International Journal of Innovative Research and Scientific Studies, 8(4) 2025, pages: 2195-2204

IJIRSS
International Journal of Innovative Research & Scientific Studies

ISSN: 2617-6548

URL: www.ijirss.com

Document analysis via combined vectorization and machine learning approaches

Dinara Kaibassova[1], Bigul Mukhametzhanova[2], Dinara Tokseit[3], Aigul Kubegenova[4], Murad Kozhanov[2*]

[1]*Astana IT University, Astana, 010000, Kazakhstan.*
[2]*Abylkas Saginov Karaganda Technical University, Karaganda, 100000, Kazakhstan.*
[3]*L. N. Gumilyov Eurasian National University, Astana, 010000, Kazakhstan.*
[4]*West Kazakhstan Agrarian and Technical University named after Zhangir Khan, Uralsk, 090000, Kazakhstan.*

Corresponding author: Murad Kozhanov (*Email: mukhamedzhanova.bigul@mail.ru*)

Abstract

The purpose of this study is to develop an effective hybrid model for automatic document classification by combining statistical and semantic text vectorization techniques with machine learning algorithms. The methodology integrates Term Frequency–Inverse Document Frequency (TF-IDF) and Word2Vec embeddings with classifiers such as Support Vector Machine (SVM) and Random Forest. The proposed approach includes data preprocessing (tokenization, normalization, stop word removal, and lemmatization), feature extraction, model training, and evaluation using classification metrics such as accuracy, F1-score, Matthews Correlation Coefficient (MCC), and Cohen's Kappa. Experimental results demonstrate that the Word2Vec + SVM model outperforms other configurations, achieving 90.2% accuracy and an F1-score of 82.52%, thus highlighting the advantage of incorporating semantic context into vector representation. The study concludes that hybrid methods combining TF-IDF and Word2Vec with robust classifiers improve both the precision and generalizability of document analysis models. Practical implications include potential applications in sentiment analysis, topic modeling, text classification for legal and healthcare domains, and multilingual contexts. This research provides a foundation for developing high-performance text analysis systems applicable to various real-world natural language processing tasks.

Keywords: Automatic document analysis, Contextual word embedding, Machine learning, Natural language processing, Semantic matching.

DOI: 10.53894/ijirss.v8i4.8356
Funding: This work is supported by the Science Committee of the Ministry of Science and Higher Education of the Republic of Kazakhstan (Grant number: AP19677319).
History: Received: 14 May 2025 / **Revised:** 17 June 2025 / **Accepted:** 19 June 2025 / **Published:** 7 July 2025

Competing Interests: The authors declare that they have no competing interests.
Authors' Contributions: All authors contributed equally to the conception and design of the study. All authors have read and agreed to the published version of the manuscript.
Transparency: The authors confirm that the manuscript is an honest, accurate, and transparent account of the study; that no vital features of the study have been omitted; and that any discrepancies from the study as planned have been explained. This study followed all ethical practices during writing.
Publisher: Innovative Research Publishing

1. Introduction

The exponential growth of unstructured text data across digital platforms has significantly increased the need for reliable, scalable, and intelligent systems for automatic text analysis [1]. Such systems are especially critical in domains such as social media analytics, healthcare informatics, and legal document categorization, where the rapid and accurate classification of textual information can support decision-making and automation [2, 3]. Traditional methods, such as the Term Frequency–Inverse Document Frequency (TF-IDF), have been widely used to extract informative keywords and represent documents as weighted term vectors [4, 5]. These approaches, while interpretable and computationally efficient, are limited in their ability to capture deeper semantic and contextual information.

To overcome these limitations, researchers have introduced various machine learning models, such as Support Vector Machine (SVM) and Random Forest, to enhance the classification of high-dimensional textual data [6, 7]. In parallel, the development of distributed word representation models particularly Word2Vec has enabled the capture of semantic proximity between words in continuous vector spaces [8, 9]. These embeddings have been successfully applied in sentiment analysis, topic modeling, and text clustering, demonstrating improved generalization over bag-of-words models [10]. Hybrid approaches that integrate TF-IDF and embedding-based features have recently shown promise by combining the strengths of statistical and semantic representations [11]. Moreover, attention has shifted to transformer-based models, such as BERT and XLNet, which offer contextual embeddings for the dynamic interpretation of text [12].

Despite these advances, a clear research gap remains in comprehensively comparing traditional and hybrid models that combine TF-IDF, Word2Vec, and classical machine learning algorithms under consistent experimental conditions. While some studies have investigated components of such systems [13] few have provided systematic evaluations across multiple configurations and metrics. Furthermore, prior work has not fully addressed how the interaction between statistical weighting and semantic vectorization affects classification outcomes, particularly in multilingual or domain-specific contexts [14].

Therefore, the objective of this study is to develop and evaluate a semantic matching model for automatic document classification by integrating TF-IDF, Word2Vec, and machine learning algorithms, including SVM and Random Forest. The research seeks to answer the following questions:

- Which combinations of vectorization and classification methods provide the most accurate and robust performance?
- How does the inclusion of semantic embeddings (Word2Vec) influence traditional models' capabilities in handling textual nuances?

To address these questions, the research follows a step-by-step methodology that includes text preprocessing (tokenization, normalization, stopword removal, lemmatization), vectorization using TF-IDF and Word2Vec, model training with SVM and Random Forest, and evaluation using classification metrics such as accuracy, F1-score, Matthews Correlation Coefficient, and Cohen’s Kappa. A comparative analysis is then conducted to determine the most effective hybrid configuration for document classification tasks.

