Enhancing Cross-Media Semantic Text Similarity Estimation in Dynamic Data: A Weighted Hashing Algorithm Comparison

1st Prof. ***Dr. ST Patil***
Department of Computer Engineering
Vishwakarma Institute of Technology, Pune, India
Patil.st@vit.edu

2nd ***Kaivalya Aole***
Department of Computer Engineering
Vishwakarma Institute of Technology, Pune, India
Kaivalya.aole20@vit.edu

3rd ***Abu Ansari***
Department of Computer Engineering
Vishwakarma Institute of Technology, Pune, India
Abu.ansari20@vit.edu

4th ***Aarya Tiwari***
Department of Computer Engineering
Vishwakarma Institute of Technology, Pune, India
Aarya.tiwari20@vit.edu

5th ***Harshal Abak***
Department of Computer Engineering
Vishwakarma Institute of Technology, Pune, India
Harshal.abak20@vit.edu

Abstract

Amidst the relentless and ever-accelerating expansion of digital content diversity and dynamism, the need for accurate cross-media semantic text similarity estimation has become increasingly important. This research focuses on enhancing the measurement of textual similarity across different media types, particularly text and images, while considering the dynamic nature of the data. To achieve this objective, we undertake a comprehensive comparison of various weighted hashing algorithms. Notably, our research incorporates a survey to gather human input for text similarity assessment, providing a holistic understanding of the algorithms' performance. The study addresses the intricate task of comparing textual content across heterogeneous media forms, rendering its implications relevant in fields such as multimedia retrieval, content recommendation, and information retrieval. The research explores how these weighted hashing algorithms can adeptly adapt to the evolving nature of data, a critical aspect for real-time applications and systems dealing with continuously updated content. Importantly, the paper contributes valuable insights by presenting a comparative analysis of different weighted hashing techniques, thereby refining our approach to measuring similarity between textual content across diverse media formats. This research caters to the demands of a dynamic digital landscape, offering a nuanced perspective on the efficacy of weighted hashing algorithms in enhancing cross-media text similarity estimation.

Keywords: *BERT, Content Recommendation, Cosine Similarity, Cross-media, dynamic data, Kendall's Tau, semantic text analysis, TF-IDF.*

1. Introduction

In today's expansive digital landscape, the amalgamation of text and images across various platforms has given rise to an unparalleled array of multimedia content. Deciphering the semantic likeness between textual and visual elements within this vast sea of information poses both a challenge and a necessity. Accurate measurement of cross-media semantic text similarity holds pivotal importance across diverse domains, spanning from multimedia retrieval to content recommendation and information retrieval.

This research is dedicated to enhancing the estimation of cross-media text similarity, recognizing the foundational role of Term Frequency-Inverse Document Frequency (TF-IDF) while acknowledging the growing need for more adaptive methodologies in our digital age. The study explores a spectrum of techniques, ranging from conventional measures like cosine similarity to more advanced methodologies such as machine learning models (Random Forest, Logistic Regression, Support Vector Machines), Informational Semantic Hashing (ISH), Word Mover's Distance (WMD), and the integration of BERT embeddings.

This exploration aims to uncover the distinctive contributions of these methodologies in measuring cross-media text similarity, evaluating their individual performance, strengths, and weaknesses. By employing tools like Kendall's Tau, the study seeks not only to gauge the efficacy of these techniques but also to assess their correlation and consistency in ranking similarity scores. Understanding the effectiveness of these diverse methodologies within the dynamic digital landscape is pivotal for achieving precise semantic text similarity estimation.

In a world where information constantly evolves, this study serves as a guide, offering an in-depth analysis of advanced methods and their applicability. The ultimate goal is to provide insights into how these methodologies can effectively adapt and be utilized in an ever-changing information ecosystem, where precise semantic text similarity estimation is indispensable.

2. Litreture survey

Fei Lan in their research introduces an innovative hybrid algorithm, merging term semantic information with the traditional TF-IDF method, to enhance text similarity measurement. In response to the limitations of TF-IDF, particularly its inability to capture semantic nuances, the proposed algorithm leverages the term similarity weighting tree (TSWT) data structure and semantic information from HowNet. Through meticulous text preprocessing and filtering, key terms are identified, and text similarities are calculated based on feature weights exceeding a predetermined threshold. Experimental evaluations across various K-means clustering methods showcase the algorithm's superiority over pure TF-IDF and semantic understanding methods, exhibiting improved accuracy, recall, and F1-metric. The results affirm the efficacy of this hybrid approach, offering a promising solution for accurate text similarity measurement in diverse natural language processing applications [1].

In their groundbreaking study, Hao Liu and Xi Chen present an innovative approach to text sentiment analysis by introducing a weight distributing method that combines rule-based sentiment dictionaries with machine learning using TF-IDF. Addressing the limitations of each individual method, the proposed approach enhances sentence vectors, accentuating sentiment-laden words while retaining vital text information. Empirical

findings underscore the method's efficacy, showcasing a notable 82.1% accuracy rate in text sentiment analysis. This substantial improvement surpasses both the rule-based sentiment dictionary method (by 13.9%) and the TF-IDF weighting method (by 7.7%). Liu and Chen's novel integration of sentiment analysis methods represents a significant stride in achieving heightened accuracy and comprehensiveness in deciphering sentiment within textual data [2].

In their study, Luiz Gomes and Ricardo da Silva Torres address the critical issue of predicting long-lived bugs in Free/Libre Open-Source Software (FLOSS) to assist maintenance teams in planning and improving software quality. Focusing on the comparative analysis of feature extraction methods, the research compares the effectiveness of Bidirectional Encoder Representations from Transformers (BERT) and Term Frequency-Inverse Document Frequency (TF-IDF) in long-lived bug prediction using various machine learning classifiers. The findings reveal that BERT-based feature extraction consistently outperforms TF-IDF, with Support Vector Machines (SVM) and Random Forest standing out as superior classifiers across datasets when utilizing BERT. Notably, even smaller BERT architectures demonstrate competitiveness [3].

Gisela Yunanda and Dade Nurjanah addressed the challenge of information overload in rapidly growing news portals by proposing a recommendation system utilizing TF-IDF and Cosine Similarity methods. Recognizing the diminishing time relevance of news, the system employs TF-IDF to assign weights to words in news titles and then evaluates similarity through cosine similarity. To validate the accuracy of the recommendation system, the study matches its results with the reader's actual news history on the Microsoft News online portal, using a hit-rate metric. Impressively, the study reports an 80.77% hit-rate, indicating the system's effectiveness in recommending news aligned with readers' preferences [4].

