International Journal on Recent and Innovation Trends in Computing and Communication
ISSN: 2321-8169 Volume: 11 Issue: 11s
DOI: https://doi.org/10.17762/ijritcc.v11i11s.8094
***Article Received:** 26 June 2023 **Revised:** 21 August 2023 **Accepted:** 04 September 2023*

A Systematic and Comparative Analysis of Semantic Search Algorithms

Dr. Priya Shelke[1], Chaitali Shewale[2], Riddhi Mirajkar[3], Suruchi Dedgoankar[4], Pawan Wawage[5], Riddhi Pawar[6]
Department of Information Technology, Associate Professor, Vishwakarma Institute of Information Technology, Pune[1,2,3,4,5,6]

Abstract— Users often struggle to discover the information they need online because of the massive volume of data that is readily available as well as being generated every day in the today's digital age. Traditional keyword-based search engines may not be able to handle complex queries, which could result in irrelevant or insufficient search results. This issue can be solved by semantic search, which utilises machine learning and natural language processing to interpret the meaning and context of a user's query. In this paper we focus on analyzing the BM-25 algorithm, Mean of Word Vectors approach, Universal Sentence Encoder model, and Sentence-BERT model on the CISI Dataset for Semantic Search Task. The results indicate that, the Finetuned SBERT model performs the best.

Keywords- Natural language processing, Semantic Search, Information Retrieval.

I. INTRODUCTION

In the current digital era, there is an enormous amount of data available and accessible through the internet, which can makeit challenging for users to quickly and effectively obtain the information they need. The user experience can be frustratingwhen using traditional keyword-based search engines because they are frequently unable to handle complicated queries and can return irrelevant or inadequate search results. The answer to this issue is semantic search, which analyses user queries using machine learning and natural language processing to determine their intent and context. Semantic search is advantageous and necessary since it can result in more precise and pertinent search results, particularly for difficult or ambiguous questions, resulting in an improved user experience and better results. Additionally, it can aid companies and organizations in managing their data and knowledge resources more effectively, which will result in better decisions and increased efficiency.

The necessity of efficiently and quickly getting the necessary information can be understood by the study and survey conducted by statista which states that 'by 2025, it is anticipated that the amount of data generated, consumed, copied, and stored will exceed 180 zettabytes'. Due to the COVID-19 epidemic, which forced the majority of the world's population to work from home and use the internet for both business and enjoyment, global data generation increased dramatically in 2020. The total amount of data generated and consumed in 2020 was 64.2 zettabytes [1].

Semantic search is a sort of search that analyses the meaning of search queries and the content of web pages, documents, and other sources of information using natural language processing (NLP) and machine learning techniques. Semantic search seeks to comprehend the context and intent behind a query to give more accurate and relevant results, in contrast to standard keyword-based search, which matches search terms with the precise words or phrases found in documents.

Sentiment analysis, entity linking, topic modelling, named entity identification, and semantic parsing are just a few of the methods used by semantic search systems to evaluate and comprehend the meaning of text. These systems can better search results by discovering associations between concepts and things by examining the structure, syntax, and semantics of text. The ability to search for information using natural language queries rather than depending on specific keywords or phrases is one of the main advantages of semantic search. In addition to making search more user-friendly and simple, this can also assist surface information that might not have been discovered using conventional keyword-based search techniques.

Several applications, including web search, enterprise search, e-commerce search, and recommendation systems, use semantic search. As academics attempt to create more sophisticated methods for comprehending and interpreting human language, it is also becoming more and more significant in domains like natural language processing, machine learning, and artificial intelligence. Because semantic search enhances the precision and relevancy of search results, it can be helpful in daily chores. Conventional keyword-based search engines work by comparing the exact words in a query

International Journal on Recent and Innovation Trends in Computing and Communication
ISSN: 2321-8169 Volume: 11 Issue: 11s
DOI: https://doi.org/10.17762/ijritcc.v11i11s.8094
***Article Received:** 26 June 2023 **Revised:** 21 August 2023 **Accepted:** 04 September 2023*

