Published online on the journal's webpage: **http://journal.ittelkom-pwt.ac.id/index.php/dinda**

Journal of Dinda

Data Science, Information Technology, and Data Analytics

Vol. 4 No. 2 (2024) 149 - 153 E-ISSN: 2809-8064

Document Similarity using Term Frequency-Inverse Document Frequency Representation and Cosine Similarity

Adi Widianto[1*], Eka Pebriyanto[2], Fitriyanti[3], Marna[4]

[1*,4]Data Science, Faculty of Science and Informatic, Universitas Pertiba

[2]Computer Systems Engineering, Faculty of Science and Informatic, Universitas Pertiba

[3]Information Systems and Technology, Faculty of Science and Informatic, Universitas Pertiba

[1*]adiwidianto@pertiba.ac.id, [2]ekapebri@pertiba.ac.id, [3]fitriyanti@pertiba.ac.id, [4]marna@pertiba.ac.id

Abstract

Document similarity is a fundamental task in natural language processing and information retrieval, with applications ranging from plagiarism detection to recommendation systems. However, the current method is not suitable with long documents. The need to calculate the semantic meaning, but robust enough to be implemented in long document become the problem. In this study, we leverage the term frequency-inverse document frequency (TF-IDF) to represent documents in a high-dimensional vector space, capturing their unique content while mitigating the influence of common terms. Subsequently, we employ the cosine similarity metric to measure the similarity between pairs of documents, which assesses the angle between their respective TF-IDF vectors. To evaluate the effectiveness of our approach, we conducted experiments on the Document Similarity Triplets Dataset, a benchmark dataset specifically designed for assessing document similarity techniques. Our experimental results demonstrate a significant performance with an accuracy score of 93.6% using bigram-only representation. However, we observed instances where false predictions occurred due to paired documents having similar terms but differing semantics, revealing a weakness in the TF-IDF approach. To address this limitation, future research could focus on augmenting document representations with semantic features. Incorporating semantic information, such as word embeddings or contextual embeddings, could enhance the model's ability to capture nuanced semantic relationships between documents, thereby improving accuracy in scenarios where term overlap does not adequately signify similarity.

Keywords: *Document Similarity, TF-IDF, Cosine Similarity*

1. Introduction

Document similarity serves as a fundamental cornerstone in numerous applications within natural language processing and information retrieval. Its significance spans a wide range of applications, from aiding in plagiarism detection to facilitating recommendation systems, as highlighted in [1]. The ability to accurately quantify the similarity between documents enables a deeper understanding of their content and relevance, thereby enhancing the efficiency and effectiveness of various computational tasks. This capability is crucial for improving search engine results, organizing large collections of documents, and providing personalized content recommendations. Numerous studies have explored and employed various methods to perform document similarity tasks, leveraging techniques such as term frequency-inverse document frequency (TF-IDF), word embeddings, and deep learning models [1], [2], [3], [4]. These methods have been rigorously tested and refined to improve the accuracy and scalability of document similarity measures, making them integral to the development of advanced natural language processing systems.

In document similarity tasks, there are some document representations that are already used. These representations are word-based like bag of word, Latent Dirichlet Allocation (LDA), and paragraph vectors [4]. Dai et al. in [4] used paragraph vectors as representation to document similarity based on clustering of Wikipedia English articles. Then, the document distance is calculated by cosine similarity. The proposed approach tested on hand-built triplets of Wikipedia articles dataset. The dataset consists of 172 triplets URL of English Wikipedia article, or 516 documents in total. The test gives the result prediction accuracy 93% for paragraph vector, LDA 82%, and bag of words 86%.

Received: 01-08-2024 | Accepted: 11-08-2024 | Published: 12-08-2024

However, this method is not suitable with long documents. The need to calculate the semantic meaning, but fast enough to be implemented in long document become the problem for method like paragraph vector.