Yudi Setiawan researched the feature extraction for text documents, particularly focusing on the challenging task of cyberbullying text classification. Acknowledging the complexities inherent in natural language and machine learning classifications, Setiawan emphasizes the significance of effective feature extraction to capture essential text elements for accurate document classification. The paper zeroes in on the Term Frequency-Inverse Document Frequency (TF-IDF) method, a statistical approach based on word occurrences, and explores variations in the model approach. These variations include weighting on word occurrences, filtering processes on document words, rule creation on term documents, extraction for two or more syllables, and combinations with other extraction methods [5].

Anne Stockem Novo in her research tackles the challenge of detecting near-duplicate news articles on various websites, a task crucial for enhancing the efficiency of search engines and recommender systems. While straightforward for lexically identical documents, identifying semantically similar articles poses a greater challenge. The study focuses on leveraging named entity recognition, particularly entities like people, places, and organizations, to discern near-duplicates. Using a fine-tuned BERT model, the research achieves impressive performance measures exceeding 97%. Notably, the SHAP library provides insights into the model's decisions and highlights the importance of individual words in text documents [6].

Mamata Das and Selvakumar K conduct a comprehensive study on feature weighting methods for text classification, specifically focusing on Term Frequency-Inverse Document Frequency (TF-IDF) in the realm of Natural Language Processing (NLP). The research employs two key features, N-Grams and TF-IDF, applied to unstructured data from IMDB movie reviews and Amazon Alexa reviews datasets for sentiment analysis. The study employs state-of-the-art classifiers such as Support Vector Machine (SVM), Logistic Regression, Multinomial Naïve Bayes (Multinomial NB), Random Forest, Decision Tree, and k-nearest neighbors (KNN) to validate the proposed model. Notably, TF-IDF outperforms N-Gram, demonstrating a significant increase in accuracy (93.81%), precision (94.20%), recall (93.81%), and F1-score (91.99%) values, particularly excelling in the Random Forest classifier. [7].

Maarten Grootendorst introduces BERTopic, a novel topic model that advances traditional approaches by incorporating a class-based variation of Term Frequency-Inverse Document Frequency (TF-IDF). Utilizing pre-trained transformer-based language models, BERTopic generates document embeddings, clusters these embeddings, and refines topic representations through the class-based TF-IDF procedure. The result is a topic model that produces coherent topics and proves competitive across diverse benchmarks, encompassing classical models and contemporary clustering approaches in topic modeling [8].

Jiaen Guo introduces a novel solution for cross-modal vessel image retrieval in the domain of remote sensing data analysis. Acknowledging the lack of attention to the retrieval of target images like surface vessels, Guo addresses the challenges posed by complex geometric features and modality gaps. The proposed approach, Distillation-Based Hashing Transformer (DBHT), leverages a vision transformer (ViT) as the feature extractor for target images, integrating a hash token designed and attended to ViT for hashing generation. To overcome precision challenges in common feature space construction, a two-step feature learning strategy is employed, building a well-performing unimodal hashing retrieval framework first and then transferring hashing knowledge to another modality [9].

Manling Li introduces CLIP-Event, a novel approach within the domain of vision-language pretraining models. Unlike existing models that primarily focus on understanding objects or entities, CLIP-Event aims to bridge the gap at the level of events and their associated argument structures. Leveraging contrastive learning, the framework encourages pretraining models to comprehend events and participant roles by utilizing text information extraction technologies for event structural knowledge. Multiple prompt functions are employed to contrast difficult negative descriptions, and an event graph alignment loss based on optimal transport captures event argument structures. The approach is evaluated on a large event-rich dataset, demonstrating impressive results in argument extraction on Multimedia Event Extraction [10].

Mete Eminağaoğlu addresses the vital role of accurate and efficient textual data processing and document classification in knowledge management and related domains. Recognizing the impact of text mining, information retrieval, and document classification on various applications, Eminağaoğlu introduces a new similarity measure designed for use with well-known algorithms like k-nearest neighbors (k-NN) and Rocchio. This novel

measure is tested on structured textual datasets and compared against standard metrics like Cosine similarity, Euclidean distance, and Pearson correlation coefficient [11].

Rutger van der Spek researches for the fast estimation of Kendall's Tau and conditional Kendall's Tau matrices, crucial multivariate dependence measures in the analysis of random vectors. Recognizing the computational challenges of existing estimators, particularly in large dimensions, van der Spek proposes new estimators that significantly reduce computational costs while maintaining a comparable error level. The approach leverages structural assumptions on the underlying matrices, assuming block structures with constant values in off-diagonal blocks. This allows for more efficient averaging over pairwise estimates within each block. Explicit variance expressions highlight the improved efficiency of the estimators. The study covers both unconditional and conditional settings, demonstrating joint asymptotic normality and reduced asymptotic variance. [12].

Daniel Deutsch addresses a critical aspect of machine translation (MT) evaluation metrics, focusing on the challenges posed by ties in the context of meta-evaluation. With Kendall's τ often used for this purpose, the handling of ties becomes a nuanced and contentious issue, leading to various variants in the literature. Deutsch demonstrates weaknesses in existing tie-handling methods, suggesting potential gaming of the system in some instances. To address these concerns, he proposes a novel approach using a version of pairwise accuracy that acknowledges and credits metrics for accurately predicting ties. This is complemented by a tie calibration procedure that introduces ties into metric scores, allowing for a fair comparison between metrics that predict ties and those that do not [13].

3. Methodology

The methodology employed in this research endeavors to enhance cross-media semantic text similarity estimation within the dynamic landscape of digital content. By comparing and contrasting various weighted hashing algorithms, we delve into the intricacies of textual similarity across diverse media types, including both text and images. The journey unfolds with meticulous steps, from assembling a comprehensive dataset representative of real-world scenarios to deploying advanced algorithms like TF-IDF, machine learning models, Informational Semantic Hashing (ISH), Word Mover's Distance (WMD), Cosine Similarity, BERT embeddings, and Kendall's Tau. Each method contributes to the multifaceted exploration of text similarity, offering unique perspectives and insights. Our methodology not only encompasses traditional techniques but also embraces state-of-the-art approaches, catering to the evolving demands of a dynamic digital landscape. The ensuing sections detail the steps taken, highlighting the rationale behind each choice and showcasing the depth of our comparative analysis. Fig 1. illustrates the comprehensive flowchart encapsulating the sequential steps undertaken throughout the research. This visual representation provides a roadmap detailing the diverse stages involved in our investigation, guiding the progression from data collection to the implementation of various algorithms.