to the words in documents, which can result in results that are not relevant if the query words do not adequately reflect the meaning of the search task. Semantic search, in contrast, seeks to comprehend the meaning of the query and the pages being searched, and it makes use of this comprehension to provide more accurate and pertinent results. Semantic search can help to catch these nuances and deliver more accurate results, which is vital in day-to-day tasks where individuals frequently use different words and phrases to describe the same thing. Consider the scenario when you are looking for an Italian restaurant close to your house. With the terms "restaurant," "Italian," and "near me," a keyword-based search engine would offer results based on exact matches, which might include eateries that serve Italian food far from your location. A semantic search engine, on the other hand, might be able to comprehend that you're looking for an Italian restaurant within a specific driving distance of where you are, and it will likely return more accurate and pertinent results. In daily activities like research, where people may need to locate pertinent data on a specific subject, semantic search can be helpful. Semantic search can aid in the weeding out of pointless results and the provision of more accurate and valuable information by comprehending the meaning of the search question and the pages being searched. Semantic search is helpful in everyday tasks because it helps to increase the relevance and accuracy of search results, which can help individuals locate the information they need more quickly and with less effort.

This paper focuses on analyzing the BM-25 algorithm, Mean of Word Vectors approach, Universal Sentence Encoder model, and Sentence-BERT model on the CISI Dataset [2] for Semantic Search Task. The rest of this paper is laid out as follows. The related research in the natural language processing (NLP) field is discussed in Section 2, with an emphasis on the Semantic Search task. Section 3 explains the dataset feature and the techniques utilized in this paper. The experimental results and analysis are reported in Section 4. Finally, Section 5 brings this paper to a close.

II. LITERATURE SURVEY

NLP has been the subject of extensive research in recent years. This offers a wide range of models for semantic search tasks. Some of them are summarized below:

The survey [3] highlights and examines a number of popular research areas in semantic search, including the RDF model, simple concept location, fuzzy logic formalisms, and fuzzy concepts, all of which enable the combination of keyword search results as partners in complex constraint querying. It also extracts the most commonly used methodology in them.

This study [4] examines many existing systems and divides them into various different groups based on their methodology, reach, and functionality. Ontology learning and entity consolidation are two of the framework's components that are discussed in detail. It also suggests an extensible and flexible framework that addresses frequent tasks and problems in the associated research.

This survey [5] provides a brief summary of several of the early semantic search engines, including Kosmix, XCDSearch, Hakia, and Swoogle.

The most important topic covered in this paper [6] is the structure of Semantic Web documents, which allows us to represent semantic in a machine-readable manner. This paper explores the particular problems with structured information retrieval and makes recommendations for how weighting schemas might be changed to improve semantic document retrieval.

This survey [7] shows the research of the primary literature on semantic search technology by dividing it into six major categories and examining each individual categories characteristics. In addition, four perspectives are used to analyse and draw conclusions about the problems with the assessed semantic methodologies and engines.

In-depth analysis of the various ranking functions used in information retrieval are provided in the study [8]. The characteristics term frequency, inverse document frequency, and length normalisation are frequently employed in information retrieval.

The study [9] explains popular vector-space information retrieval method which is Term Frequency and Inverse Document Frequency, sometimes known as TF-IDF. It weighs how closely the query and the document resemble one other and penalises the use of common phrases.

The paper [10] proposed another way where shorter documents are rewarded using TF-centered IDF's document length normalisation.

The paper [11] proposes another state-of-the-art retrieval technique called Okapi BM25, which is a complicated variant of pivoting length normalisation.

The principles for building generalised ranking functions with application-specific features are provided in the work [12]. With the help of feature engineering principles and the provided data set, the paper prescribes a specific case of a generalised function for recommendation system. On the unstructured textual data, the behaviour of both generalised and particular functions is investigated and put into practise. The ranking algorithm based on proximity features has outperformed the standard BM25 by 52%.

International Journal on Recent and Innovation Trends in Computing and Communication
ISSN: 2321-8169 Volume: 11 Issue: 11s
DOI: https://doi.org/10.17762/ijritcc.v11i11s.8094
***Article Received:** 26 June 2023 **Revised:** 21 August 2023 **Accepted:** 04 September 2023*

The paper [13] presents a technique for categorising text sentiment that combines Term Frequency-Inverse Document Frequency (TF-IDF) and Next Word Negation (NWN). For text classification, it compares the results of binary bag of words model, TF-IDF model, and TF-IDF with next word negation (TF-IDF-NWN) model. It proposes that the TF-IDF-NWN model performs best when coupled with a linear SVM.

