverse frequency across the corpus. This results in high weights being attributed to terms that are rare in the corpus but frequently appear in a particular document, as they are deemed to potentially be more indicative of that document's content than frequently occurring words (Balani & Varol, 2021).

- DOC2VEC method
  Concept Document to Vector (Doc2Vec), first presented by Le and Mikolov in 2014[20], is an unsupervised algorithm, an extension to the word2vec-approach toward documents. It intends to encode documents, consisting of lists of sentences, used to generate representation vectors (embed) of a document regardless of length. Doc2Vec computes a feature vector for every document in the corpus(Le & Mikolov, 2014).
- BERT method
  Bidirectional Encoder Representations from Transformers (BERT) is a network architecture introduced in 2018 by Devlin et al. It features a sequence of Transformer encoders without including a decoder stack. The original Transformer network features 6 encoders with 8 attention heads per layer. In contrast, BERT has two variations: BERTbase, which includes 12 encoder layers and 12 attention heads per layer, producing embeddings of 768 dimensions, and BERTlarge, boasting 24 encoders and 16 attention heads per layer, with embeddings of 1024 dimensions(Devlin et al., 2019).
- SBERT method
  The Sentence-BERT (SBERT) network is a modified version of a pre-trained BERT network that utilizes Siamese and triplet network structures to generate semantically rich sentence embeddings (Reimers & Gurevych, 2019). A Siamese neural network consists of two identical neural networks that share weights. These networks work together, and their final outputs are compared, typically using a distance metric such as cosine distance. Siamese networks are well-suited for similarity problems, as the weight-sharing property of such networks ensures consistent predictions since each network calculates the same function. Unlike BERT, which outputs embedding for each token in a sentence, SBERT outputs a single embedding for the entire sentence. The authors of SBERT contend that the embeddings produced by SBERT are superior to sentence representations that can be derived from a standard BERT network(Chicco, 2021).

5.7 Cosine Similarity

Cosine Similarity is a measure of semantic similarity, which calculates the cosine of the angle between two vectors that are projected in a multi-dimensional space (embeddings), regardless of their size. The value of cosine similarity is bounded by -1 and 1, as demonstrated by Equation 1, the cosine equation(Magara et al., 2018) [2].

$$\text{Cosine similarity} = (A.\ B) / (\|A\|\ \|B\|) \quad (1)$$

Where:

- A and B are the two vectors being compared
- $(A.\ B)$ is the dot product of vectors A and B
- $\|A\|$ and $\|B\|$ represent the magnitude (length) of vectors A and B, respectively.

The result of this equation is a value in the range of -1 to 1, where 1 indicates a completely similar orientation of the two vectors, -1 represents completely dissimilar vectors, and a value of 0 indicates that the vectors are orthogonal (perpendicular) to each other and have no similarity, as shown in Figure 2.

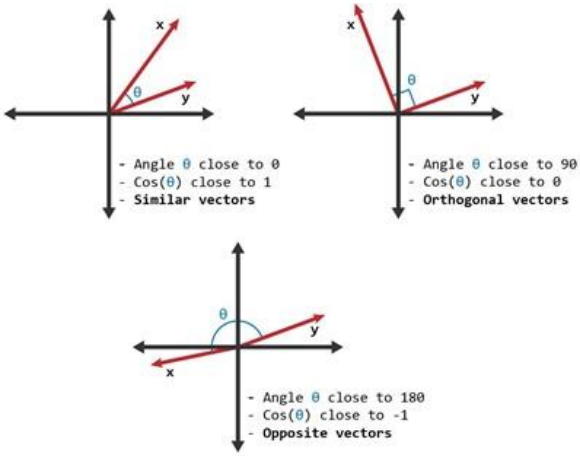

Figure 2: Vector of two documents

5.8 Threshold

The threshold for cosine similarity is the boundary determining whether two documents are considered semantically similar. In this study, the upper limit for the threshold should not exceed 0.5, as a higher similarity between two documents indicates a strong similarity (Brandt, 2019). There is no precise method for determining the threshold. It may vary based on the technique used to compare the semantic similarity between documents and the desired level of similarity. For instance, a threshold value of 0.4 may designate any semantic similarities above this value as a perfect match. The threshold value is usually established through experience and by observing the effectiveness of the comparison method(Brandt, 2019).