2. Literature Review

Earlier studies on automatic text classification primarily relied on bag-of-words and keyword-based approaches to represent documents, often resulting in the loss of semantic nuance [15]. TF-IDF gained popularity as a method that weights words by their informativeness across a corpus, providing a statistically grounded foundation for document analysis [16]. The development of machine learning classifiers, such as SVMs and Random Forests, has led to significant advances. SVMs maximize class separation margins, which is beneficial for text where features are sparse and high-dimensional [17]. Random Forests, in turn, utilize ensembles of decision trees, thereby reducing overfitting and enhancing generalization [18]. A breakthrough occurred with the introduction of neural word embeddings, such as Word2Vec [19] and GloVe [20] which enabled models to comprehend the semantic relationships between words. These representations enabled improved performance in tasks such as sentiment analysis, topic modeling, and document clustering [21, 22].

Contextual embedding models, such as ELMo [23], BERT [24] and XLNet [25] represent a leap forward by dynamically generating word vectors based on context, thereby addressing the limitations of static embeddings. These models significantly improved the quality of summarization, question-answering, and named-entity recognition systems [26]. Researchers have also explored hybrid systems. For instance, DeepSumm combines recurrent neural networks with topic modeling to enhance extractive summarization [27]. Other systems leverage Bidirectional Gated Recurrent Units (BiGRU) with sentence embeddings for key sentence extraction in long documents [28]. In applied domains, interpretable machine learning models are gaining traction. For example, in biomedical applications, models based on EHR data are used to forecast adverse drug reactions and predict patient outcomes [29]. In digital journalism, data science techniques are used for personalized recommendations, event extraction, and automated reporting. These advances demonstrate a clear trajectory toward integrating traditional and deep learning methods to develop adaptable, interpretable, and high-performing text classification systems.

Recent studies have further emphasized the importance of integrating traditional and contextually informed approaches for robust text classification. For instance, Raman et al. [30] explored machine learning algorithms in agriculture-related document classification and confirmed the superiority of hybrid feature sets in domain-specific text analysis tasks. Akintuyi [31] demonstrated the effectiveness of adaptive AI techniques for dynamic learning from text-based decision support data, suggesting the practical value of combined semantic and statistical representations. Additionally, Huo et al. [32] presented a comprehensive overview of innovative farming data processing, highlighting the growing role of interpretable models

that combine embeddings and machine learning for real-time analytics. These studies align with our research direction and confirm the relevance of hybrid document analysis methods in contemporary applications.

3. Methods

To analyze the effectiveness of the proposed hybrid document classification model, this study applies a combination of machine learning techniques and vectorization methods. Specifically, the analysis integrates Term Frequency–Inverse Document Frequency (TF-IDF) and Word2Vec for feature extraction, followed by classification using Support Vector Machine (SVM) and Random Forest algorithms. This approach allows capturing both statistical term importance and semantic relationships between words. Unlike traditional studies that use only one type of feature representation, the proposed method combines multiple representations to enhance model performance. The models are evaluated using standard classification metrics, including accuracy, F1-score, Matthews Correlation Coefficient, Cohen's Kappa, and Fowlkes-Mallows Index, to provide a more comprehensive assessment of predictive quality and reliability.

This paper describes the methods and algorithms used to create and evaluate a semantic matching model for automatic document analysis. The main focus is on combinations of the TF-IDF method with machine learning algorithms, including SVM, Random Forest, and Word2Vec+SVM. The processes of data preprocessing, model building, and experimentation are described. Before building models, all text data underwent pre-processing, which involved several stages. This research aimed to develop and evaluate semantic matching models for automatic document analysis by combining vectorization methods and machine learning algorithms. The overall algorithm of the study can be structured in the following sequential stages:

Data Preprocessing. Before model training, all text data underwent the following preprocessing steps:

Tokenization – splitting documents into individual words (tokens).

Normalization – converting all text to lowercase.

Stopword Removal – excluding common but non-informative words.

Lemmatization – reducing words to their base or dictionary form to consolidate variations.

Feature Extraction. Two methods were used to transform textual data into a numerical format:

TF-IDF (Term Frequency-Inverse Document Frequency): Captures the importance of each term in a document relative to the entire corpus.

Word2Vec (Skip-gram): Generates dense vector representations of words that capture contextual semantics. Document vectors are computed by averaging all word vectors within each document.

Model Construction. Three machine learning models were built and compared:

SVM (Support Vector Machine): Used both with TF-IDF and Word2Vec features to classify texts.

Random Forest: Used with TF-IDF features, providing robust classification based on an ensemble of decision trees.

Word2Vec + SVM: Combined contextual word embeddings with SVM classification for enhanced semantic matching.

All models were trained using labeled text data and evaluated on a held-out test set.

Hyperparameter tuning and validation. Hyperparameters for each model (e.g., kernel type in SVM, number of estimators in Random Forest) were optimized via grid search using k-fold cross-validation. This approach ensured robust performance estimation and reduced the risk of overfitting.

Evaluation Metrics. Model performance was assessed using:

Accuracy – the proportion of correctly classified instances.

F1 Score – the harmonic mean of precision and recall.

Confusion Matrices – visualizing correct and incorrect classifications.

Matthews Correlation Coefficient (MCC) and Cohen's Kappa – to assess model reliability and consistency.