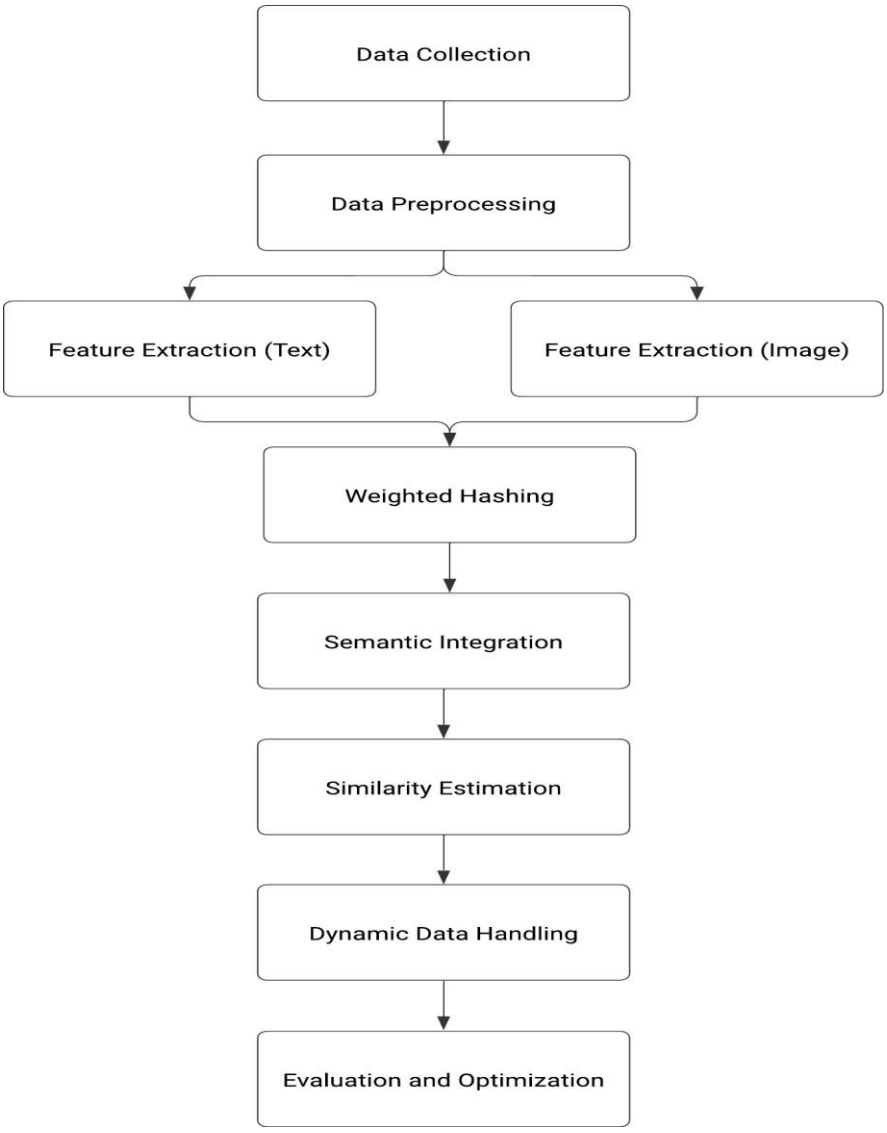

Fig 1. Flowchart

3.1. Data Collection and Preprocessing

3.1.1. Data Collection

In the initial phase of the research, data is collected from diverse online platforms, focusing on articles, blogs, and posts. Employing web scraping principles, a method involving the automated extraction of data from websites, curated a dataset representative of the dynamic content present on the internet. Furthermore, for specific algorithms such as logistic regression and random forest, integration of dataset from Quora in 2017 was carried. This dataset, comprising pairs of questions with a binary indicator of similarity, was generated collaboratively by Quora users who assessed the similarity of given questions. The dataset encompasses 40,291 entries across four columns: Sr no, Question 1, Question 2, and isSimilar. The research utilizes 19,200 rows for testing and 22,800 rows for training purposes, ensuring a robust evaluation of the selected algorithms.

Additionally, for methods like cosine similarity, a bespoke dataset was crafted. Leveraging web scraping techniques, we collected 52 user inputs to estimate similarity rankings, ranging from 1 to 10, for specific paragraphs. This dataset served as a valuable resource for evaluating

algorithmic performance using various statistical metrics. The ensuing sections detail the intricacies of data processing and the specific methodologies employed for each algorithm.

3.1.2. Data Preprocessing

In this phase, the obtained data underwent meticulous cleaning to enhance its quality and prepare it for subsequent analysis. The cleaning process involved the removal of extraneous elements such as links, punctuation, and stop words, contributing to a refined dataset. By eliminating irrelevant components, aimed to streamline the data for more effective and accurate processing in the subsequent stages.

Moreover, the data preprocessing incorporated principles of attention span, a technique centered around ranking words based on their significance. Employing mathematical methods, the attention span approach allowed to assign weights to words, emphasizing their importance in the subsequent stages of analysis. This careful curation and refinement of the dataset lay the foundation for robust and meaningful results to the research.

3.1.3. Tokenization and Vectorization

The subsequent step in the methodology involves two pivotal processes—tokenization and vectorization. Tokenization is the practice of segregating and compiling all the extracted words, essentially creating a comprehensive list. Following tokenization, the dataset is transformed into a machine-readable format through vectorization. Various algorithms, such as TF-IDF, doc2vec, word2vec, GloVe, Bag of Words, among others, are employed to accomplish this conversion.

3.2. Feature extraction.

Feature extraction is an important phase in the research methodology, aimed at identifying and isolating the most pertinent elements within the dataset. This process involves distilling relevant information from the raw data to construct robust and discriminative features essential for subsequent analysis. The efficacy of feature extraction techniques significantly impacts the accuracy and performance of similarity estimation models.

3.2.1. TF-IDF (Term Frequency-Inverse Document Frequency)

TF-IDF is a statistical measure used in information retrieval and text mining to evaluate the significance of a term in a document relative to a collection of documents (corpus). It calculates a weight representing the importance of a term in a document within a corpus. This technique is particularly useful in various natural language processing tasks, including text classification, document similarity, and information retrieval. TF measures the frequency of a term in a document. It signifies how often a term appears in a specific document relative to the total number of terms in that document. (1)

The formula to calculate TF.

TF(t, d) = (Number of times term t appears in document d) /(Total number of terms in document d) (1)

IDF determines the importance of a term across a corpus by penalizing terms that appear frequently across documents and emphasizing terms that are less common. (2) Formula to calculate IDF.