6. RESULTS AND DISCUSSION

The experiment for each of the four methods was performed on the Samples taken from the two collections of theses and dissertations (Duhok Polytechnic University and ProQuest.com) to compare their semantic similarities using the four methods. The Duhok Polytechnic University collection consists of 27 original documents, resulting in a total of 729 comparisons of semantic similarity. The ProQuest collection consists of 100 original documents, resulting in a total of 10,000 comparisons of semantic similarity. Each document is compared to all the other documents in the collection using cosine similarity, with a threshold of four.

The following metrics were used to measure the performance of the four methods of semantic similarity:

- Accuracy is the proportion of correct predictions made by the model.
- Precision, in the context of a model, refers to the ratio of true positive predictions to the total number of positive predictions made by the model.
- The Recall, in the context of a model, is defined as the ratio of true positive predictions made by the model to the total number of actual positive cases.
- The F1 score measures a model's accuracy, which combines precision and Recall.
- Time consumption evaluation measures the period of time it takes for a model to make a prediction.

Table 1 shows the result of the experiment applied to the first collection of theses and dissertations at the Duhok polytechnic university.

Table 1: Evaluation metrics and a threshold value of the first collection

Method		threshold	accuracy	Precision	Recall	F1 Score	Time (second)
TF-IDF		4	98	84	99	90	1
Doc2Vec		4	84	59	91	61	226
BERT		4	22	10	90	8	1000
SBERT		4	7	3	92	7	1000

Based on the results indicated in Table 1, it appears that the TF-IDF method performed the best in accuracy, precision, Recall, and F1 score, followed by the Doc2Vec method. The BERT and SBERT methods had significantly lower performance and took much longer to run. It is unclear from the table what the threshold value represented in this context.

Table 2 shows the result of the experiment applied to the second collection of theses and dissertations collected from the proquest.com.

Table 2: Evaluation metrics and a threshold value of the second collection

Method	threshold	accuracy	Precision	Recall	F1 Score	Time (second)
TF-IDF	4	97	66	98	73	2
Doc2Vec	4	96	60	98	66	1200
BERT	4	35	50	53	40	7000
SBERT	4	30	50	51	30	9000(2.5 h)

Based on the results shown in Table 2, The experiment involved applying four different methods (TF-IDF, Doc2Vec, BERT, and SBERT) to the data of 100 original theses and dissertations depending on threshold 4 and evaluating the results using several different metrics (accuracy, precision, Recall, and F1 score). The table also shows the time it took for each method to complete the experiment. The TF-IDF method had the highest accuracy, with a value of 97, while the BERT and SBERT methods had lower accuracy, with values of 35 and 30, respectively.

The TF-IDF method also had the highest precision, Recall, and F1 score values, indicating that it was generally the most effective method for identifying relevant documents and classifying them correctly. The Doc2Vec method had an accuracy of 96, a precision of 60, a recall of 98, and an F1 score of 66. These results indicate that the Doc2Vec method effectively identified relevant documents and classified them correctly. However, it was not as precise as the TF-IDF method in correctly identifying all the relevant documents. The Doc2Vec method also took significantly longer to complete the experiment than the TF-IDF method, with a completion time of 1200 seconds.

The BERT and SBERT methods took longer to complete the experiment than the other methods. This is likely because these methods are based on deep learning models, which are more computationally intensive and require more time to train and process data. Additionally, BERT and SBERT are designed to handle a wide range of natural language tasks. They may be less efficient at processing large amounts of texts compared to other methods that are specifically designed for a particular task.

The observation of the results showed that when the results of the comparison between the documents for each method were presented, the TF-IDF method was able to classify and distinguish documents from the same fields of study and cultivation, where the results were illogical due to the lack of semantic similarity between those documents.

The comparison shown in Table 3 provides a summary of the studies reviewed in this paper which investigated and compared the performance of different text embedding (text vectorization) methods for detecting semantic similarity in various datasets with the result of the proposed work of this paper.

7. CONCLUSION

Based on the experiment results, it was found that the TF-IDF method was the most effective and efficient for measuring semantic similarity between thesis and dissertation documents as long text documents. This method consistently outperformed the other methods (Doc2Vec, BERT, and SBERT) in terms of accuracy, precision, Recall, and F1 score, while being significantly faster than the BERT and SBERT methods.

The Doc2Vec method was also effective at identifying relevant documents and classifying them correctly, but it was not as precise as the TF-IDF method in terms of correctly identifying all the relevant documents. The BERT and SBERT methods took significantly longer to complete the experiment and had lower accuracy, precision, Recall, and F1 score values compared to the other methods.