Our focus is on utilizing the Term Frequency-Inverse Document Frequency (TF-IDF) approach, a widely recognized technique for document representation in the field of information retrieval [5], [6], [7], [8], [9]. The TF-IDF method transforms textual data into high-dimensional vector representations, where each term's frequency is adjusted based on its importance across the entire corpus. This approach effectively weighs the relevance of terms by diminishing the impact of common words that may not significantly contribute to the document's distinctiveness. Such a method is particularly well-suited for comparing long documents due to its computational efficiency and ability to handle large-scale text data with relatively low resource requirements.

Following TF-IDF vectorization, we employ the cosine similarity metric to measure the similarity between pairs of documents [10], [11], [12], [13], [14], [15]. Cosine similarity calculates the cosine of the angle between the TF-IDF vectors of two documents, offering a robust measure of similarity that remains consistent regardless of variations in document length and term frequency distributions. By evaluating the angle between these vectors, cosine similarity quantifies the degree of similarity between documents, which is crucial for various tasks such as document clustering, retrieval, and categorization. This metric helps ensure that similarity assessments are based on the content and contextual relevance of the documents rather than their length or overall term frequency.

Our primary objective is to assess the performance of a document similarity model that integrates TF-IDF representation with the cosine similarity metric. Through this evaluation, we aim to determine the effectiveness of this approach in accurately quantifying document similarity and to identify potential areas for improvement in existing methodologies.

The structure of this study is as follows: Section 2 details the research methods employed, Section 3 discusses the results and their implications, and Section 4 presents the conclusions drawn from the research.

2. Research Methods

The dataset used to evaluate our proposed method comprises hand-built triplets of Wikipedia articles, as created by Dai et al. in [4]. This dataset includes triplets of document URLs from English Wikipedia articles, each labeled with one of 27 categories, such as machine learning, books, places, animals, and others. Certain categories, such as "places," require a more nuanced understanding of the document content due to their inherent complexity and the need for contextual knowledge.

In this dataset, each triplet consists of three document URLs. The task is to predict whether the first document URL is more similar to the second document URL than to the third document URL. In the context of cosine similarity, this means that the prediction is considered correct if the cosine similarity score between the first and second documents is higher (indicating closer similarity) compared to the score between the second and third documents. This approach allows for an evaluation of the effectiveness of the similarity measure in distinguishing between closely related and less related documents within the provided categories.

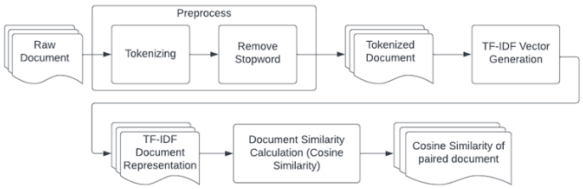

Figure 1. The Architecture of Implemented System

In Figure 1, the implemented system is composed of several key components: preprocessing, TF-IDF vector generation, and document similarity calculation. The outcome is determined by predicting whether the first document is more like the second or third document in each triplet.

2.1 *Preprocess*ing

Preprocessing steps are crucial for cleaning and reducing noise in the raw documents. The dataset is already cleaned with removed punctuation and unnecessary symbol. Preprocessing involves tokenizing the text and removing stop words, processes that are carried out using the Python package NLTK (Natural Language Toolkit). Tokenization breaks the text into individual words or tokens, while stop word removal eliminates common words that are unlikely to contribute significant meaning to the analysis, such as "and" "the," and "is."

2.2 *TF-IDF*

Once the text is preprocessed, TF-IDF vectors are generated using the Python package scikit-learn. The Term Frequency-Inverse Document Frequency (TF-IDF) vectorization transforms the cleaned text into numerical representations that reflect the importance of each term within a document relative to its frequency in the entire corpus. In this study, experiments are conducted using three different n-gram configurations: only unigrams, unigrams combined with bigrams, and

only bigrams. Unigrams consider single words, while bigrams account for pairs of adjacent words, providing more contextual information.

Term Frequency (TF) measures how frequently a term occurs within a document. For bi-grams, this involves counting the frequency of a pair of consecutive words. For instance, in the phrase "machine learning is fun," the bi-gram "machine learning" and "learning is" are considered.