IDF(t, D) = log_e(Total number of documents in the corpus D / Number of documents containing term t) (2)

Hence, the TF-IDF score for a term in a specific document is computed by multiplying the TF and IDF values.(3)

TF-IDF(t, d, D) = TF(t, d) * IDF(t, D) (3)

3.2.2. Hashing

Textual data analysis often involves transforming unstructured text into structured, machine-understandable representations. Hashing algorithms is often used in this process by converting textual data into fixed-size hash codes or values. The methodology uses Locality Sensitive Hashing (LSH) algorithm which is integral in text analysis for approximate similarity searches, particularly in scenarios where comparing an extensive number of data points becomes impractical due to scale. LSH offers an efficient strategy for narrowing down searches by effectively reducing dimensions and identifying highly related data points. It operates by utilizing hash values to group similar input points into designated buckets. Unlike conventional hashing methods, LSH aims to maximize the chances of grouping similar points together into a single bucket, increasing the efficiency of similarity search operations

Following are the algorithmic steps involved in LSH:

- Generation of arbitrary hyperplanes traversing the unit sphere containing input points.
- Assignment of binary bit values to these hyperplanes based on the relative positioning of input points with respect to the hyperplane's orientation.
- Iterative repetition of the above step to create hashed vectors with reduced dimensionality.
- Calculation of the Hamming distance between a query vector and each index vector to identify the closest matches. Smaller Hamming distances signify higher similarities.

Computational complexity can be discussed with an example, suppose K hyperplanes are created, approximately equivalent to the natural logarithm of P (log P), with N-dimensional P points as inputs. In this scenario, the computational cost remains proportional to N x K, providing an estimation of the final bucket placement. The overall computational expense in LSH, operating at O(logP), signifies the expected collisions and comparisons for all dimensions, significantly enhancing the efficiency of similarity searches. Fig. 2 visualizes the comparison of LSH with other hashing functions.

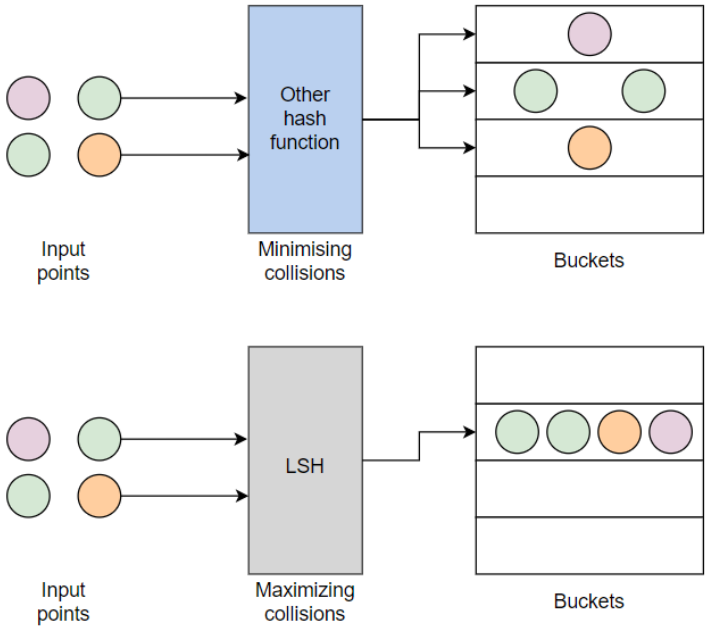

Fig 2. LSH's comparison with other hash functions

3.2.3. Word Embeddings

Word Embeddings, a popular technique in Natural Language Processing (NLP), represent words as vectors in a high-dimensional space. These word representations are capable of capturing semantic relationships between words based on their context and meaning within a given corpus. It encodes semantic meanings and relationships among words by mapping them into continuous vector spaces. It is constructed with algorithms Word2Vec and GloVe each with an approach to capturing semantic nuances. It can significantly reduce the dimensionality of text data while retaining semantic information.

3.3. Similarity Estimation Methods

Similarity Estimation Methods are integral in text analysis, allowing the evaluation of likeness or distinctions between textual elements. These methods gauge the similarity among documents, sentences, or words, facilitating diverse tasks in Natural Language Processing (NLP), such as information retrieval, document clustering, and recommendation systems.

3.3.1. Cosine similarity

Cosine similarity serves as a metric to quantify the likeness between two non-zero vectors in an inner product space, widely utilized across domains such as information retrieval, natural language processing, and machine learning. This measure computes the cosine of the angle formed between the vectors, generating a numerical value within the range of -1 to 1.

- A cosine similarity value of 1 denotes perfect similarity, implying the vectors align perfectly in the same direction.
- A value of 0 signifies orthogonality, indicating no similarity between the vectors.
- A score of -1 reflects perfect dissimilarity, signifying the vectors are in opposing directions.

Mathematically, the cosine similarity between two vectors A and B is computed as (4)

$$\text{Cosine Similarity}(A, B) = (A \cdot B) / (||A|| * ||B||) \qquad (4)$$

Where:

- A · B represents the dot product of vectors A and B.
- ||A|| and ||B|| denote the magnitudes (Euclidean norms) of vectors A and B, respectively.

In practical applications, cosine similarity finds extensive use in natural language processing tasks such as document similarity assessment, text categorization, and recommendation systems.

For instance, when dealing with textual data, documents can be represented as vectors using methods like Term Frequency-Inverse Document Frequency (TF-IDF) and word embeddings such as Word2Vec or GloVe. The cosine similarity metric is then applied to measure the similarity between these document vectors. This facilitates the retrieval of related documents, clustering similar texts, and making content-based recommendations. Refer to Fig. 3 for the ranking and evaluation following the utilization of this algorithm.