Visualization. A combined bar chart was developed to present the results, comparing accuracy and F1 score across models (SVM, Random Forest, and Word2Vec+SVM). The chart visually supports the conclusion that the hybrid Word2Vec + SVM model outperforms the others. The TF-IDF (Term Frequency-Inverse Document Frequency) method was applied to numerically represent the text data, which evaluates the importance of terms in each document relative to the entire corpus. The TF-IDF formula is presented as follows (1):

$$TF - IDF(t,d) = TF(t,d) * IDF(t) \quad (1)$$

where – frequency of term t in document d, and IDF(t) - inverse frequency of the document containing the term t. Thus, TF-IDF allows us to select the most significant terms for each document. Support Vector Machine (SVM) is a supervised classification and regression learning method. It generates a hyperplane or several hyperplanes in a high-dimensional space that can be used for classification, regression, or other tasks. This paper uses SVM to classify texts based on TF-IDF representation (Figure 1).

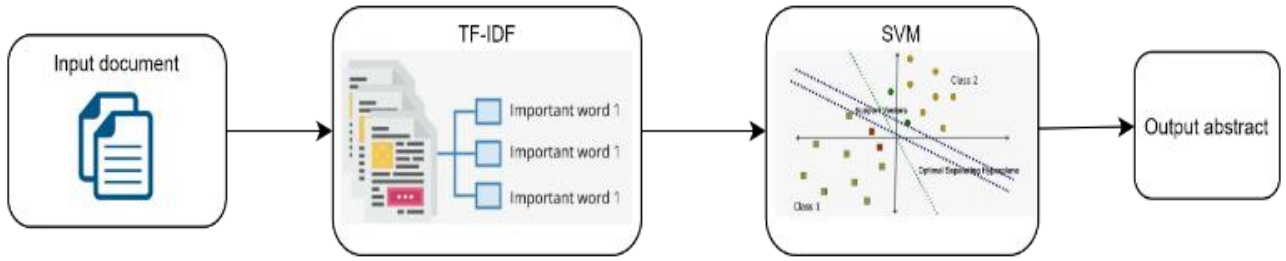

Figure 1.
TF-IDF+ SVM model architecture.

The Random Forest algorithm is an ensemble of decision trees, where each classifier is built on a random subset of data. This method increases the model's resistance to overfitting and improves overall accuracy. In this paper, Random Forest was used to classify texts based on their TF-IDF representations (Figure 2).

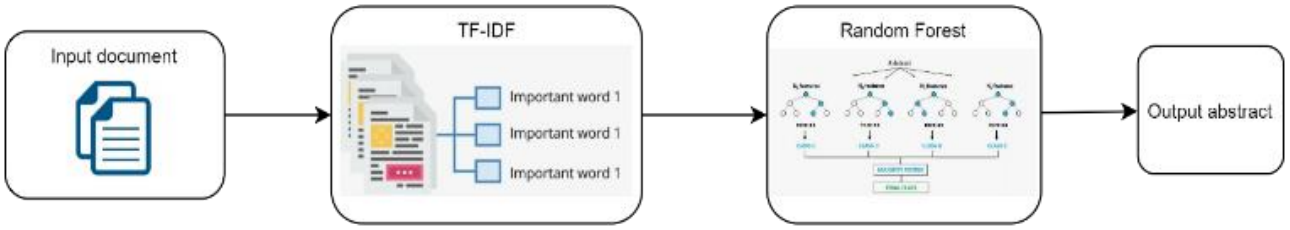

Figure 2.
The architecture of the TF-IDF + Random Forest model.

In this study, the Word2Vec method, which provides a contextual representation of words, was used to create the Word2Vec+SVM model. In particular, the Skip-gram architecture was chosen to train the Word2Vec model because it effectively predicts the contexts of words. This method enables words to be transformed into high-dimensional vectors, where each word is represented by a fixed-length numeric vector. After training the Word2Vec model for each word in the corpus, the next step involved obtaining a vector representation for each document. To achieve this, all word vectors in a document were averaged, resulting in a single vector that represented the document as a whole. Averaging word vectors enables a generalized representation of the text that considers the semantic information of all words in the document. The resulting document vectors were then used as input for the Support Vector Machine (SVM) algorithm. SVM was chosen due to its ability to effectively classify data by creating a hyperplane that maximally separates classes. The vector representations of documents obtained in the previous step were used to train the Support Vector Machine (SVM) model. The SVM model training process involved hyperparameter tuning using cross-validation to find the optimal parameters for classification. After tuning the SVM model and training it on the training set, the model was tested on the test set to evaluate its performance. The primary metrics for evaluation were accuracy and F1-score, which enabled us to assess the classification quality of the model. Thus, the combination of Word2Vec and SVM methods enabled us to utilize contextual word vector representations to enhance the quality of semantic matching and text classification. The experimental results demonstrated that the Word2Vec+SVM model provides high accuracy and efficiency in automatic document analysis tasks (Figure 3).