The paper [14] proposes two novel model architectures: Continuous Bag-of-Words Model and Continuous Skip-gram Model for calculating continuous vector representations of words from very large data sets and comparing the results to the prior top performing methods based on several different types of neural networks. The proposed models observe large improvements in accuracy at much lower computational cost. Additionally, it demonstrates that these vectors offer state-of-the-art performance for determining word similarity in both syntactic and semantic terms

The study [15] proposes a novel method based on the skipgram model, where each word is represented as a bag of character n-grams. Each character in an n-gram has a corresponding vector representation, and words are represented as the sum of these representations. The suggested approach is quick, enabling speedy model training on huge corpora and word representation computation for terms that did not exist in the training data. By giving each word its own vector, this gets beyond the drawbacks of prevalent models that neglect the morphology of words.

The research [16] investigates a straightforward and effective baseline for text categorization. Its experiments done using fast text classifier, or fastText, reveal that it is significantly quicker for both training and evaluation than deep learning classifiers while frequently matching their accuracy. With a normal multicore CPU, fastText can train on more than one billion words in less than ten minutes and classify half a million sentences among 312K classes in under a minute.

The paper [17] suggests a brand-new, straightforward network architecture called the Transformer that relies purely on attention mechanisms and does away with both recurrence and convolutions. Tests on two machine translation tasks reveal that these models outperform other models in terms of quality while being more parallelizable and taking much less time to train. Additionally, it demonstrates how well the Transformer generalises to various other tasks.

This study [18] presents two models: transformer based sentence encoding model and Deep Averaging Network (DAN) model for encoding sentences into embedding vectors with the sole purpose of facilitating transfer learning to other NLP tasks. The proposed encoding models allow for accuracy and computation resource trade-offs.

Recent research [19] using pre-trained sentence level embeddings has shown high transfer task performance.

The research [20] developed a language representation model known as Bidirectional Encoder Representations from Transformers (BERT), which aims to jointly condition on both left and right context in all layers in order to pretrain deep bidirectional representations from unlabelled text. It achieves state-of-the-art results on eleven natural language processing activities. Without making significant task-specific architecture alterations, the pre-trained BERT model can be further finetuned with just one extra output layer to develop cutting-edge models for a wide range of tasks.

The paper [21] present Sentence-BERT (SBERT), a modification of the pretrained BERT network, generates semantically significant sentence embeddings that can be compared using cosine-similarity. It does this by employing siamese and triplet network architectures. With the accuracy of BERT, this cuts down on the time needed for finding the most similar pair from 65 hours with BERT / RoBERTa to around 5 seconds with SBERT.

III. METHODOLOGY

A. Dataset

The CISI dataset [2], a text-based dataset made publicly available by the University of Glasgow, is used for information retrieval (IR). The Centre for Inventions and Scientific Information ("CISI") has gathered this information. There are three files in this dataset: CISI.ALL, CISI.QRY, and CISI.REL. The first file has text information for 1,460 documents, each of which has a unique ID, title, author, abstract and list of cross-references to other documents and the second file has 112 related queries with unique ID and query text. The query-document matching ground truth is contained in the file CISI.REL, which may be used to compare trained models and assess their performance.

SR No.	File	Description
1	CISI.ALL	A file of 1,460 "documents" each with a unique ID (.I), title (.T), author (.A), abstract (.W) and list of cross-references to other documents (.X)
2	CISI.QRY	A file containing 112 queries each with a unique ID (.I) and query text (.W)
3	CISI.REL	A file containing the mapping of query ID (column 0) to document ID (column 1). A query may map to more than one document ID. This file contains the "ground truth" that links queries to documents.

International Journal on Recent and Innovation Trends in Computing and Communication
ISSN: 2321-8169 Volume: 11 Issue: 11s
DOI: https://doi.org/10.17762/ijritcc.v11i11s.8094
***Article Received:** 26 June 2023 **Revised:** 21 August 2023 **Accepted:** 04 September 2023*

B. Data Pre-processing

In order to increase the precision and dependability of machine learning models, data preparation is crucial. It entails cleaning and modifying unprocessed text data, which makes it simpler to draw out important conclusions and patterns from the data. Preprocessing also aids in reducing the amount of data, which can increase the effectiveness of NLP algorithms and shorten computation times. Data preparation has a big impact on the quality and effectiveness of machine learning model. The data preprocessing steps listed below are used to prepare data before feeding it to machine learning models.

The text is cleaned up in the text cleaning step, which also lowercases all of the text and removes all symbols and numbers (including punctuation). Next using python package NLTK, separate the text into individual words or tokens to easily analyse and manipulate the text data which is done with the help of word tokenizer. Stop word removal is then used to reduce the text data and boost the accuracy of machine learning models. WordNetLemmatizer from NLTK is then used to condense words to their base form in order to capture their basic meaning.

