COMPLETE
Journal of Computer, Electronic, and Telecommunication
ISSN 2723-4371, E-ISSN 2723-5912

Article

Application Of TF-IDF And Word2vec For Feature Extraction In Sentiment Analysis Of Free Nutritious Food Policies

Qisthi Annisa Haq[1], Alam Rahmatulloh[2]

[1-2] Department of Engineering, Siliwangi University, Tasikmalaya, Indonesia
* Correspondence: alam@unsil.ac.id

Received: 1 December 2025; Revised: 19 December 2025; Accepted: 30 December 2025; Published: 31 December 2025

Abstract: The free nutritious meal policy has become a hot topic of discussion among the public because it is related to improving health and education quality. However, its implementation has given rise to a variety of pros and cons that need to be analyzed systematically. This study aims to analyze sentiment toward the policy by utilizing Term Frequency–Inverse Document Frequency (TF-IDF) and Word2Vec as feature extraction methods on public review data obtained from social media X. After undergoing preprocessing and automatic labeling, the data was classified into positive and negative sentiments using the Support Vector Machine (SVM) algorithm. The analysis results show that the sentiment data is unbalanced, with the positive class dominating at 75% and the negative class at 25%. In model testing, TF-IDF achieved an accuracy of 81%, while Word2Vec achieved an accuracy of 80%. This difference shows that TF-IDF is more stable in handling short and informal texts, while Word2Vec still has the potential to capture the semantic context between words. This research opens up opportunities for further research, it is recommended to balance the data between classes and combine the TF-IDF and Word2Vec methods, or use a deep learning approach such as BERT to obtain more accurate results and capture deeper semantic context.

Keywords: Accuracy, Automatic Labelling, Positive and Negative Sentiments, Public Review Data, Semantic Context

1. Introduction

The development of social media has become one of the main platforms for debating government policies. Social media, which initially served only as a platform for personal interaction, has now evolved into the main arena for shaping and disseminating public opinion [1]. One of the most hotly debated public opinions is the government's Free Nutritious Meals (MBG) policy. This free nutritious meals program aims to provide lunch and milk in the hope of overcoming malnutrition and stunting among school children [2]. However, its implementation has elicited mixed responses from the public. Some support it on humanitarian grounds and for the sake of nutritional equality, while others criticize the program's effectiveness, transparency, and logistical readiness [3]. There have even been incidents such as mass poisoning in several schools [4]. This further reinforces the urgency of understanding public perception of this policy.

In this case, sentiment analysis is an important way to gauge public opinion regarding government policies. With this analysis, public opinion can be divided into various categories such as positive, negative, or neutral [5]. In today's digital age, social media such as X has become one of the main sources of public opinion because the data obtained is open and real time [6]. To analyze public opinion systematically, feature extraction is needed to convert unstructured text into numerical representations that can be processed by algorithms [2]. This stage is very important because the quality of the extracted features can affect the performance of the sentiment analysis model. Two popular approaches in this field are Term Frequency-Inverse Document Frequency (TF-IDF) and Word2Vec.

The TF-IDF method is popular because it looks at how important each word is within a document. This means that words that show up a lot in a specific document but not much in all the documents together will be seen as more important [7]. In addition to being simple and easy to interpret, TF-IDF has proven to be stable as a baseline in various text processing studies [8]. However, the main weakness of TF-IDF lies in its inability to capture the semantic meaning between words. Meanwhile, the Word2Vec method converts words into dense vectors based on the words that appear in a sentence. Word2Vec works with two main methods, namely Continuous Bag of Words (CBOW) and Skip-gram. Both methods are capable of capturing the meaning and sentence structure relationships between words. In this way, words with similar meanings will have vectors that are close to each other in multidimensional space [9].

TF-IDF and Word2Vec were chosen for this research because they are two important and commonly used methods for showing text features, often serving as basic examples in studies about understanding feelings in text. It has been proven to work well in identifying feelings in short and casual messages, especially when used with classic machine learning methods like Support Vector Machine (SVM) [7], [10]. On the other hand, Word2Vec is a method that looks at the meanings of words. It creates dense vectors that keep similar words close together, helping to show how words relate to each other [9], [11].