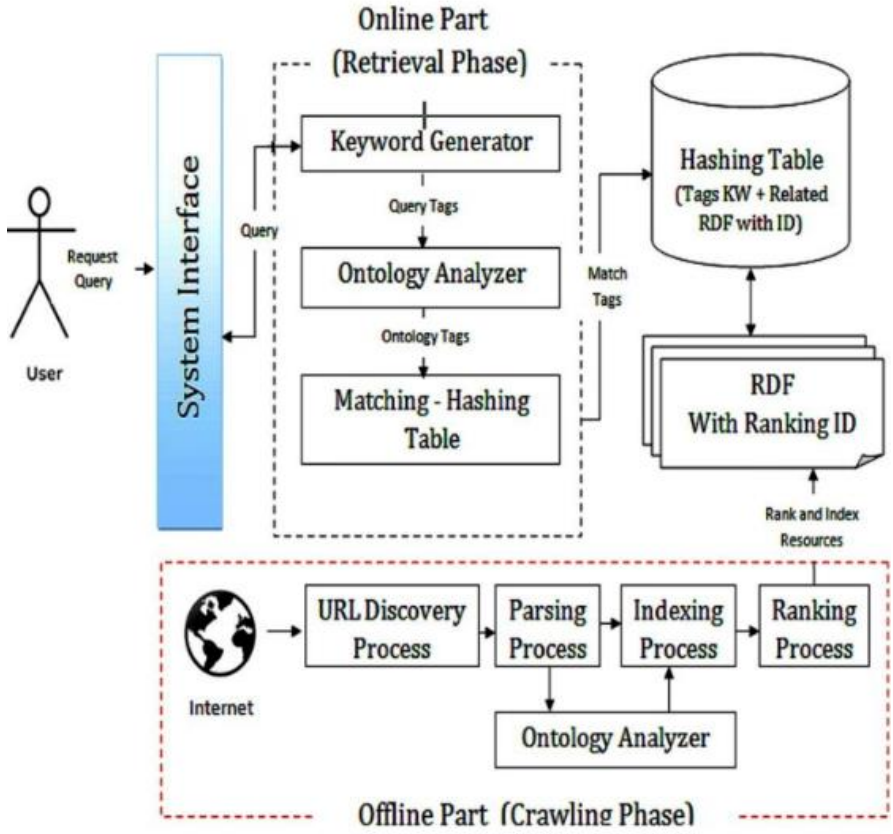

Fig 3. Evaluation

3.3.2. Word Mover's Distance

Word Mover's Distance (WMD) capitalizes on recent advancements in word embeddings, leveraging innovative techniques like word2vec and GloVe to construct meaningful word representations based on localized word co-occurrences within sentences.

Drawing upon the capabilities of sophisticated embedding models, such as word2vec and GloVe, WMD benefits from high-quality word embeddings capable of effectively handling extensive datasets. These embedding methodologies demonstrate that word vectors possess semantic relationships, observable through mathematical operations on the vectors. For instance, vector arithmetic involving word embeddings can reflect semantic relations;

subtracting the "Germany" vector from "Berlin" and adding the "France" vector results in a vector close to "Paris."

WMD conceptualizes textual content as weighted point clouds composed of embedded words, aiming to exploit the semantic characteristics of word vector embeddings. By determining the minimum cumulative distance that words from one text document must traverse to align with those from another document's point cloud, WMD calculates the distance between the two documents, A and B. Refer to the accompanying visual depiction in Fig 4.

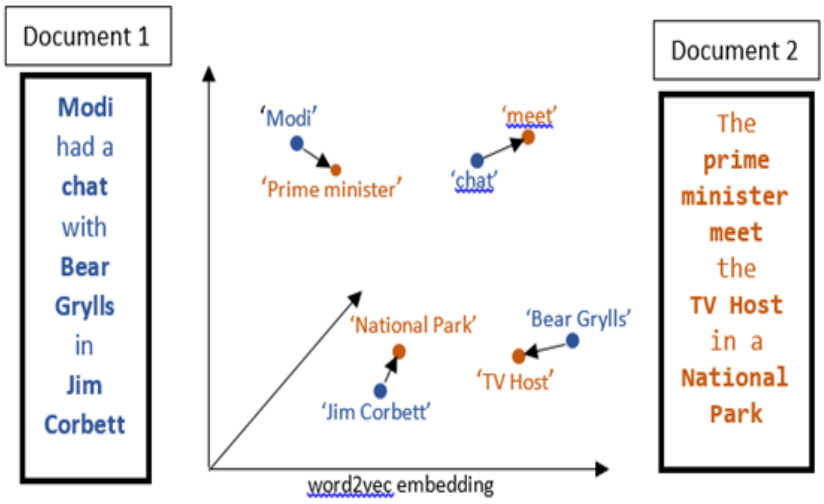

Fig 4. Word Mover's Distance

In contrast to methods focusing solely on syntactic or semantic word embeddings, WMD seamlessly integrates both syntactic and semantic dimensions to gauge similarity across text documents. This approach measures the least distance that embedded words from one text must traverse to align with embedded words from another. It's worth noting that WMD considers this distance measure as a specific case of the Earth Mover's Distance—a well-explored concept in transportation theory with established solutions.

3.3.3. BERT

BERT, short for Bidirectional Encoder Representations from Transformers, is Google's prominent language representation model known for its versatility in natural language processing (NLP) applications. It operates through two key stages: pre-training and fine-tuning enabling its adaptation to diverse downstream tasks, Fig 5.

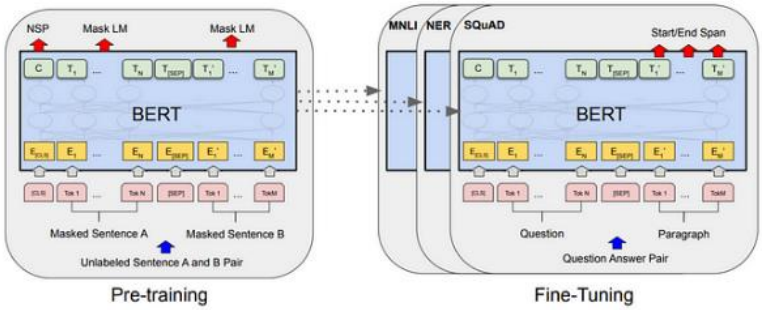

Fig 5. BERT consists of two steps

BERT's pre-training involves two unsupervised tasks:

- Masked Language Model (MLM): Here, a fraction of input tokens is randomly masked, and the model is trained to predict these masked tokens. Approximately 15% of words are masked, comprising 80% as [MASK], 10% random, and 10% unchanged tokens.
- Next Sentence Prediction (NSP): NSP aids BERT in understanding sentence relationships. It learns from pairs of sentences, where 50% correspond to consecutive sentences, and the rest are random pairings.

These pre-training tasks equip BERT with comprehensive bidirectional language understanding, making it adaptable for various NLP applications without necessitating task-specific labeled data during pre-training.

Fine-tuning BERT involves adding task-specific layers while retaining the core BERT model. This approach minimizes parameter retraining, ensuring a swift and cost-effective adaptation process. For tasks like Sentence Pair Classification and Single Sentence Classification, additional layers are appended, using the [CLS] token output. Fig 6.