C. BM-25

The BM25 method is a probabilistic model and ranking function used to determine how relevant a document is for a particular query. BM25 stands for Best Match 25. BM25 works by awarding a relevance score to each document depending on how well it match the query. The frequency of the query words within that text, the length of the document, and the frequency of the query terms throughout the corpus are all taken into account when calculating the relevance score. To avoid term frequency saturation, which occurs when documents with high term frequencies are overemphasised, the algorithm considers both the frequency of the query terms in the document and the inverse frequency of the query terms in the corpus. It is frequently used as a benchmark algorithm to assess more advanced methods.

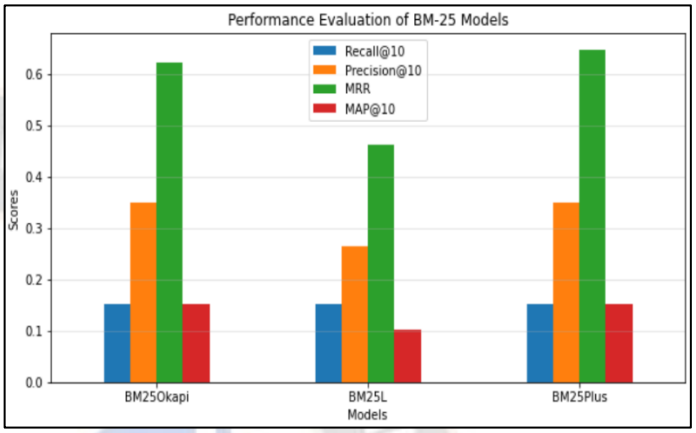
D. BM25F

A variant of standard BM25, has length normalisation, term relevance saturation, and the assumption that the document is made up of many fields with potentially varying degrees

of importance are all taken into consideration[22][23]. **BM25+** another modification of BM25, corrects a flaw in the standard BM25 that causes long documents that match the query term to sometimes be unfairly scored as being equally relevant to shorter documents that don't contain the query term at all. It introduces one additional free parameter delta [24].

The performance of the BM-25 algorithm and its modifications on the CISI dataset is shown in the table below.

SR No.	Model Name	Recall@10	Precision@10	MRR	MAP@10
1	BM25Okapi	0.15	0.35	0.62	0.15
2	BM25L	0.15	0.27	0.46	0.10
3	BM25Plus	0.15	0.35	0.65	0.15



E. Mean of Word Vectors

In order to express the meaning of a text document in semantic search, the Mean of Word Vectors (MWV) approach provides a straightforward and efficient method. It involves representing a document as the word vector average of the words from document. Using a pre-trained word embedding model, each word in the document is initially represented as a high-dimensional vector representing each word's semantic meaning within the context of a huge text corpus. To create a single vector representation of the document, the word vectors in the document are averaged. Based on the meanings of the individual words, this vector depicts the document's overall meaning.The **word2vec** algorithm learns word correlations from a huge corpus of text using a neural network model. It is a two-layer, shallow neural network that has been trained to reconstruct word contexts in linguistic discourse. Using a huge corpus of text as input, Word2vec creates a vector space, generally with several hundred dimensions, and assigns each distinct word in the corpus a corresponding vector in the space. Word2vec can generate distributed representations of words using either the continuous bag-of-words (CBOW) model architecture or the continuous skip-gram model architecture [14][25].

Global Vectors, or **GloVe**, is a distributed word representation model. It is unsupervised method of generating word vector representations which arranges words in a useful space where the distance between them is correlated with their semantic closeness. The representations produced by GloVe training on global word-word co-occurrence statistics from a corpus

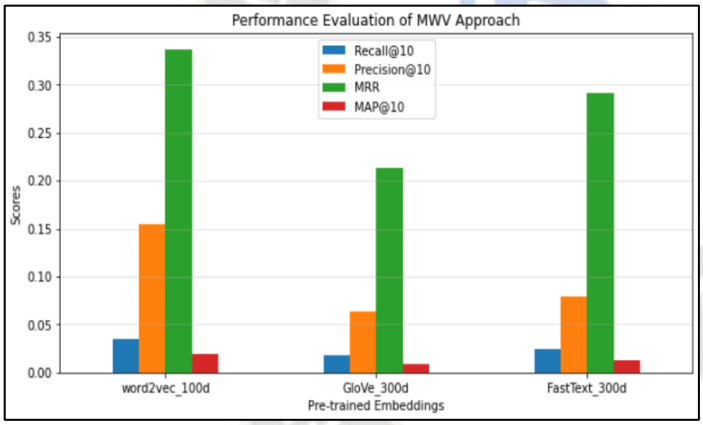
International Journal on Recent and Innovation Trends in Computing and Communication
ISSN: 2321-8169 Volume: 11 Issue: 11s
DOI: https://doi.org/10.17762/ijritcc.v11i11s.8094
***Article Received:** 26 June 2023 **Revised:** 21 August 2023 **Accepted:** 04 September 2023*

demonstrate the word vector space's linear substructures. [26][27].