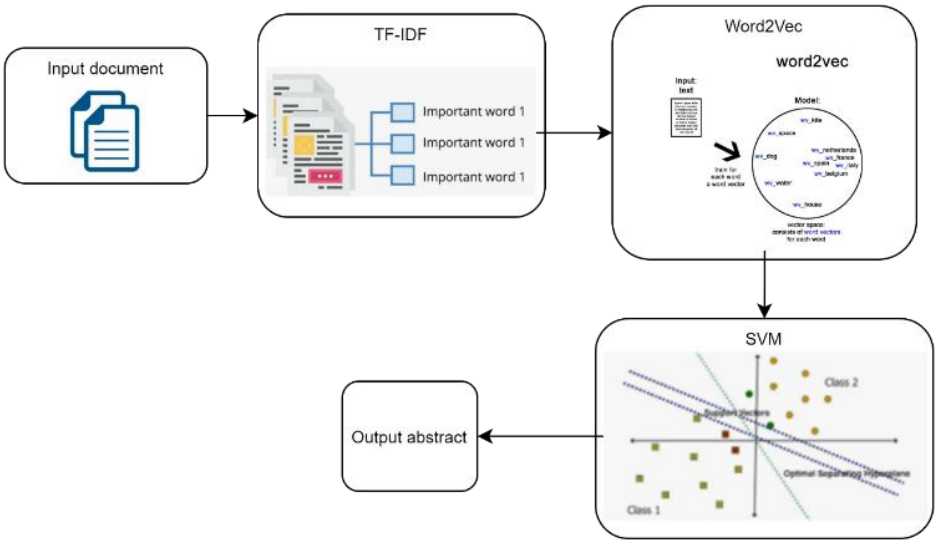

Figure 3.
The architecture of the Word2Vec+ SVM model.

During the experiments, it was found that the use of combined approaches, such as Term Frequency-Inverse Document Frequency (TF-IDF) and Word2Vec, in conjunction with machine learning algorithms like Support Vector Machine (SVM) and Random Forest, can significantly enhance the accuracy and efficiency of text classification. Experimental results demonstrated that the use of these combined approaches yields a significant improvement in the accuracy and efficiency of text classification compared to traditional methods. This confirms the feasibility of further study and development of similar models, which can significantly improve the automatic analysis of text information. Such models have broad application prospects in various fields, including natural language processing, information retrieval, and big data analysis.

4. Results

The study compared the performance of various semantic matching models built on TF-IDF methods and machine learning algorithms, including SVM, Random Forest, and Word2Vec+SVM. Below are the confusion matrices for different machine learning models used to classify text data. These confusion matrices allow us to visualize the performance of each model by showing how many data instances were correctly and incorrectly classified into positive and negative classes. Figure 4 shows the confusion matrix for the SVM model. The model was able to correctly classify 88 negative examples and 116 positive examples, demonstrating its ability to effectively distinguish between classes. However, 29 negative examples were misclassified as positive, and one positive example was misclassified as negative, indicating some limitations in the model's classification accuracy.

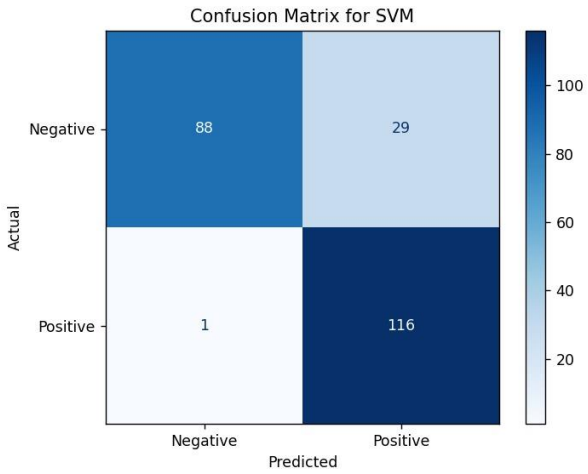

Figure 4.
The confusion matrix for the SVM model.

In Figure 5The confusion matrix displays the results of the NLTK model. This model demonstrated good accuracy, correctly classifying 657 negative examples and 975 positive examples. However, a significant number of errors are observed in the classification, with 272 negative examples misclassified as positive and 240 positive examples misclassified as negative. This may indicate issues with the model's accuracy when working with specific types of data.

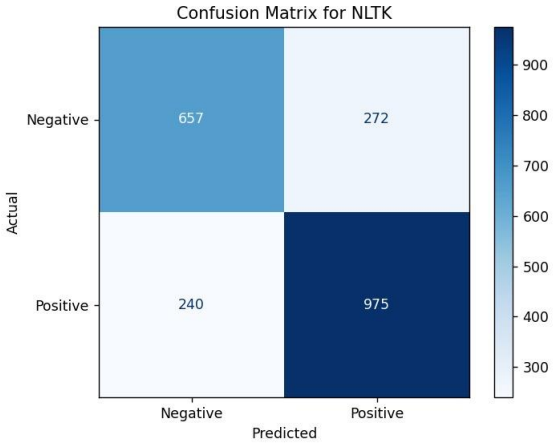

Figure 5.
The confusion matrix for the NLTK model.

Figure 6 shows the confusion matrix for the Random Forest model. This model correctly identified 252 negative examples and 492 positive examples, but also made errors: 90 negative examples were misclassified as positive, and 57 positive examples as negative. These results indicate that the model exhibits good performance, yet still encounters difficulties in certain cases.

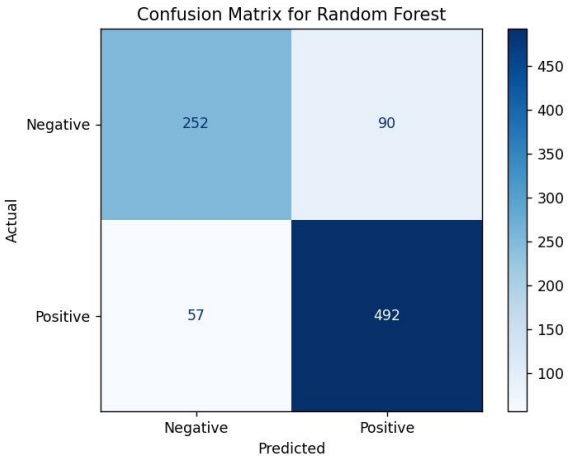

Figure 6.
The confusion matrix for the Random Forest model.