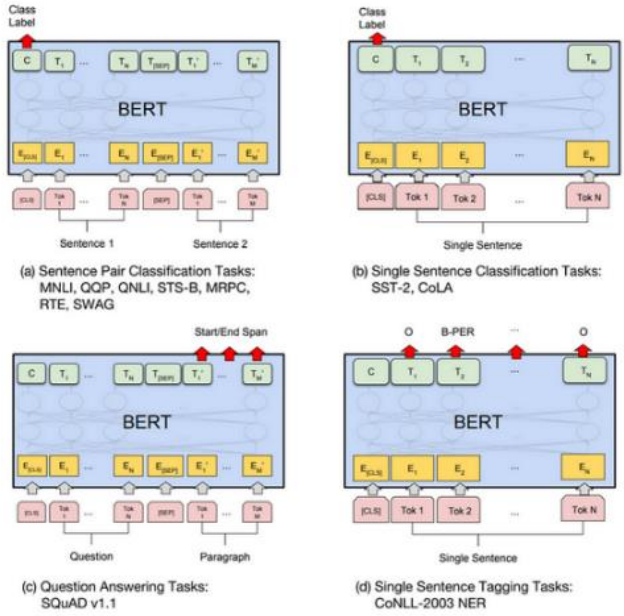

Fig 6. Fine-tuning BERT for various downstream tasks

In Question-Answering tasks, the model incorporates start and end vectors during fine-tuning, facilitating answers' span identification within paragraphs. The method selects the span with the highest score as the prediction.

3.3.4. Kendall's Tau

Kendall's Tau, a measure of correlation or concordance between rankings, serves as an essential tool in assessing the similarity of ordered data or rankings. In the context of text similarity estimation, Kendall's Tau is often utilized to quantify the agreement or discrepancy between different methods or human annotations when ranking text elements by their similarity.

It evaluates the consistency of rankings by comparing the number of concordant and discordant pairs within two rankings. A concordant pair exists when the relative order of

elements in two rankings is consistent, while a discordant pair represents an inversion in their orders. The value of Kendall's Tau ranges from -1 to 1, where:

- 1 signifies perfect agreement or correlation between rankings.
- 0 indicates no correlation or independence between rankings.
- -1 represents perfect disagreement or inverse correlation between rankings.

In the context of text similarity estimation, Kendall's Tau is employed to compare and evaluate the alignment of similarity rankings generated by different methods or human assessments. It helps in understanding the consistency or discrepancies in the rankings produced by various similarity estimation techniques, aiding researchers in selecting the most reliable method for assessing text similarity.

3.4. Machine Learning models made using TF-IDF and Performace Evaluation

Upon applying TF-IDF vectorization to textual data, diverse machine learning models were employed to predict text similarity by training them with the transformed data. Notably, the models utilized in this research include Logistic Regression, Multinomial Naive Bayes, Random Forest Classifier, and Decision Tree Classifier. An intriguing observation surfaced during this investigation: the performance of these models exhibited variation contingent on the specific vectorization method employed. This variance in performance highlights the influence of the vectorization technique on the predictive capabilities of the models in assessing text similarity.

The performance assessment of the machine learning models, including LSH and BERT-based models, involved a comprehensive evaluation encompassing metrics such as accuracy, precision, recall, and F1-scores. Additionally, the evaluation of Cosine Similarity and Word Mover's Distance comprised the analysis of statistical indicators, namely Pearson’s Coefficient, Spearman’s Coefficient, Coefficient of Concordance, and Kendall’s Tau. These metrics and coefficients provided a robust framework to gauge and compare the effectiveness of the various models and similarity estimation methods employed in the research.

4. Results

The results obtained from the comprehensive evaluation of various machine learning models and similarity estimation methods in this study reveal noteworthy insights into text similarity estimation techniques. Through a meticulous examination of performance metrics and statistical indicators, the effectiveness and suitability of different approaches in capturing textual resemblance and dissimilarity have been analyzed. This section presents a detailed overview and critical analysis of the outcomes obtained from the experimental evaluations, shedding light on the comparative performance of diverse methods across a spectrum of text similarity estimation tasks.

4.1. Models Performance and Confusion Matrices

The subsequent Table 1. showcases the performance metrics attained by the ML models utilized in this study.

Table 1. Performance table of the models

Name	Accuracy	Precision	Recall	F1-Score
Logistic Regression	0.712	0.687	0.783	0.69
Multinomial Naïve Bayes	0.748	0.79	0.68	0.73
Random Forest	0.783	0.81	0.73	0.77
Decision Tree	0.716	0.72	0.70	0.71

The table exhibits a comprehensive summary of the performance metrics for various machine learning models used in our study. Each model, including Logistic Regression, Multinomial Naïve Bayes, Random Forest, and Decision Tree, was evaluated based on essential metrics such as accuracy, precision, recall, and F1-Score. These metrics are crucial in assessing the models' predictive abilities and how well they classify or predict similarity among textual elements.

The confusion matrices for ML models are given in the fig. 7, fig 8, fig 9, fig 10.