The research conducted by Sitanggang et al. [2] focused on the use of the naïve bayes algorithm with the TF-IDF feature extraction method. Text preprocessing was carried out thoroughly, from cleaning to stemming. The results obtained from this study were an accuracy of 72.2%. Although this research can be considered sufficient, it did not compare the Word2Vec method to determine the semantic meaning between words in each dataset.

A different research project was carried out by Dani and others [10], where they used the SVM method to look at how well TF-IDF-SVM worked compared to Word2Vec-SVM. Text preprocessing was carried out thoroughly. The results of this study show that TF-IDF-SVM has a higher accuracy of 83% compared to Word2Vec at 77.5%. However, the TF-IDF method is unable to understand the semantic meaning between words.

Ati et al. [12] conducted research comparing the naïve bayes, SVM, KNN, and ensemble (stacking) algorithms with the TF-IDF feature extraction method and the SMOTE-Tomek technique for balancing the dataset and complete preprocessing. This study aimed to determine which algorithm was the most optimal. The results showed that the SVM algorithm achieved the highest accuracy of 95.05%. However, this study has limitations in the form of neutral class bias and dependence on automatic translation results, which can affect accuracy.

Furthermore, the study by Nadia et al. [13] examined sentiment analysis of COVID-19 vaccination using a Recurrent Neural Network (RNN) and two feature extraction methods, namely TF-IDF and Word2Vec. The results showed that the combination of RNN-Word2Vec produced the highest accuracy, namely 53%, and the combination of RNN-TF-IDF produced 51%. Although the improvement is relatively small, it shows that feature extraction with Word2Vec is superior. The low accuracy is due to the variation in the algorithms used, so that in future research, other more suitable algorithms can be used.

Finally, research by Zhan [11] compared feature extraction between TF-IDF and Word2Vec for food reviews using the Logistic Regression algorithm. Based on testing, TF-IDF had a percentage of 99.16% on the training data, but had a significant decrease on the test data of 73.9%. Meanwhile, using Word2Vec feature extraction showed more stable results with an accuracy of 68.4% for the training data and 68.65% on the test data. This shows that TF-IDF is more prone to overfitting than Word2Vec.

Based on several studies presented earlier, most studies rely on preprocessing stages such as stemming and stopword removal, which have the potential to remove the natural context in sentences. Unlike previous studies that used stemming and stopword removal for text normalization, this study deliberately omitted these two stages to see how the model's performance changed when semantic features and word context were left unchanged, without morphological modification. In this way, this study not only compares the performance of TF-IDF and Word2Vec in the usual way, but also helps to understand how much influence the text preprocessing process has on the effectiveness of feature representation in public policy sentiment analysis using data from social media. So, the TF-IDF and Word2Vec ways of picking out important details with the SVM method are likely to give a better understanding of how well we can show feelings about public policy by looking at social media information.

2. Materials and Methods

The research looks at how well two ways of pulling important information from text work, specifically Term Frequency-Inverse Document Frequency (TF-IDF) and Word2Vec, when analyzing opinions about the Free Nutritious Meals (MBG) policy. The research process consists of several main stages as shown in Figure 1.

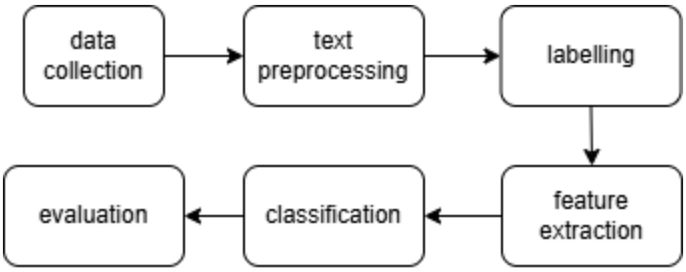


Figure 1. Research Stages

Figure 1 shows the stages of research that will be conducted, from data collection to evaluation. The following is an explanation of each research flow.