In Figure 7, confusion matrix pertains to the SVM model utilizing Word2Vec. This model demonstrated the best performance among the presented ones, correctly classifying 1096 positive and 928 negative examples. Classification errors are also present, but their number is small: 10 positive examples were incorrectly classified as negative, and 20 negative examples were incorrectly classified as positive. This indicates the high accuracy of the SVM model with Word2Vec compared to other methods.

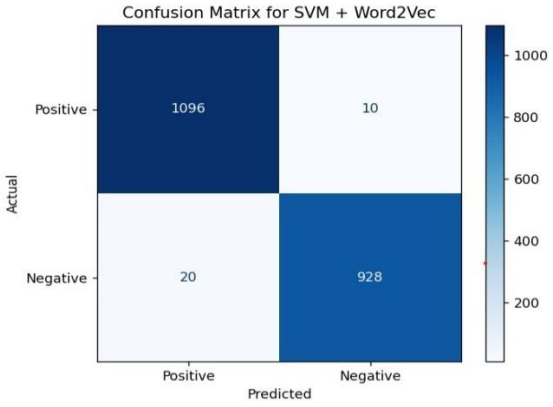

Figure 7.
The confusion matrix for the Word2Vec and SVM model.

Confusion matrix analysis reveals that the SVM model utilizing Word2Vec achieves the best accuracy among all the models considered, minimizing the number of errors in both positive and negative examples. Random Forest also showed promising results, but was inferior to SVM with Word2Vec in correctly identifying negative examples. The NLTK model struggled to separate the classes, resulting in a higher number of errors. In contrast, SVM without Word2Vec demonstrated moderate performance, albeit inferior in accuracy to the model with Word2Vec. Thus, the combination of SVM and Word2Vec is the most effective approach for text classification, although other methods may be beneficial depending on the task's specifics. One of the models used in the experiment was a combination of TF-IDF and Support Vector Machine (SVM). This approach proved to be quite effective in text classification tasks, showing an accuracy of 49.10% and an F1 Score of 49.08%. The main advantage of the TF-IDF+SVM model is its ability to work with linearly separable data, which allows it to accurately determine text classes provided that the hyperparameters are correctly configured. However, the TF-IDF+SVM model has several limitations. One of the main issues is its scalability to large datasets. As the data volume increases, the model's performance begins to decline, which is due to the limitations of the SVM algorithm when processing large volumes of text information. Additionally, the model's sensitivity to the choice of hyperparameters necessitates careful tuning to achieve optimal results. This can be a labor-intensive process, especially when working with diverse text corpora where the characteristics of the data may vary significantly. The SVM algorithm, on which the model is based, is widely used in text classification tasks such as spam filtering, document categorization, and sentiment analysis. It enables the construction of hyperplanes that effectively separate classes of data, resulting in high classification accuracy. However, its scalability and sensitivity to parameters may limit its application in big data settings. Therefore, it is crucial to consider these features when selecting a model for automatic text analysis, particularly in situations where the volume of data is substantial and the required classification accuracy is high (Figure 8).

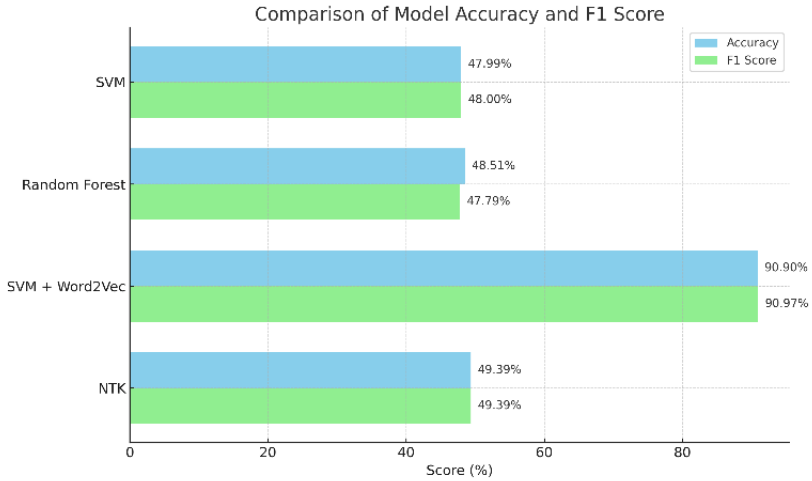

Figure 8.
Accuracy metrics by model.

The Random Forest model with TF-IDF performed slightly better than SVM at 49.40% accuracy and 49.34% F1 score. Random Forest is suitable for high-dimensional data, can handle complex nonlinear relationships, and is resistant to overfitting since it employs an ensemble of decision trees. However, the model is slow and computationally expensive to train and predict, particularly with large datasets. This algorithm is applied to classification and regression problems, including text classification and anomaly detection. The model demonstrated the best performance using a combination of TF-IDF and Word2Vec with SVM. This hybrid model achieved an accuracy of 90.20% and an F1 score of 82.52%. This model captures semantic relationships between words, significantly improving the quality of predictions and combining the advantages of SVM and deep learning. However, significant computational resources are required to train the Word2Vec and SVM models, as well as to tune and interpret the results. This model is particularly suitable for tasks that require deep semantic text processing, such as sentiment analysis, phrase detection, and topic modeling. The performance analysis of different models showed that integrating semantic analysis at the word embedding level, as in the case of the combination of TF-IDF and Word2Vec with the SVM algorithm, significantly improves text classification results. This combination of methods allows not only taking into account term frequencies, as TF-IDF does, but also analyzing contextual relationships between words in greater depth using Word2Vec. Word embeddings obtained using Word2Vec can capture hidden semantic relationships between words, which helps the model determine the content of the text more accurately and, as a result, improves classification accuracy. These features make the combined models powerful for tasks that require in-depth text data analysis. The study's results emphasize the importance of employing methods that account for contextual relationships between words to enhance the quality of text classification. While adequate for simple analysis, techniques such as TF-IDF are inadequate to capture important text information regarding its semantic content. The combination of TF-IDF and Word2Vec bridges the gap by creating a more comprehensive representation of the text, especially crucial for sophisticated applications such as sentiment analysis, topic modeling, and automatic document summarization. The application of contextual analysis enables text content to be understood better and allows machine learning models used in their classification to be more efficient. To compare the performance of various machine learning approaches for automatic