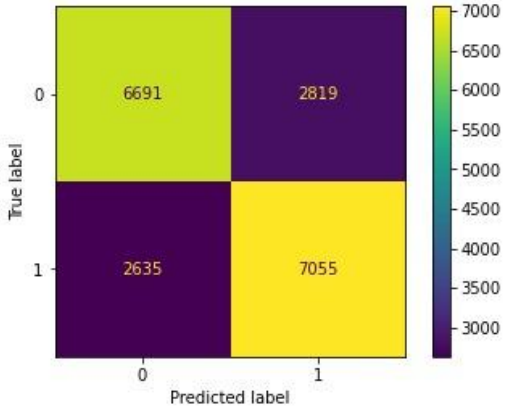

Fig 7. Confusion matrix for decision tree

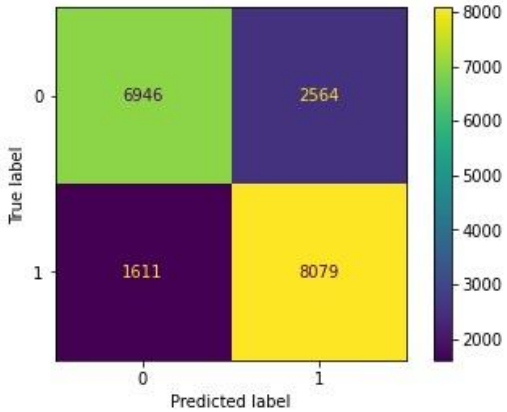

Fig 8. Confusion matrix for random forest

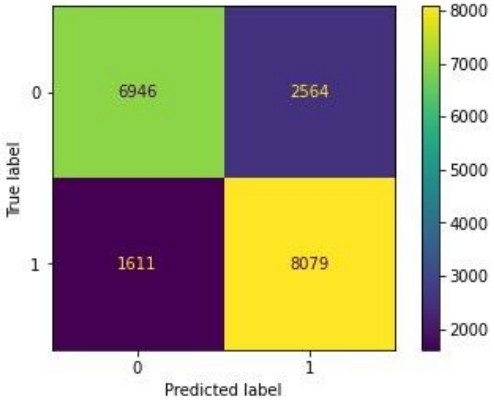

Fig 9. Confusion matrix for Multinomial Naïve bayes

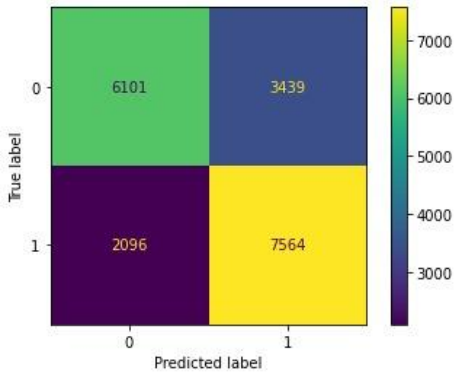

Fig 10. Confusion matrix for logistic regression

These performance metrics were acquired following TF-IDF vectorization. Among the methods employed, the results indicate that the Random Forest model exhibited the highest performance across accuracy, precision, recall, and F1-Score. The slightly lower-than-expected accuracy values could be attributed to the considerable size of the testing dataset, influencing the overall performance of the models.

4.2. Cosine similarity and Word mover's distance

Below Table 2. are the metrics obtained from Word Mover's Distance, Cosine Similarity using TF-IDF, and Cosine Similarity using Word2Vec embeddings.

Table 2. Performance table

Name	Pearson's Coefficient	Spearman's rho	Coefficient of Concordance	Kendall's tau
Word Mover's distance	-0.4229	-0.428	-0.016	-0.33
Cosine similarity with TF-IDF	0.5208	0.200	0.010	0.067
Cosine similarity with Word2Vec	0.3216	0.2	0.004	0.067

Upon analysis, it is evident that Word Mover's Distance demonstrates the least performance among the methods evaluated. On the contrary, Cosine Similarity using TF-IDF displays superior performance. This discrepancy can be attributed to the consideration of contextual information by Cosine Similarity, even when the textual distance is considerable, compared to the sensitivity of Word Mover's Distance towards larger textual differences.

4.3. BERT Based Model

The graph in Figure 11 illustrates the performance of a basic BERT model featuring two attention masks and two global average pooling layers. This model exhibits commendable accuracy at 88.7%, along with a recall of 88%, an F1-Score of 89%, and a precision of 92%.

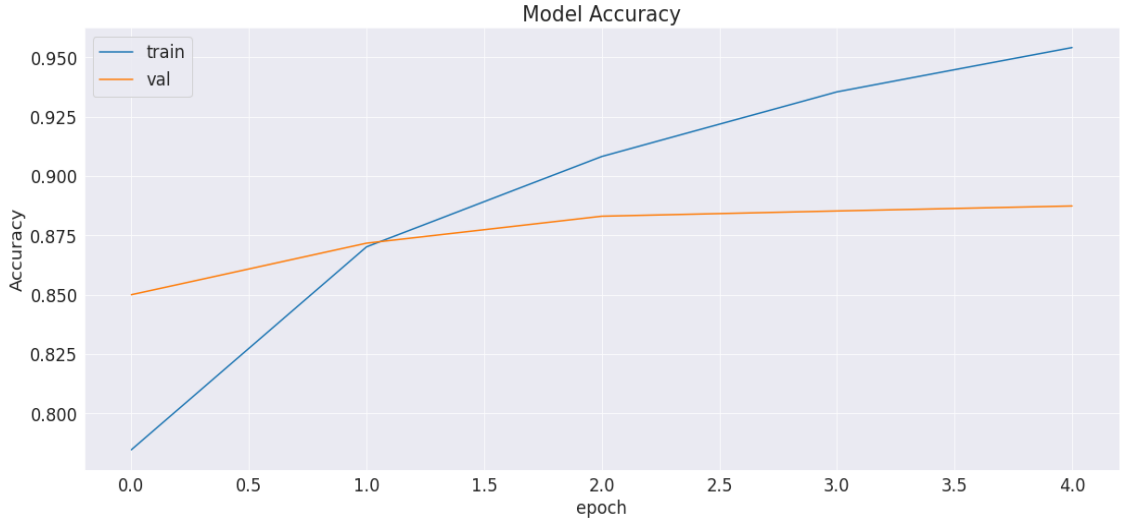

Fig 11. Training and validation accuracy

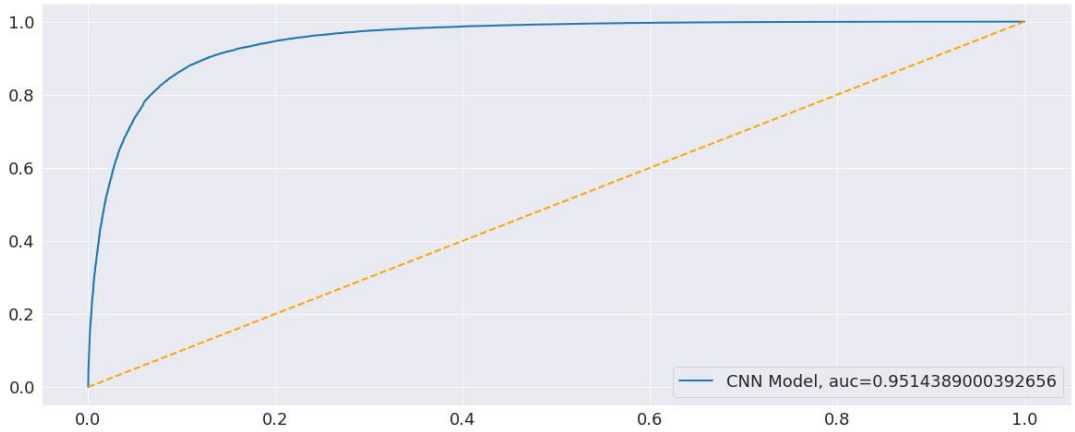

Fig 12. AUC of BERT

The AUC score of 95% gives a very clear idea of the great performance of the model.