2.1 Data Collection

The research data will be obtained from Kaggle in csv format containing the results of scraping from social media X. After the cleaning process, the total data will be divided into two parts, namely 80% as training data and 20% as test data.

2.2 Text Preprocessing

This stage aims to clean the text so that it is ready to be processed in the feature extraction stage. However, unlike previous studies [4], [5], this study does not use the stemming and stopword removal stages in order to observe the influence of the natural context of words on sentiment analysis. The preprocessing stages that will be carried out are case folding, cleansing, and tokenizing. The outcome of this phase is a group of writings that can now be prepared for feature extraction.

2.3 Labeling

The labeling stage is carried out to assign a sentiment class to each text, namely positive, negative, and neutral. This process is carried out automatically using the pre-trained BERT Multilingual Sentiment model from the Hugging Face Transformers library, which is capable of recognizing sentiment in various languages, including Indonesian. The labeling results show that most of the data falls into the positive class, followed by the negative class, and a small portion of neutral data, which was then removed because there was too little to analyze further.

2.4 Feature Extraction

The part where we extract features is meant to change words into numbers so that computers can understand and work with them. Two methods are used.

1. Term Frequency-Inverse Document Frequency

   TF-IDF shows how significant a word is in one paper when you think about all the papers in a group. The TF-IDF value is determined by using a specific formula (1).

$$TF - IDF(t, d) = TF(t, d) \times \log \frac{N}{DF(t)} \tag{1}$$

   In this case, TF(t,d) shows how often the word t appears in the document d. DF(t) represents how many documents have the word t, and N is the overall count of all the documents [6]. This method produces a sparse word representation matrix and is widely used in sentiment analysis research due to its simplicity [7].

2. Word2Vec

   Word2Vec changes words into a simpler format using a lower-dimensional space by training a neural network. It does this through two main methods: Continuous Bag of Words (CBOW) and Skip-Gram. However, this study uses the Skip-Gram model because it is more effective at capturing semantic meaning for small to medium datasets [8].

2.5 Classification Using Support Vector Machine (SVM)

The SVM algorithm is used as the main classification model due to its ability to separate high-dimensional data using an optimal hyperplane and its proven stable performance in various previous analysis studies. In this study, the Radial Basis Function (RBF) kernel is used, and the model will be trained using the results of TF-IDF and Word2Vec feature extraction separately to compare their performance.

2.6 Model Evaluation

Model evaluation was performed by comparing the classification results of the TF-IDF-SVM and Word2Vec-SVM models using the accuracy, precision, recall, and F1-score metrics, with the following formulas (2).

$$\begin{gathered} Precision = \frac{TP}{TP + FP} \\ Recall = \frac{TP}{TP + FN} \\ F1\ score = 2 \times \frac{Precision}{Precision} \end{gathered} \tag{2}$$

These values are obtained from the confusion matrix of the model's predictions on the test data.

3. Results and Discussion

3.1 Data Collection

The data was obtained from a dataset available on Kaggle called "Public Sentiment towards Free Nutritious Food." This dataset contains 3,461 text data points from social media X discussing the MBG policy.

3.2 Text Preprocessing

Preprocessing steps were carried out to ensure that the data was suitable for the feature extraction stage. The steps taken included:

1. Cleaning: removing links (URLs), user tags (@username), numbers, punctuation marks, non-Latin symbols, and emojis.
2. Case folding: converting all letters to lowercase for consistency.

	full_text	cleaning	case_folding
0	Makan Siang Bergizi Gratis https://t.co/f27alt...	Makan Siang Bergizi Gratis	makan siang bergizi gratis
1	Momen Prabowo Tanda Tangani Sepatu Siswa di Bo...	Momen Prabowo Tanda Tangani Sepatu Siswa di Bo...	momen prabowo tanda tangani sepatu siswa di bo...
2	@lenteradata Semoga program makan bergizi grat...	Semoga program makan bergizi gratis dri MDA te...	semoga program makan bergizi gratis dri mda te...
3	Pemprov Babel dan DPRD Bahas Anggaran Program ...	Pemprov Babel dan DPRD Bahas Anggaran Program ...	pemprov babel dan dprd bahas anggaran program ...
4	@lenteradata Menurutku berhasil skliki program...	Menurutku berhasil skliki programnya MDA yg ma...	menurutku berhasil skliki programnya mda yg ma...