Inverse Document Frequency (IDF) gauges the importance of a term by assessing its distribution across a corpus. Terms appearing in many documents are considered less informative. For bi-grams, IDF evaluates the rarity of a bi-gram across the document set. For example, if "machine learning" appears in many documents, it will have a lower IDF value compared to a bi-gram like "quantum computing," which might be less common.

2.3 *Cosine Similarity*

The next step involves calculating document similarity using cosine similarity, which is also implemented with scikit-learn. Cosine similarity measures the cosine of the angle between two non-zero vectors, providing a metric that reflects how similar two documents are based ontheir vector representations. For each triplet of documents, the system calculates the similarity between the first and the second document, as well as between the first and the third document.

The results of these similarity calculations are then compared to the work of Dai et al. as referenced in [4]. The accuracy of the system is quantified as the percentage of correctly predicted document triplets, where the system successfully identifies which of the second or third documents is more similar to the first document. This comparative analysis provides insights into the effectiveness of the implemented methods and configurations in accurately assessing document similarity.

3. Results and Discussion

This study aims to identify the most effective TF-IDF n-gram representation for the document similarity task. To achieve this, we utilize a test dataset consisting of 172 document triplets. Each triplet contains three documents, which can be paired and compared to assess their similarity. By systematically evaluating these triplets, we seek to determine which n-gram configurations—whether unigram, bigram, or combinations thereof—provide the most accurate and meaningful representations for assessing document similarity. This evaluation will help in understanding the impact of different n-gram choices on the performance of the TF-IDF model in capturing and quantifying document similarity.

Table 1. Experiment using different range of n-gram

N-Gram (range)	Accuracy
Unigram (1,1)	89.5%
Unigram and Bigram (1,2)	91.9%
Bigram (2,2)	**93.6%**

The experiment using different ranges of n-gram gave mixed results. The proposed system using bigram-only gives the best result with accuracy of 93.6%. Bigrams are proven more effective than unigrams in text representation because they capture context and relationships between adjacent words, providing more meaningful and nuanced information, especially in document similarity tasks.

Table 2. The Best Predicted Article

document-to-document Comparison	Cosine Similarity
Serena_Williams - Venus_Williams	0.598
Serena_Williams - Novak_Djokovic	0.333

The best predicted data, or with biggest cosine similarity, is article *Serena_Williams*, *Venus_Williams*, and *Novak_Djokovic*, which all is in the same topic of professional tennis Player. The dataset true label is article *Serena_Williams* is more similar to *Venus_Williams* than *Novak_Djokovic*. The model successfully predicted the long document of Wikipedia article similarity with the same topic.

Table 3. Comparison of proposed model to other method

Model	Accuracy
LDA [4]	82.0%
Averaged word embedding [4]	84.9%
Bag of words [4]	86.0%
Paragraph Vector [4]	93.0%
TF-IDF Bigram (Our Proposed Method)	**93.6%**

The proposed method demonstrates superior performance compared to other word-based representation models when applied to the same dataset. Specifically, the robust TF-IDF representation proves to be more effective with long documents, as evidenced by the dataset results.

However, false predictions can occur when documents share similar topics and terms, which may lead to ambiguities in similarity assessments. In such cases, a purely term-based approach may fall short, necessitating a semantic approach to better differentiate between document pairs. The accuracy of similarity determination hinges on the semantic context used for differentiation.

For instance, consider a triplet comparison involving the articles "Java," "C++," and "C." In the context of programming paradigms, "Java" and "C++" are more closely related, which aligns with the dataset's classification of this pair as a true prediction. However, when examining the historical development and syntax similarities, "C++" and "C" exhibit a closer resemblance. This example illustrates the importance of context in evaluating document similarity and highlights the need for integrating semantic understanding into similarity assessments to address potential discrepancies.

4. Conclusion

In this study, we developed a document similarity model for English texts using a TF-IDF representation combined with the cosine similarity metric. The model was evaluated using hand-built triplets of Wikipedia articles provided by Dai et al. The proposed method, which utilizes a TF-IDF representation with only bigrams, achieved a notable accuracy score of 93.6%. This performance is competitive with other word-based representation models, indicating the effectiveness of the approach.