FastText is a library developed by Facebook Research for the rapid learning of word representations and text categorization. Both supervised (classifications) and unsupervised (embedding) word and phrase representations are supported by FastText. It offers models that have been trained for 157 distinct languages [16].

The performance result for the Mean of Word Vectors approach on the CISI dataset using word2vec, GloVe, and FastText pre-trained word embeddings is shown in the table below.

SR No.	Model Name	Recall@10	Precision@10	MRR	MAP@10
1	word2vec_100d	0.04	0.16	0.34	0.02
2	GloVe_300d	0.02	0.06	0.21	0.01
3	FastText_300d	0.03	0.08	0.29	0.01



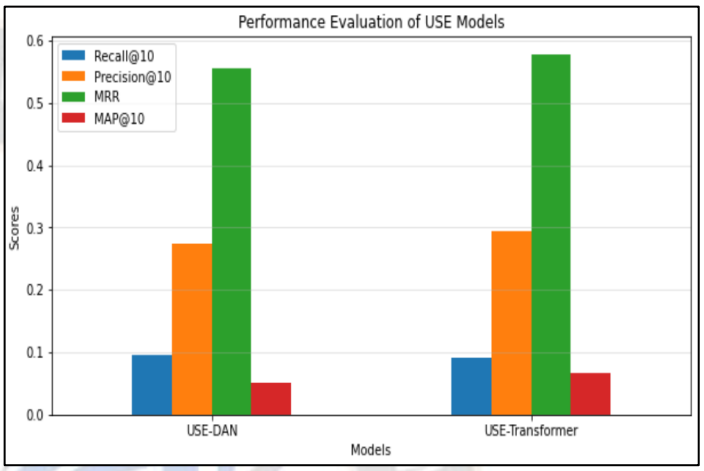
F. Universal Sentence Encoder

The Universal Sentence Encoder (USE), a pre-trained model created by Google, is intended to encode the semantic meaning of a phrase into a fixed-length vector. A sentence is fed into the USE, which outputs a fixed-length vector representation of the sentence. This vector can be used to determine the semantic similarity between sentences or to find most similar document from the collection of documents [18].

There are two versions of the pre-trained Universal Sentence Encoder: one trained using **Transformer encoder** and the other trained with **Deep Averaging Network (DAN)**. Accuracy and the need for processing resources are trade-offs between the two. Although the one with the Transformer encoder is more accurate, it requires more calculation. The one that uses DAN encoding is less accurate and computationally costly.

The performance result of pretrained Universal Sentence Encoder models accessed from TensorFlow Hub, namely Transformer encoder and Deep Averaging Network (DAN), on the CISI dataset is shown in the table below.

SR No.	Model Name	Recall@10	Precision@10	MRR	MAP@10
1	USE-DAN	0.09	0.27	0.56	0.05
2	USE-Transformer	0.09	0.29	0.56	0.07



G. SBERT

The Sentence-BERT (SBERT) is a modification of the standard pretrained BERT network. It is a pre-trained model that encodes the semantic meaning of sentences into fixed-length vectors. It is based on a transformer architecture and is trained on a large corpus of text using diverse range of unsupervised learning techniques. It builds sentence embeddings for each sentence using siamese and triplet networks, which can then be compared using a cosine-similarity [21]. It allows for the feasibility of semantic search for a huge number of sentences (only requiring a few seconds of training time). The siamese neural networks used are unique networks that comprise two or more subnetworks that are similar to one another, with the two models sharing the same parameters/weights and both the sub-models update their parameters in the same manner.