Figure 2. Cleaning and case folding stages

Figure 2 shows the results of the cleaning and case folding stages that have been carried out.

3. Tokenizing: breaking sentences into word segments using the nltk library.
4. Stopword: removing words that are considered unimportant, such as auxiliary words like di, pada, and others, using the nltk library.
5. Stemming: returning each inflected word to its root form using the sastrawi library.

tokenize	stopword	stemming
[makan, siang, bergizi, gratis]	[makan, siang, bergizi, gratis]	[makan, siang, gizi, gratis]
[momen, prabowo, tanda, tangani, sepatu, siswa...	[momen, prabowo, tanda, tangani, sepatu, siswa...	[momen, prabowo, tanda, tangan, sepatu, siswa,...
[semoga, program, makan, bergizi, gratis, dri,...	[semoga, program, makan, bergizi, gratis, dri,...	[moga, program, makan, gizi, gratis, dri, mda,...
[pemprov, babel, dan, dprd, bahas, anggaran, p...	[pemprov, babel, dprd, bahas, anggaran, progra...	[pemprov, babel, dprd, bahas, anggar, program,...
[menurutku, berhasil, skliki, programnya, mda,...	[menurutku, berhasil, skliki, programnya, mda,...	[turut, hasil, skliki, program, mda, yg, makan...

Figure 3. Tokenization, stopword, and stemming stages

Figure 3 shows the results of the tokenization, stopword, and stemming processes. The stopword and stemming processes were only performed on the TF-IDF feature extraction method because in the word2vec method, both preprocessing steps were omitted in order to preserve the semantic meaning of each word.

3.3 Labeling

After all preprocessing stages are complete, the next step is to label each tweet. Labeling is done with a pretrained model from huggingface, which is divided into two classes: positive and negative. The following is the percentage of the labeling results.

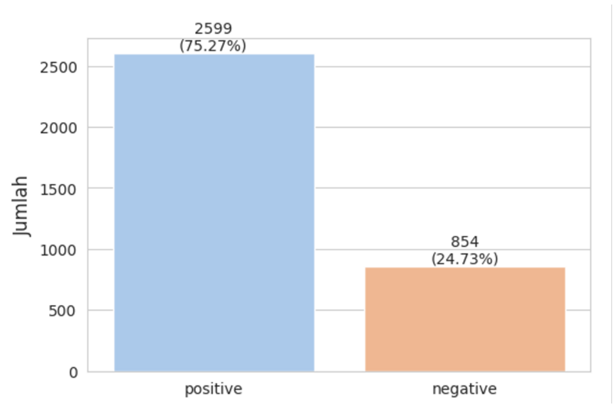


Figure 4. Sentiment percentage

Figure 4 shows that the data is unbalanced between sentiment classes. The positive class dominates at 75%, while the negative class is only 24%. This imbalance has the potential to affect model performance, especially in the ability to detect minority (negative) sentiment.

3.4 Feature Extraction

1. Term Frequency-Inverse Document Frequency (TF-IDF)
   Weighting is performed using the sklearn.feature_extraction.text library, namely Tfidfvectorizer. Feature extraction is also performed using ngram with a range of 2 because there are several words that have meaning when taken from a sentence, not just one word, such as the negative sentence "not good". The weighting process is shown in Figure 5.

```
#vektorisasi
vectorizer = TfidfVectorizer(max_features=5000, ngram_range=(1, 2))
X_train_vec = vectorizer.fit_transform(X_train)
X_test_vec = vectorizer.transform(X_test)
```