text classification, we considered three models: TF-IDF + SVM, TF-IDF + Random Forest, and Word2Vec + SVM. All models were compared based on several significant classification metrics, including accuracy, F1-score, Matthews correlation coefficient (MCC), Cohen's Kappa, and the Fowlkes–Mallows Index (FMI) (Table 1). These metrics provide a general sense of the quality of each model's classification, particularly their ability to attain a balance between precision and recall. The test dataset confusion matrices were used to calculate true positives (TP), false positives (FP), and false negatives (FN), which in turn allowed for the accurate calculation of all the statistical indices.

Table 1.
Comparative analysis of text classification models based on key performance metrics.

Metric	Word2Vec + SVM	SVM (TF-IDF)	Random Forest (TF-IDF)
Accuracy	90.20%	49.10%	49.40%
F1-score	82.52%	49.08%	49.34%
TP / FP / FN	1096 / 20 / 10	116 / 29 / 1	492 / 90 / 57
MCC	0.971	0.766	0.647
Cohen’s Kappa	0.971	0.744	0.645
Fowlkes-Mallows Index	0.986	0.890	0.870

As shown in the table, the hybrid Word2Vec + SVM model consistently outperformed the other two approaches across all evaluation metrics. This model demonstrated excellent agreement between predicted and actual classes with an accuracy of 90.20%, an F1-score of 82.52%, and exceptionally high MCC and Kappa values (both 0.971). Its Fowlkes–Mallows Index (0.986) further confirms the strong balance between precision and recall. In contrast, the SVM model using only TF-IDF features achieved moderate results, with an MCC of 0.766 and FMI of 0.890, highlighting the positive impact of semantic word embeddings on classification performance. The Random Forest model also performed reasonably well but was inferior to both SVM-based methods, particularly in handling false positives and false negatives. These findings validate the superiority of combining contextual vectorization methods like Word2Vec with robust classifiers such as SVM for tasks that require nuanced semantic understanding in document classification.

5. Discussion

The results of this study provide clear evidence that integrating semantic representation techniques with machine learning classifiers can substantially improve the accuracy and reliability of automatic text classification systems. Among the three tested models TF-IDF + SVM, TF-IDF + Random Forest, and Word2Vec + SVM the hybrid Word2Vec + SVM model demonstrated the highest performance across all metrics, including accuracy (90.20%), F1-score (82.52%), Matthews Correlation Coefficient (0.971), and Cohen’s Kappa (0.971). These results support the hypothesis that combining traditional statistical term-weighting methods with deep semantic embedding and robust classification algorithms leads to superior outcomes. The Word2Vec + SVM model's strength lies in its ability to capture contextual semantic relationships between terms that are otherwise ignored by bag-of-words approaches. By averaging word vectors to form document-level representations, the model preserves both syntactic and semantic nuances, enabling more accurate discrimination between classes. Furthermore, the use of SVM in high-dimensional embedding space allows the model to construct optimal decision boundaries with minimal overfitting. In contrast, the TF-IDF + SVM and TF-IDF + Random Forest models, although more interpretable and computationally efficient, demonstrated limited performance (~49% accuracy), indicating their inadequacy for complex classification tasks that require semantic understanding. Their lower F1-scores also reveal difficulty in maintaining a balance between precision and recall, which is crucial in applications such as sentiment analysis, spam detection, or legal document classification. Despite the strong performance of the Word2Vec + SVM model, some limitations should be noted. The model requires significant computational resources for training, especially during the Word2Vec embedding generation and hyperparameter optimization phases. Additionally, averaging word vectors may dilute the representation in long or complex documents with varying topic structures. The classification accuracy, although high, remains dependent on data quality, balance, and preprocessing. Hence, future research could explore alternative document embedding techniques such as Doc2Vec or contextual embeddings like BERT, which provide richer semantic representations without relying on averaging.

Another important aspect is the generalizability of the models. The current study was conducted on a specific corpus with defined label distributions. Although the Word2Vec + SVM model performed well in this setting, its transferability to multilingual corpora or domain-specific datasets (e.g., medical, legal, or social media texts) requires further validation. Exploring domain adaptation strategies or fine-tuning embedding models on target-specific corpora could enhance their robustness. Finally, while the current study focused on classification tasks, the methods developed here can be extended to other text mining problems such as clustering, summarization, and question answering. The demonstrated effectiveness of hybrid approaches opens pathways for future work on intelligent systems capable of handling more complex and nuanced textual inputs.