Therefore, it is recommended to use the TF-IDF method for measuring semantic similarity between thesis and dissertation documents, especially for large datasets, due to its high performance and computational efficiency. However, it is also important to note that the threshold value used in the experiment should be carefully selected based on the specific needs of the research, as it can significantly affect the results of the analysis.

Table 3:Comparison table.

Study	Problem	Dataset	Methods Used	Best Method	Results
Malmberg et al. [3]	Semantic similarity search	SQuAD 1.0 and STS-B	Traditional and modern sentence embedding methods (TF-IDF, Sentence-BERT, BERT)	TF-IDF	TF-IDF outperformed neural network-based methods (BERT, Sentence-BERT) in nearly all experiments.
Gumel, Nigeria et al. [4]	investigate and compare the performance of different text vectorization methods	collection of book reviews and associated metadata from Goodreads.com	Bag of Words, TF-IDF, Word2Vec, Doc2Vec	TF-IDF	TF-IDF outperformed Word2Vec and Doc2Vec.
Mandal et al. [5]	the challenge of computing similarity between legal documents	Supreme Court of India case statements	56 different approaches (traditional and advanced context-aware methods)	TF-IDF and LDA	TF-IDF and LDA outperformed more advanced approaches like Law2Vec and BERT in computing document-level similarity.
Singh and Shashi [6]	identifying and summarizing news articles related to top trending topics/hashtags	DUC 2004 benchmark dataset	TF-IDF, Word2Vec, Doc2Vec	TF-IDF	TF-IDF was the most effective in creating pure clusters for static datasets.
Zhu et al. [7]	need for a scholarly recommender system to aid scholars in identifying related literature	Gene Expression Omnibus (GEO) from PubMed	BM25, TF-IDF, Doc2Vec, ELMo, BERT	TF-IDF and BM25	TF-IDF and BM25 performed better compared to other techniques, including popular NLP embedding models.
Shahmirzadi et al. [8]	Patent-to-patent semantic similarity	collection of all (publicly available patents) from (USPTO) the United States Patent and Trademark-Office	Basic TF-IDF, extensions of TF-IDF, LSI, Doc2Vec	TF-IDF	TF-IDF was a suitable choice considering its performance and cost.
Vrbanec and Meštrović [9]	Semantic similarity detection	Webis Crowd Paraphrase Corpus 2011, Clough and Stevenson, and Microsoft Research Paraphrase Corpus.	USE, Glove, Word2Vec, ELMO, Fast-Text, Doc2Vec, LSI, TF-IDF	TF-IDF	TF-IDF exhibited superior performance compared to other models, including deep-learning methods.
Pranjic and Podpecˇan [10]	Link recommendation methods on the news archive	News archive from a popular Croatian website, 24sata.	TF-IDF, BERT, XLM-R, Doc2Vec, and latent semantic indexing	TF-IDF	TF-IDF provided better matches with manually selected links by journalists than more advanced methods (BERT, XLM-R, Doc2Vec, LSI).
Proposed work	detecting semantic similarity in theses and dissertations as long text documents	Two datasets consisting of 27 documents from Duhok Polytechnic University and 100 documents from ProQuest.com	TF-IDF, Doc2Vec , BERT, SBERT	TF-IDF	TF-IDF method was more effective and efficient than Doc2vec, BERT, and SBERT methods for measuring semantic similarity between theses and dissertations documents as long-text documents, with an accuracy of 98% and 97%.

REFERENCES

Balani, Z., & Varol, C. (2021). Combining Approximate String Matching Algorithms and Term Frequency In The Detection of Plagiarism. 9.

Brandt, J. (2019). Text mining policy: Classifying forest and landscape restoration policy agenda with neural information retrieval (arXiv:1908.02425). arXiv. http://arxiv.org/abs/1908.02425

Chawla, S., Aggarwal, P., & Kaur, R. (2022). Comparative Analysis of Semantic Similarity Word Embedding Techniques for Paraphrase Detection. In Emerging Technologies for Computing, Communication and Smart Cities (pp. 15–29). Springer.

Chicco, D. (2021). Siamese neural networks: An overview. Artificial Neural Networks, 73–94.