Figure 5. TF-IDF weighting process

2. Word2vec
   Weighting is performed using the gensim library by forming a dense 100-dimensional vector that represents the semantic meaning between words. The vector results are shown in Figure 6.

```
Contoh vektor: [ 4.4683829e-02  2.0138405e-01 -4.0741217e-01 -1.1170293e-01
  9.6124060e-02 -4.7956163e-01  7.4206030e-01  1.1123683e+00
 -9.2041296e-01 -5.0495160e-01  2.6599023e-02 -1.0221407e+00
 -2.2153553e-01  5.4185659e-01  4.8920295e-01 -3.4648809e-01
  7.7135265e-01 -2.1820754e-01 -2.7130967e-01 -1.5140573e+00
  1.5270750e-01 -4.2066485e-01  6.4909178e-01  8.4196448e-02
 -2.5029770e-01 -1.7926675e-01 -4.2642337e-01  9.7632274e-02
 -3.6353946e-01  2.6682001e-01  1.0827038e+00  1.3716710e-01
  1.6061056e-01 -1.0781510e+00 -2.8858504e-01  7.1740329e-01
  6.9700783e-01  5.7481136e-02 -2.8648388e-01 -8.3722836e-01
  3.4274989e-01 -4.3552020e-01 -5.7815391e-01  3.9090189e-01
  3.9069852e-01  1.6226390e-01 -8.8316655e-01 -2.7380165e-01
  5.6738073e-01  6.2506253e-01 -3.9525044e-01 -4.4990775e-01
 -7.8249162e-01 -1.9732749e-01 -2.4721050e-01  2.3442966e-01
  1.3149440e-01 -1.8677150e-01 -5.3830773e-01  4.8539239e-01
  1.3626905e-01  3.9991775e-01  7.5395894e-01  1.9297811e-01
 -8.3394444e-01  4.7199324e-01  1.1477207e+00  5.5433238e-01
 -9.8499960e-01  7.8503740e-01  2.0710133e-01  8.0250964e-02
  4.1789353e-01  2.0184699e-01  8.5436761e-01 -1.8714327e-01
 -3.3753313e-02  1.3662539e-03 -8.7240577e-01 -2.5870734e-01
  1.4543296e-01  6.4099245e-02 -2.3534240e-01  6.5208167e-01
```

Figure 6. Word2vec vector results

3.5 Classification using Support Vector Machine (SVM)

After the data was converted into vector form, the SVM model was trained using two types of input, namely TF-IDF extraction results and word2vec extraction results. The kernel used was the Radial Basis Function (RBF) kernel due to its ability to separate non-linear data well. The accuracy results using the TF-IDF feature extraction method from the modeling that has been carried out are shown in Figure 7.

```
Test Accuracy: 0.8147612156295224

              precision    recall  f1-score   support

    negative       0.61      0.56      0.58       160
    positive       0.87      0.89      0.88       531

    accuracy                           0.81       691
   macro avg       0.74      0.73      0.73       691
weighted avg       0.81      0.81      0.81       691
```

Figure 7. Accuracy Results with TF-IDF Feature Extraction

The accuracy results using the word2vec feature extraction method are shown in Figure 8.

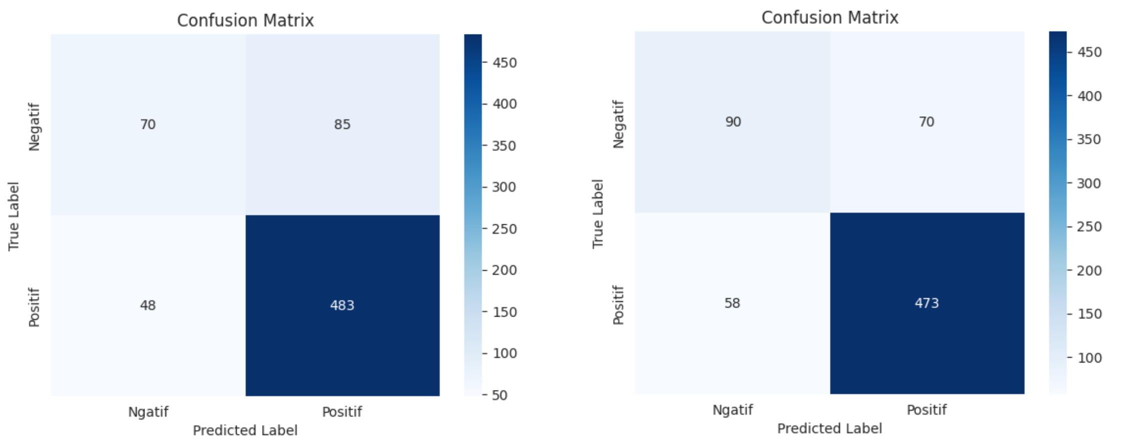
```
Test Accuracy: 0.8061224489795918
              precision    recall  f1-score   support

    negative       0.59      0.45      0.51       155
    positive       0.85      0.91      0.88       531

    accuracy                           0.81       686
   macro avg       0.72      0.68      0.70       686
weighted avg       0.79      0.81      0.80       686
```