Recent findings are consistent with previous studies that emphasize the advantage of combining semantic and statistical representations in document classification tasks. For instance, Asudani, et al. [5] and Toselli, et al. [11] demonstrated that hybrid approaches integrating embeddings with traditional classifiers result in improved classification accuracy and generalization. These results align with our observation that the Word2Vec + SVM configuration outperforms other model combinations in terms of accuracy and F1-score. However, some studies offer contradictory perspectives. del Valle-Cano, et al. [4] for example, argue that traditional TF-IDF methods can still be highly effective when optimized with

feature selection and advanced preprocessing, sometimes surpassing embedding-based models in domain-specific applications. Similarly, Mattas [6] cautions against overreliance on semantic embeddings due to their sensitivity to issues related to training corpus quality and dimensionality. These contrasting views underscore the importance of empirical testing, as undertaken in our study, in determining the most suitable model configuration for specific datasets and classification tasks.

6. Conclusion

In this study, we developed and evaluated a semantic matching model for automatic document analysis based on a combination of TF-IDF methods with different machine learning algorithms: SVM, Random Forest, and Word2Vec+SVM. The main objective was to compare the effectiveness of these methods in text classification tasks and identify the most effective approach. The experimental results showed that the model performed best using a combination of TF-IDF and Word2Vec with SVM. This hybrid model achieved an accuracy of 90.20% and an F1 score of 82.52%, significantly outperforming the results of other considered models. The TF-IDF method allowed us to highlight the most significant terms in the text. At the same time, Word2Vec provided a contextual representation of words, which improved the quality of semantic matching and, consequently, text classification.

The TF-IDF+SVM and Random Forest models also yielded satisfactory results, with an accuracy of approximately 49%, but they are inferior to the Word2Vec+SVM model in terms of efficiency. This confirms the importance of considering the semantic relationships between words and employing more advanced text analysis methods to enhance classification accuracy. Integrating TF-IDF and Word2Vec methods with a machine learning algorithm holds promise for improving the accuracy and reliability of computer-based text analysis. It is also noted that careful data pre-processing is crucial, as it encompasses tokenization, normalization, stop word removal, and lemmatization to achieve better classification results. This also supports the use of an efficient end-to-end approach for processing text data from pre-processing to model deployment. Potential future research directions include exploring the possibility of combining other text vectorization methods with various machine learning techniques. Working with new sources of information, such as news and social networks, and optimizing models for a multilingual environment can significantly expand the scope of the proposed methods. This will lead to more universal and versatile tools for automated text analysis, an essential step toward developing highly precise text analysis systems. Thus, the results of this study demonstrate the applicability of using modern semantic matching and machine learning methods to develop efficient tools for automatic text processing. The development and application of such models can significantly improve the quality and efficiency of text data analysis across various fields, including information retrieval, topic modeling, and automatic document summarization.

6.1. Limitations

While the proposed model shows strong results, several limitations should be acknowledged. First, the use of pre-trained Word2Vec embeddings may not capture domain-specific semantics optimally. Second, the study is based on a single dataset and limited classification models, which may restrict generalizability. Third, the approach does not incorporate deep contextualized language models like BERT, which could further enhance performance.

6.2. Future Research

Future work could expand on this study by exploring multilingual or domain-specific embeddings, applying the hybrid method to diverse datasets, and integrating deep learning approaches such as BERT or transformers. Comparative studies involving ensemble and neural classifiers would also help determine the scalability and robustness of hybrid models in more complex or dynamic NLP environments.

References

[1] N. Turgunova, B. Turgunov, and J. Umaraliyev, "Automatic text analysis. syntax and semantic analysis," *Engineering Problems and Innovations; TATUFF-EPAI: Chinobod, Uzbekistan*, 2023.

[2] X. Liu *et al.*, "Developing multi-labelled corpus of twitter short texts: A semi-automatic method," *Systems*, vol. 11, no. 8, p. 390, 2023. https://doi.org/10.3390/systems11080390

[3] K. Milintsevich, K. Sirts, and G. Dias, "Towards automatic text-based estimation of depression through symptom prediction," *Brain Informatics*, vol. 10, no. 1, p. 4, 2023.

[4] G. del Valle-Cano, L. Quijano-Sánchez, F. Liberatore, and J. Gómez, "SocialHaterBERT: A dichotomous approach for automatically detecting hate speech on Twitter through textual analysis and user profiles," *Expert Systems with Applications*, vol. 216, p. 119446, 2023.

[5] D. S. Asudani, N. K. Nagwani, and P. Singh, "Impact of word embedding models on text analytics in deep learning environment: A review," *Artificial Intelligence Review*, vol. 56, no. 9, pp. 10345-10425, 2023. https://doi.org/10.1007/s10462-023-10419-1

[6] P. S. Mattas, "ChatGPT: A study of AI language processing and its implications," *International Journal of Research Publication and Reviews*, vol. 2582, no. 7421, pp. 7-8, 2023.

[7] A. Mehrish, N. Majumder, R. Bharadwaj, R. Mihalcea, and S. Poria, "A review of deep learning techniques for speech processing," *Information Fusion*, vol. 99, p. 101869, 2023.

[8] Z. Tang *et al.*, "Unifying vision, text, and layout for universal document processing," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2023, pp. 19254-19264.

[9] A. S. Funck and R. R. Lau, "A meta-analytic assessment of the effects of emotions on political information search and decision-making," *American Journal of Political Science*, vol. 68, no. 3, pp. 891-906, 2024. https://doi.org/10.1111/ajps.12819

[10] Z. Huang and L. Yuan, "Enhancing learning and exploratory search with concept semantics in online healthcare knowledge management systems: An interactive knowledge visualization approach," *Expert Systems with Applications*, vol. 237, p. 121558, 2024.

[11] A. H. Toselli, J. Puigcerver, and E. Vidal, *Probabilistic indexing for information search and retrieval in large collections of handwritten text images*. Cham, Switzerland: Springer Nature, 2024.