However, there is still room for improvement. Future research should explore ways to enhance the model further, such as incorporating semantic-based approaches to capture the semantic context of documents more accurately. By integrating semantic analysis, the model could potentially improve its ability to discern nuanced similarities between documents, leading to even more accurate similarity assessments.

References

[1] N. Gahman and V. Elangovan, "A Comparison of Document Similarity Algorithms," *arXiv preprint arXiv:2304.01330*, 2023

[2] R. S. de Oliveira and E. G. S. Nascimento, "Analysing similarities between legal court documents using natural language processing approaches based on Transformers," *arXiv preprint arXiv:2204.07182*, 2022.

[3] M. Ostendorff, T. Ruas, T. Blume, B. Gipp, and G. Rehm, "Aspect-based document similarity for research papers," *arXiv preprint arXiv:2010.06395*, 2020.

[4] A. M. Dai, C. Olah, and Q. V. Le, "Document embedding with paragraph vectors," *arXiv preprint arXiv:1507.07998*, 2015.

[5] M. Umadevi, "Document comparison based on tf-idf metric," *International Research Journal of Engineering and Technology (IRJET)*, vol. 7, no. 02, pp. 1546-1550, 2020.

[6] P. M. Hasugian, J. Manurung, L. Logaraz, and U. Ram, "Implementation of Tf-Idf and cosine similarity algorithms for classification of documents based on abstract scientific journals," *Infokum*, vol. 9, no. 2, pp. 518-526, Jun. 2021.

[7] A. A. Huda, R. Fajarudin, and A. Hadinegoro, "Sistem Rekomendasi Content-Based Filtering Menggunakan TF-IDF Vector Similarity Untuk Rekomendasi Artikel Berita," *Building of Informatics, Technology and Science (BITS)*, vol. 4, no. 3, pp. 1679-1686, 2022.

[8] R. Saputra, M. G. Pradana, et al., "Implementasi Algoritma Cosine Similarity dan TF-IDF dalam Menentukan Rumpun Jabatan," *Krea-TIF: Jurnal Teknik Informatika*, vol. 12, no. 1, pp. 1-11, 2024.

[9] J. Joni and J. Halim, "Implementasi Metode Cosine Similarity Dan Tf-Idf Dalam Klasifikasi Pengaduan Masyarakat," *Jurnal Ilmiah Core IT: Community Research Information Technology*, vol. 10, no. 4, 2022.

[10] A. Kurnianti, P. Pahlevi, and I. Mufidah, "Recommendation System for Prospective Bride and Groom Using Cosine Similarity Algorithm," *Emerging Information Science and Technology*, vol. 4, no. 1, pp. 8-15, 2023.

[11] D. Marta, G. L. Ginting, and A. M. Hatuaon Sihite, "Deteksi Berita Palsu Tentang Vaksinasi Covid-19 Dengan Menggunakan Text Mining Dan Algoritma Cosine Similarity," *KOMIK (Konferensi Nasional Teknologi Informasi dan Komputer)*, vol. 6, no. 1, pp. 129-139, 2023.

[12] Y. Liu, Y. Zeng, R. Li, X. Zhu, Y. Zhang, W. Li, T. Li, D. Zhu, and G. Hu, "A Random Particle Swarm Optimization Based on Cosine Similarity for Global Optimization and Classification Problems," *Biomimetics*, vol. 9, no. 4, p. 204, 2024, MDPI.

[13] K. Park, J. S. Hong, and W. Kim, "A methodology combining cosine similarity with classifier for text

classification," *Applied Artificial Intelligence*, vol. 34, no. 5, pp. 396-411, 2020.

[14] M. R. Hasan and J. Ferdous, "Dominance of AI and machine learning techniques in hybrid movie recommendation system applying text-to-number conversion and cosine similarity approaches," *Journal of Computer Science and Technology Studies*, vol. 6, no. 1, pp. 94-102, 2024..

[15] M. Jain and H. Rastogi, "Automatic text summarization using soft-cosine similarity and centrality measures," in *4th International Conference on Electronics, Communication and Aerospace Technology (ICECA)*, 2020, pp. 1021-1028.