Figure 8. Accuracy Results with word2vec Feature Extraction

The test results show that the combination of TF-IDF – SVM achieved an accuracy of 81% and word2vec – SVM achieved an accuracy of 80%. Although the difference is not significant, the TF-IDF model is slightly superior to the word2vec model. This is because TF-IDF focuses on word weights based on frequency, which is more suitable for short texts with non-standard structures such as tweets.

3.6 Model Evaluation

Model assessment was done by looking at accuracy, precision, recall, and f1-score measurements. Also, a confusion matrix was created to show how the classification mistakes were spread across different sentiment categories.



(a) (b)

Figure 9. (a) Confusion Matrix TF-IDF - SVM Model; (b) Confusion Matrix Word2vec - SVM Model

Figure 9 shows the confusion matrices for each model, indicating that TF-IDF–SVM has a precision of 74% and a recall of 73%. On the other hand, Word2Vec–SVM has a precision of 72% and a recall of 68%. Both models make more mistakes when identifying negative cases because there are far fewer negative samples than positive ones. This difference in class sizes makes the models more likely to predict positive outcomes, which then hurts their ability to recall negative sentiments.

In addition to these numbers, we can also understand why TF-IDF–SVM performs better by looking at how the algorithms work. TF-IDF creates clear and detailed feature sets that highlight important words linked to sentiment, allowing the SVM to make better decisions when analyzing short and casual texts like tweets. Meanwhile, Word2Vec creates more complex vectors that focus on how words relate to each other rather than their importance, which can weaken the signals that show sentiment when combining word vectors in short texts. Therefore, even though Word2Vec does a good job of understanding relationships between words, it does not represent sentiment as effectively when dealing with uneven data distribution, which causes a lower recall for negative cases.

4. Conclusions

This study shows that the TF-IDF-SVM method produces slightly higher accuracy, namely 81%, compared to Word2Vec-SVM with 80%. These results indicate that TF-IDF is more stable and efficient for analyzing sentiment in short Indonesian texts. Data imbalance, where the positive class dominates 75% and the negative class only 25%, affects the model's performance, which tends to recognize positive sentiment. For further research, it is recommended to balance the data between classes and combine the TF-IDF and Word2Vec methods, or use a deep learning approach such as BERT to obtain more accurate results and capture deeper semantic context.