Davoodifard, M. (2022). Automatic Detection of Plagiarism in Writing. Studies in Applied Linguistics and TESOL, 21(2). https://doi.org/10.52214/salt.v21i2.9058

Dept. of Computer Science, Jigawa State Colledge of Education, Gumel, Nigeria, Abubakar, H. D., Umar, M., & Dept. of Computer Sceince, Faculty of Science, Sokoto State University, Sokoto, Nigeria. (2022). Sentiment Classification: Review of Text Vectorization Methods: Bag of Words, Tf-Idf, Word2vec and Doc2vec. SLU Journal of Science and Technology, 4(1 & 2), 27–33. https://doi.org/10.56471/slujst.v4i.266

Devlin, J., Chang, M.-W., Lee, K., & Toutanova, K. (2019). BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding (arXiv:1810.04805). arXiv. http://arxiv.org/abs/1810.04805

Goldberg, Y. (2017). Neural network methods for natural language processing. Synthesis Lectures on Human Language Technologies, 10(1), 1–309.

Jones, K. S. (1999). What is the Role of NLP in Text Retrieval? In T. Strzalkowski (Ed.), Natural Language Information Retrieval (Vol. 7, pp. 1–24). Springer Netherlands. https://doi.org/10.1007/978-94-017-2388-6_1

Le, Q. V., & Mikolov, T. (2014). Distributed Representations of Sentences and Documents (arXiv:1405.4053). arXiv. http://arxiv.org/abs/1405.4053

Magara, M. B., Ojo, S. O., & Zuva, T. (2018). A comparative analysis of text similarity measures and algorithms in research paper recommender systems. 2018 Conference on Information Communications Technology and Society (ICTAS), 1–5.

Malmberg, J. (2021). Evaluating semantic similarity using sentence embeddings. http://urn.kb.se/resolve?urn=urn:nbn:se:kth:diva-291425

Mandal, A., Ghosh, K., Ghosh, S., & Mandal, S. (2021). Unsupervised approaches for measuring textual similarity between legal court case reports. Artificial Intelligence and Law, 29(3), 417–451. https://doi.org/10.1007/s10506-020-09280-2

Mikolov, T., Chen, K., Corrado, G., & Dean, J. (2013). Efficient estimation of word representations in vector space. ArXiv Preprint ArXiv:1301.3781.

Ofer, D., Brandes, N., & Linial, M. (2021). The language of proteins: NLP, machine learning & protein sequences. Computational and Structural Biotechnology Journal, 19, 1750–1758. https://doi.org/10.1016/j.csbj.2021.03.022

P., S., & Shaji, A. P. (2019). A Survey on Semantic Similarity. 2019 International Conference on Advances in Computing, Communication and Control (ICAC3), 1–8. https://doi.org/10.1109/ICAC347590.2019.9036843

Park, K., Hong, J. S., & Kim, W. (2020). A methodology combining cosine similarity with classifier for text classification. Applied Artificial Intelligence, 34(5), 396–411.

Pranjic, M., & Podpecˇan, V. (2020). Evaluation of related news recommendations using document similarity methods. Digital Humanities, 6.

Reimers, N., & Gurevych, I. (2019). Sentence-BERT: Sentence Embeddings using Siamese BERT-Networks (arXiv:1908.10084). arXiv. http://arxiv.org/abs/1908.10084

Resta, O. A., Aditya, A., & Purwiantono, F. E. (2021). Plagiarism Detection in Students' Theses Using The Cosine Similarity Method. SinkrOn, 5(2), 305–313. https://doi.org/10.33395/sinkron.v5i2.10909

Shahmirzadi, O., Lugowski, A., & Younge, K. (2018). Text Similarity in Vector Space Models: A Comparative Study (arXiv:1810.00664). arXiv. http://arxiv.org/abs/1810.00664

Singh, A. K., & Shashi, M. (2019). Vectorization of Text Documents for Identifying Unifiable News Articles. International Journal of Advanced Computer Science and Applications, 10(7). https://doi.org/10.14569/IJACSA.2019.0100742

Sitikhu, P., Pahi, K., Thapa, P., & Shakya, S. (2019). A Comparison of Semantic Similarity Methods for Maximum Human Interpretability. 2019 Artificial Intelligence for Transforming Business and Society (AITB), 1–4. https://doi.org/10.1109/AITB48515.2019.8947433

Vrbanec, T., & Meštrović, A. (2020). Corpus-Based Paraphrase Detection Experiments and Review. Information, 11(5), 241. https://doi.org/10.3390/info11050241

Zhu, J., Patra, B. G., & Yaseen, A. (2021). Recommender system of scholarly papers using public datasets. AMIA Summits on Translational Science Proceedings, 2021, 672–679.