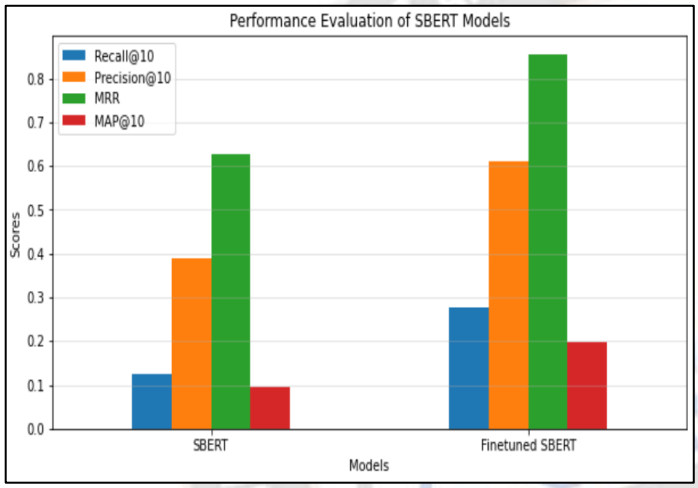
For finetuning SBERT, the training data is prepared by using the **InputExample** class which stores each training sample as a list of strings denoting sentence pairs, along with a label indicating their semantic similarity. The standard PyTorch **DataLoader** is then used to wrap this training data, which allows shuffling the data and creating batches of a specific size. In order to fine-tune the network to recognize the similarity of sentences we apply **CosineSimilarityLoss**. For each phrase pair, this loss determines the cosine similarity of the embeddings u and v created by passing sentences A and B

International Journal on Recent and Innovation Trends in Computing and Communication
ISSN: 2321-8169 Volume: 11 Issue: 11s
DOI: https://doi.org/10.17762/ijritcc.v11i11s.8094
***Article Received:** 26 June 2023 **Revised:** 21 August 2023 **Accepted:** 04 September 2023*

through the SBERT network. The SBERT model is trained for a single epoch and warms up for the first 10% of our training steps, with a list of train objectives consisting of tuples containing a dataloader and a loss function object passed as input.

The performance result of pretrained Sentence BERT and finetuned Sentence BERT model accessed from HuggingFace, on the CISI dataset is shown in the table below.

SR No.	Model Name	Recall@10	Precision@10	MRR	MAP@10
1	SBERT	0.13	0.39	0.63	0.09
2	Finetuned SBERT	0.28	0.61	0.86	0.19



IV. RESULT

In this research, offline evaluation metrics for information retrieval are utilised to assess the performance of different models. There are two sorts of offline metrics: order aware and order unaware. The Top-K predicted outcomes' ranking or order are not taken into account by the order unaware metrics. It just determines if the true relevant result was included in the predicted results; this yields the same results whether the true relevant result is ranked first or fifth in the Top-5 predicted results. When using order-aware metrics, the true relevant result at the first position of the predicted Top-5 results would be given a greater score than the true relevant result at the fifth position. This study compares model performance using both order unaware (Recall@K, Precision@K) and order aware metrics namely Mean Reciprocal Rank (MRR), Mean Average Precision@K (MAP@K), with a greater emphasis on order aware metrics scores [28].

The first approach we investigated for the semantic search task is BM25, which is a probabilistic model and superior to other algorithms like TF-IDF in terms of semantic search. When determining how relevant a document is to a query, it considers both the context in which a term appears inside a document and the overall frequency of the term across all documents in the corpus. It can also account for uncertainty in the relevance of documents to a query. When working with big and varied document collections, where there may be a great deal of variations in the quality and relevance of documents, this is extremely helpful. The model with the highest performance among the BM-25 algorithm variants is

BM-25 Plus and has an MRR score of 0.65 and a MAP@10 score of 0.15.

Instead of only matching keywords, vectorization-based techniques capture the semantic meaning of words and phrases. In addition to being a straightforward and effective method, vectorization-based approaches are more adaptable and better able to handle the unpredictability of human language. They are also a simple and effective technique that can better capture the semantic meaning of text and handle the diversity of human language. The second approach we explored is Mean of Word Vectors (MWV) which is straightforward and effective, although it has certain drawbacks. For instance, it does not consider the relationships between the words in the text or their order. The utilisation of MWV models is sometimes constrained by their inability to handle terms that are not part of their vocabulary. However, the specific model that performs best depends on the specific task and data being used. In this scenario, the MWV technique performed poorly in comparison to other algorithms for semantic search utilising the CISI Dataset. The model with the highest performance among the Mean of Word Vectors approach is the one using word2vec embeddings with MRR 0.34 score of and MAP@10 score of 0.02.

Next, we looked into pretrained Universal Sentence Encoder (USE) models, which can encode contextual information like word order and position of words in a sentence, sentence structure, and word relationships, allowing them to capture meaning that is more subtle. This enables them to handle out-of-vocabulary terms, or words that are words that are not contained in the training data, and capture more nuanced meanings. USE models have outperformed the MWV approach for semantic search tasks as they are able to capture more nuanced meaning, handle syntax and out-of-vocabulary words, and benefit from multi-task learning. USE model with Transformer encoder architecture surpasses USE model with Deep Averaging Network (DAN) architecture with MRR score of 0.56 and MAP@10 score of 0.07.