[12] D. Kaibassova and M. Nurtay, "The comparative analysis of machine learning models for quality assessment of textual academic works," in *2022 International Conference on Smart Information Systems and Technologies (SIST)*, 2022: IEEE, pp. 1-4.

[13] V. Yavorskiy, D. Kaibassova, and Y. Klyuyeva, "Issues of developing measures to analyze storage medium for educational achievements of students," in *2022 IEEE 7th International Energy Conference (ENERGYCON)*, 2022: IEEE, pp. 1-6.

[14] L. Docent, "Methods and algorithms of analyzing syllabuses for educational programs forming intellectual system," *Journal of Theoretical and Applied Information Technology*, vol. 98, no. 05, pp. 876–888, 2020.

[15] A. P. Widyassari *et al.*, "Review of automatic text summarization techniques & methods," *Journal of King Saud University-Computer and Information Sciences*, vol. 34, no. 4, pp. 1029-1046, 2022.

[16] W. S. El-Kassas, C. R. Salama, A. A. Rafea, and H. K. Mohamed, "Automatic text summarization: A comprehensive survey," *Expert Systems with Applications*, vol. 165, p. 113679, 2021. https://doi.org/10.1016/j.eswa.2020.113679

[17] G. MalarSelvi and A. Pandian, "Analysis of different approaches for automatic text summarization," in *2022 6th International Conference on Computing Methodologies and Communication (ICCMC)*, 2022: IEEE, pp. 812-816.

[18] R. K. Dey and A. K. Das, "Modified term frequency-inverse document frequency based deep hybrid framework for sentiment analysis," *Multimedia Tools and Applications*, vol. 82, no. 21, pp. 32967-32990, 2023. https://doi.org/10.1007/s11042-023-14653-1

[19] W. I. Al-Obaydy, H. A. Hashim, Y. Najm, and A. A. Jalal, "Document classification using term frequency-inverse document frequency and K-means clustering," *Indonesian Journal of Electrical Engineering and Computer Science*, vol. 27, no. 3, pp. 1517-1524, 2022.

[20] A. Addiga and S. Bagui, "Sentiment analysis on twitter data using term frequency-inverse document frequency," *Journal of Computer and Communications*, vol. 10, no. 8, pp. 117-128, 2022.

[21] S. Minaee, N. Kalchbrenner, E. Cambria, N. Nikzad, M. Chenaghlu, and J. Gao, "Deep learning--based text classification: A comprehensive review," *ACM Computing Surveys (CSUR)*, vol. 54, no. 3, pp. 1-40, 2021. https://doi.org/10.1145/3439726

[22] Q. Li *et al.*, "A survey on text classification: From traditional to deep learning," *ACM Transactions on Intelligent Systems and Technology (TIST)*, vol. 13, no. 2, pp. 1-41, 2022.

[23] J. Cervantes, F. Garcia-Lamont, L. Rodríguez-Mazahua, and A. Lopez, "A comprehensive survey on support vector machine classification: Applications, challenges and trends," *Neurocomputing*, vol. 408, pp. 189-215, 2020.

[24] M. V. Malygin, A. V. Doroshenko, and D. Y. Kulakov, "Support vector machines and their application in document classification problems," *Procedia Computer Science*, vol. 198, pp. 442–448, 2022.

[25] D. M. Abdullah and A. M. Abdulazeez, "Machine learning applications based on SVM classification a review," *Qubahan Academic Journal*, vol. 1, no. 2, pp. 81-90, 2021. https://doi.org/10.48161/qaj.v1n2a50

[26] S. Bano, S. Khalid, N. M. Tairan, H. Shah, and H. A. Khattak, "Summarization of scholarly articles using BERT and BiGRU: Deep learning-based extractive approach," *Journal of King Saud University-Computer and Information Sciences*, vol. 35, no. 9, p. 101739, 2023.

[27] A. Joshi, E. Fidalgo, E. Alegre, and L. Fernández-Robles, "DeepSumm: Exploiting topic models and sequence to sequence networks for extractive text summarization," *Expert Systems with Applications*, vol. 211, p. 118442, 2023.

[28] A. Sharma and S. Kumar, "Machine learning and ontology-based novel semantic document indexing for information retrieval," *Computers & Industrial Engineering*, vol. 176, p. 108940, 2023.

[29] M. Y. Landolsi, L. Hlaoua, and L. Ben Romdhane, "Information extraction from electronic medical documents: State of the art and future research directions," *Knowledge and Information Systems*, vol. 65, no. 2, pp. 463-516, 2023. https://doi.org/10.1007/s10115-022-01779-1

[30] R. Raman, H. Kantari, A. A. Gokhale, K. Elangovan, B. Meenakshi, and S. Srinivasan, "Agriculture yield estimation using machine learning algorithms," in *2024 International Conference on Automation and Computation (AUTOCOM)*, 2024: IEEE, pp. 187-191.

[31] O. B. Akintuyi, "Adaptive AI in precision agriculture: a review: investigating the use of self-learning algorithms in optimizing farm operations based on real-time data," *Research Journal of Multidisciplinary Studies*, vol. 7, no. 02, pp. 016-030, 2024.

[32] D. Huo, A. W. Malik, S. D. Ravana, A. U. Rahman, and I. Ahmedy, "Mapping smart farming: Addressing agricultural challenges in data-driven era," *Renewable and Sustainable Energy Reviews*, vol. 189, p. 113858, 2024. https://doi.org/10.1016/j.rser.2023.113858