References

1. R. Hidayat and R. Ramadhan, “Peran Media Sosial Dalam Mengkonstruksi Opini Publik Terkait Kebijakan Pemerintah: Studi Kasus Wacana Publik Tahun 2025,” *CORE: Journal of Communication Research*, vol. 3, no. 2, pp. 64–75, Jul. 2023.
2. A. Sitanggang, Y. Umaidah, Y. Umaidah, R. I. Adam, and R. I. Adam, “Analisis Sentimen Masyarakat Terhadap Program Makan Siang Gratis Pada Media Sosial X Menggunakan Algoritma Naive Bayes,” *Jurnal Informatika dan Teknik Elektro Terapan*, vol. 12, no. 3, Aug. 2024, doi: 10.23960/jitet.v12i3.4902.
3. U. Agustini, “Efektivitas dan Tantangan Kebijakan Program Makan Bergizi Gratis sebagai Intervensi Pendidikan di Indonesia,” *Jurnal Kiprah Pendidikan*, vol. 4, no. 3, pp. 362–368, Jul. 2025, doi: 10.33578/kpd.v4i3.p362-368.
4. O. Sumantri Riyanto and M. Rianita Elfrida Sinaga, “Penegakan Hak Anak Atas Makanan Aman Dan Sehat: Studi Kasus Keracunan Dalam Program Makan Bergizi Gratis Ditinjau Dari Tanggung Jawab Negara,” *Juris Humanity: Jurnal Riset dan Kajian Hukum HAM*, Jun. 2025.
5. R. Amelia, “Sentiment Analysis of Government Policy in Relocating the Republic of Indonesia’s Capital City,” *Journal of Public Administration and Government*, vol. 5, no. 2, pp. 177–187, Aug. 2023, [Online]. Available: https://jurnal.fisip.untad.ac.id/index.php/JPAG
6. D. W. Syahputra, B. Rahayudi, and L. Muflikhah, “Analisis Sentimen Twitter terhadap Kebijakan Pemberlakuan Pembatasan Kegiatan Masyarakat menggunakan Metode Support Vector Machine,” *Jurnal Pengembangan Teknologi Informasi dan Ilmu Komputer*, vol. 6, no. 3, pp. 1067–1072, Mar. 2022, [Online]. Available: http://j-ptiik.ub.ac.id`
7. Y. Sibaroni, “Perbandingan Pembobotan Fitur TF-IDF dan TF-ABS Dalam Klasifikasi Berita Online Menggunakan Support Vector Machine (SVM),” *e-Proceeding of Engineering*, vol. 10, no. 3, pp. 3652–3663, Jun. 2023.
8. M. I. Syafaah and L. Lestandy, “Emotional Text Classification Using TF-IDF (Term Frequency-Inverse Document Frequency) And LSTM (Long Short-Term Memory),” *JUITA: Jurnal Informatika*, vol. 10, no. 2, pp. 225–232, Nov. 2022, [Online]. Available: https://atapdata.ai/dataset/192/HIMPUNAN_DATA_E
9. F. Rifaldy, Y. Sibaroni, and S. S. Prasetiyowati, “Effectiveness of Word2Vec and TF-IDF in Sentiment Classification on Online Investment Platforms Using Support Vector Machine,” *JIPI (Jurnal Ilmiah Penelitian dan Pembelajaran Informatika)*, vol. 10, no. 2, pp. 863–874, Mar. 2025, doi: 10.29100/jipi.v10i2.6055.

10. A. H. Dani, E. Y. Puspaningrum, and R. Mumpuni, "Studi Performa TF-IDF dan Word2Vec Pada Analisis Sentimen Cyberbullying," *Jurnal Teknik Informatika dan Terapan*, vol. 2, no. 2, pp. 94–106, Jun. 2024, doi: 10.62951/router.v2i2.76.
11. Z. Zhan, "Comparative Analysis of TF-IDF and Word2Vec in Sentiment Analysis: A Case of Food Reviews," *ITM Web of Conferences*, vol. 70, p. 02013, 2025, doi: 10.1051/itmconf/20257002013.
12. G. R. Ati and P. T. Prasetyaningrum, "Analysis of Community Sentiment Towards Free Nutrition Meal Programs on Twitter Using Naïve Bayes, Support Vector Machine, K-Nearest Neighbors, and Ensemble Methods," *Journal of Information Systems and Informatics*, vol. 7, no. 2, pp. 1443–1460, Jul. 2025, doi: 10.51519/journalisi.v7i2.1098.
13. N. R. Dewi, E. Y. Puspaningrum, and H. Maulana, "Analisis Sentimen Tweet Vaksinasi COVID-19 Menggunakan RNN Dengan Metode TF-IDF Dan Word2Vec," *Jurnal Informatika dan Sistem Informasi (JIFoSI)*, vol. 3, no. 1, pp. 56–65, Apr. 2022.