The Sentence-BERT (SBERT) model is based on a transformer architecture that is able to capture contextual information about words and sentences, which is important for

International Journal on Recent and Innovation Trends in Computing and Communication
ISSN: 2321-8169 Volume: 11 Issue: 11s
DOI: https://doi.org/10.17762/ijritcc.v11i11s.8094
***Article Received:** 26 June 2023 **Revised:** 21 August 2023 **Accepted:** 04 September 2023*

semantic search tasks as it requires understanding the meaning of text in context. As SBERT is designed to be fine-tuned on specific tasks, it can adapt to the specific characteristics of the data and improve its performance on those tasks. Pretrained SBERT and Finetuned SBERT models performed better than the pretrained Universal Sentence Encoder (USE) model in semantic search tasks because of its ability to be fine-tuned, its ability to capture contextual information, and its training on a larger and more diverse dataset. Finetuned SBERT model outscored the pretrained SBERT model with MRR score of 0.86 and MAP@10 score of 0.19.

Table shows all of the metric values for the various models that were used. This data is represented graphically in table.

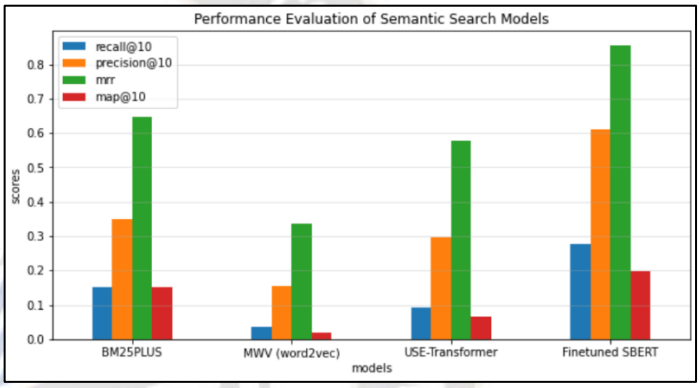
This study [18] presents two models: transformer based sentence encoding model and Deep Averaging Network (DAN) model for encoding sentences into embedding vectors with the sole purpose of facilitating transfer learning to other NLP tasks. The proposed encoding models allow for accuracy and computation resource trade-offs.

Recent research [19] using pre-trained sentence level embeddings has shown high transfer task performance.

The research [20] developed a language representation model known as Bidirectional Encoder Representations from Transformers (BERT), which aims to jointly condition on both left and right context in all layers in order to pretrain deep bidirectional representations from unlabelled text. It achieves state-of-the-art results on eleven natural language processing activities. Without making significant task-specific architecture alterations, the pre-trained BERT model can be further finetuned with just one extra output layer to develop cutting-edge models for a wide range of tasks.

The paper [21] present Sentence-BERT (SBERT), a modification of the pretrained BERT network, generates semantically significant sentence embeddings that can be compared using cosine-similarity. It does this by employing siamese and triplet network architectures. With the accuracy of BERT, this cuts down on the time needed for finding the most similar pair from 65 hours with BERT / RoBERTa to around 5 seconds with SBERT.

SR No.	Model Name	Recall@ 10	Precision@ 10	MR R	MAP@ 10
1	BM25Okapi	0.15	0.35	0.62	0.15
2	BM25L	0.15	0.27	0.46	0.10
3	BM25Plus	0.15	0.35	0.65	0.15
4	word2vec_100d	0.04	0.16	0.34	0.02
5	GloVe_300d	0.02	0.06	0.21	0.01
6	FastText_300d	0.03	0.08	0.29	0.01
7	USE-DAN	0.09	0.27	0.56	0.05
8	USE-Transformer	0.09	0.29	0.56	0.07
9	SBERT	0.13	0.39	0.63	0.09
10	Finetuned SBERT	0.28	0.61	0.86	0.19



V. CONCLUSION

In this paper, we evaluated the BM-25 algorithm, Mean of Word Vectors approach, Universal Sentence Encoder model, and Sentence-BERT model on the CISI Dataset [2] for Semantic Search Task. The Mean of Word Vectors (MWV) technique utilising word2vec embeddings offers the best performance out of all the MWV approaches we evaluated, with an MRR score of 0.34 and a MAP@10 score of 0.02. With MRR of 0.56 and MAP@10 of 0.07, the USE model with Transformer encoder architecture outperforms the USE model with Deep Averaging Network (DAN) architecture. The BM-25 Plus algorithm, which has an MRR score of 0.65 and a MAP@10 score of 0.15, is the best performing of the BM-25 algorithm versions. With an MRR score of 0.86 and a MAP@10 score of 0.19, the fine-tuned SBERT model outperformed the pretrained SBERT model. So, from our experiments, we have concluded that the fine-tuned SBERT model performs best with an MRR score of 0.86 and a MAP@10 score of 0.19, when the order-aware performance metrics scores are taken into account.