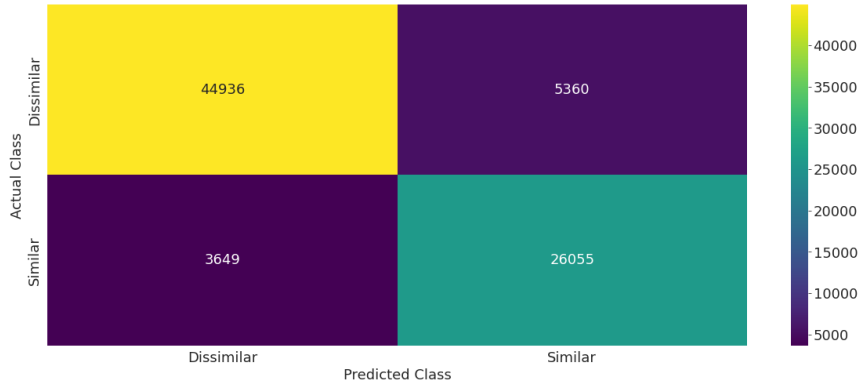

Fig 13. Confusion matrix for BERT

4.4. Locality sensitive hashing performance

Evaluating the accuracy of MinHash LSH in this study didn't yield significant insights. The focus was primarily on assessing precision and recall concerning set thresholds, which were established at 0.2, 0.4, 0.6, and 0.8.

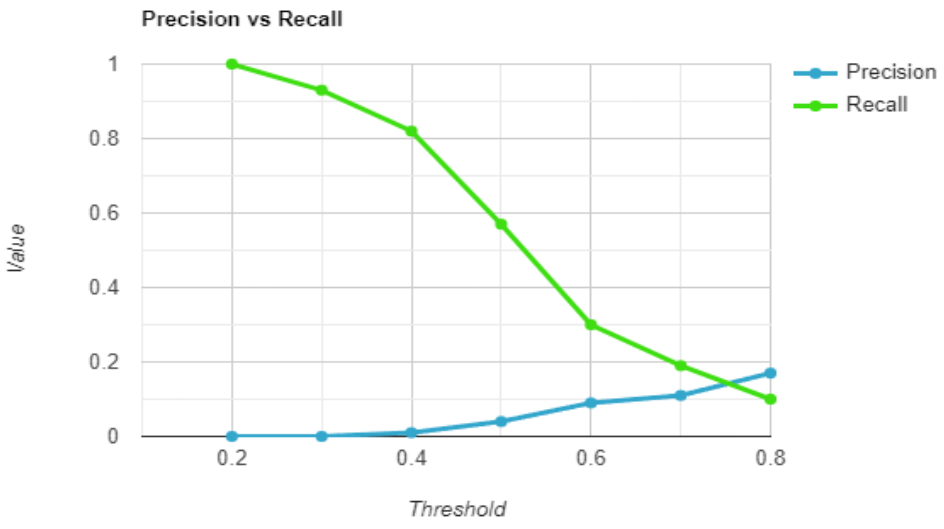

Fig 14. Graph for precision and recall

These observations reveal the anticipated trade-off between precision and recall. Lower thresholds tend to identify more potential duplicate questions but also introduce non-duplicate pairs, increasing the occurrence of false positives. Conversely, higher thresholds diminish false duplicate pairs but may overlook true duplicate pairs, leading to more false negatives. This delicate balance between precision and recall underscores the complexity of the algorithm's performance.

References

[1] *F. Lan, "Research on Text Similarity Measurement Hybrid Algorithm with Term Semantic Information and TF-IDF Method," Advances in Multimedia, vol. 2022, pp. 1-11, **(2022)**, doi: 10.1155/2022/7923262.*

[2] *H. Liu, X. Chen, and X. Liu, "A Study of the Application of Weight Distributing Method Combining Sentiment Dictionary and TF-IDF for Text Sentiment Analysis," IEEE Access, vol. 10, pp. 32280-32289, **(2022)**, doi: 10.1109/ACCESS.2022.3160172.*

[3] *L. Gomes, R. Da Silva Torres, and M. L. Côrtes, "BERT- and TF-IDF-based feature extraction for long-lived bug prediction in FLOSS: A comparative study," Information and Software Technology, vol. 160, p. 107217, **(2023)**, doi: 10.1016/j.infsof.2023.107217.*

[4] *G. Yunanda, D. Nurjanah, and S. Meliana, "Recommendation System from Microsoft News Data using TF-IDF and Cosine Similarity Methods," bits, vol. 4, no. 1, pp. 277-284, Jun. **(2022)**.*

[5] *Y. Setiawan, D. Gunawan, and R. Efendi, "Feature Extraction TF-IDF to Perform Cyberbullying Text Classification: A Literature Review and Future Research Direction," in Proc. 2022 International Conference on Information Technology Systems and Innovation (ICITSI), Bandung, Indonesia, **(2022)**, pp. 283-288, doi: 10.1109/ICITSI56531.2022.9970942.*

[6] *A. S. Novo and F. Gedikli, "Explaining BERT model decisions for near-duplicate news article detection based on named entity recognition," in Proc. 2023 IEEE 17th International Conference on Semantic Computing (ICSC), Laguna Hills, CA, USA, **(2023)**, pp. 278-281, doi: 10.1109/ICSC56153.2023.00054.*

[7] *M. Das and P. J. Alphonse, "A Comparative Study on TF-IDF feature Weighting Method and its Analysis using Unstructured Dataset," ArXiv preprint arXiv:2308.04037, **(2023)**.*

[8] *M. Grootendorst, "BERTopic: Neural topic modeling with a class-based TF-IDF procedure," ArXiv preprint arXiv:2203.05794, **(2022)**.*

[9] *J. Guo, X. Guan, Y. Liu, and Y. Lu, "Distillation-Based Hashing Transformer for Cross-Modal Vessel Image Retrieval," IEEE Geoscience and Remote Sensing Letters, vol. 20, pp. 1-5, **(2023)**, Art no. 8500605, doi: 10.1109/LGRS.2023.3294393.*

[10] *M. Li et al., "CLIP-Event: Connecting Text and Images with Event Structures," ArXiv preprint arXiv:2201.05078, **(2022)**.*

[11] *Mete Eminağaoğlu and Yılmaz Gökşen, "A New Similarity Measure for Document Classification and Text Mining," KnE Social Sciences, vol. 4, no. 1, **(2020)**, doi: 10.18502/kss.v4i1.5999.*

[12] *A. Derumigny, "Fast estimation of Kendall's Tau and conditional Kendall's Tau matrices under structural assumptions," ArXiv preprint arXiv:2204.03285, **(2022)**.*

[13] *D. Deutsch, G. Foster, and M. Freitag, "Ties Matter: Meta-Evaluating Modern Metrics with Pairwise Accuracy and Tie Calibration," ArXiv preprint arXiv:2305.14324, **(2023)***