International Journal on Recent and Innovation Trends in Computing and Communication
ISSN: 2321-8169 Volume: 11 Issue: 11s
DOI: https://doi.org/10.17762/ijritcc.v11i11s.8094
***Article Received:** 26 June 2023 **Revised:** 21 August 2023 **Accepted:** 04 September 2023*

REFERENCES

[1] "Total data volume worldwide 2010-2025 | Statista." https://www.statista.com/statistics/871513/worldwide-data-created/ (accessed Apr. 25, 2023).

[2] "CISI (a dataset for Information Retrieval) | Kaggle." https://www.kaggle.com/datasets/dmaso01dsta/cisi-a-dataset-for-information-retrieval (accessed Apr. 25, 2023).

[3] E. Mäkelä, "Survey of Semantic Search Research." [Online]. Available:

[4] W. Wei, P. M. Barnaghi, and A. Bargiela, "Search with Meanings: An Overview of Semantic Search Systems." [Online]. Available: http://www.w3.org/TR/owl-guide/

[5] Lee, C.-H. ., Noh, H.-R. ., & Kim, K.-C. . (2023). Design of Torque and Power Density Improvement According to the Rotor Shape of IPMSM. International Journal of Intelligent Systems and Applications in Engineering, 11(4s), 174–179. Retrieved from https://ijisae.org/index.php/IJISAE/article/view/2585.

[6] G. Sudeepthi, G. Anuradha, M. B.-I. J. of Computer, and undefined 2012, "A survey on semantic web search engine," Citeseer, 2012, Accessed: Apr. 25, 2023. [Online].

[7] J. R. Pérez-Agüera, J. Arroyo, J. Greenberg, J. P. Iglesias, and V. Fresno, "Using BM25F for semantic search," ACM International Conference Proceeding Series, 2010, doi: 10.1145/1863879.1863881.

[8] Thakre, B., Thakre, R., Timande, S., & Sarangpure, V. (2021). An Efficient Data Mining Based Automated Learning Model to Predict Heart Diseases. Machine Learning Applications in Engineering Education and Management, 1(2), 27–33. Retrieved from http://yashikajournals.com/index.php/mlaeem/article/view/17

[9] H. Dong, F. Hussain, E. C.-2008 2nd I. international, and undefined 2008, "A survey in semantic search technologies," ieeexplore.ieee.org, 2008, Accessed: Apr. 25, 2023. [Online]. Available:

[10] C. Zhai and S. Massung, "Text Data Management and Analysis: A Practical Introduction to Information Retrieval and Text Mining June 2016 https://doi. org/10.1145/2915031.2915054," dl.acm.org, Accessed: Apr. 25, 2023. [Online]..

[11] H. Wu, R. Luk, K. Wong, K. K.-A. T. on, and undefined 2008, "Interpreting tf-idf term weights as making relevance decisions," dl.acm.org, vol. 26, no. 3, Jun. 2008, doi: 10.1145/1361684.1361686

[12] Sherje, D. N. . (2021). Content Based Image Retrieval Based on Feature Extraction and Classification Using Deep Learning Techniques. Research Journal of Computer Systems and Engineering, 2(1), 16:22. Retrieved from https://technicaljournals.org/RJCSE/index.php/journal/article/view/14

[13] D A. Singhal, C. Buckley, and M. Mitra, "Pivoted document length normalization," SIGIR Forum (ACM Special Interest Group on Information Retrieval), pp. 21–29, 1996:

[14] S. Robertson, S. W.-S. P. of the Seventeenth, and undefined 1994, "Some simple effective approximations to the 2-poisson model for probabilistic weighted retrieval," Springer, pp. 232–241, Aug. 1994

[15] P. Agrawal, "Exploration of Proximity Heuristics in Length Normalization," Jan. 2017, [Online]. Available: http://arxiv.org/abs/1701.01417.