Segmentation Word to Improve Performance Sentiment Analysis for Indonesian Language

Siti Mujilahwati [1,2], Noor Zuraidin M. Safar [1,*], Catur Supriyanto [3]

[1] Faculty of Computer Science and Information Technology, Universiti Tun Hussein Onn, Malaysia

[2] Informatic Engineering Department, Faculty of Engineering, Universitas Islam Lamongan, East Java, Indonesia

[3] Informatic Engineering Department, Faculty of Computer Science, Universitas Dian Nuswantoro, Semarang, Indonesia

Emails: gi210037@student.uthm.edu.my; zuraidin@uthm.edu.my; catur.supriyanto@dsn.dinus.ac.id

* Correspondence: zuraidin@uthm.edu.my

Abstract

This study explores the enhancement of accuracy in Indonesian sentiment analysis by incorporating text segmentation features during the pre-processing phase. One of the most important steps in creating a high-quality Bag of Words is to separate Indonesian sentences with no spacing, which is made possible by the created text segmentation algorithm. Through the conducted observations and analyses, it was observed that text comments from social media frequently exhibit connected sentences without spacing. The segmentation process was developed through a matching model utilizing a standard Indonesian word dictionary. Implementation involved testing Indonesian text data related to COVID-19 management, resulting in a substantial increase of 3,036 features. The Bag of Words was then constructed using the Term Frequency-Inverse Document Frequency method. Subsequently, sentiment analysis classification testing was conducted using both deep learning and machine learning models to assess data quality and accuracy. The sentiment analysis accuracy for applying Deep Learning, Support Vector Machine and Naive Bayes is 86.46%, 88.02% and 86.19% respectively.

Keywords: Segmentation Text; Sentiment Analysis; Indonesian Language; CNN; SVM; Naïve Bayes.

1. Introduction

The emergence of social media and smartphones has had a significant impact on electronic data. The contemporary landscape is characterized by near-universal smartphone ownership and widespread social media engagement. Social media platforms function as flexible instruments that enable communication, information retrieval, data sharing, commerce, and opinion expression. These activities often revolve around popular subjects [1]. During the onset of the global COVID-19 pandemic in late 2019, many social media users expressed their grievances through personal accounts on these platforms. As the pandemic progressed, discussions about vaccination, mental health, economic recovery, community well-being, and the implementation of distance learning became more prevalent. Given the various sentiments expressed in comments, retrieving valuable insights from these commentaries requires utilizing algorithmic techniques known as text mining via computers. Text mining generates multiple results, including text classification, summarization, document labelling, sentiment analysis, and opinion mining [2], [3]. These methodologies enable a deeper understanding of the wealth of information embedded within the textual content, offering valuable perspectives on user sentiments and preferences.

Since the advent of digital data, considerable research has been conducted on text mining with an emphasis on sentiment analysis extracted from social media [4]–[7], as well as within the Indonesian text context.

Doi: https://doi.org/10.54216/FPA.150213
Received: January 16, 2024 Revised: April 25, 2024 Accepted: May 17, 2024

Several methodologies have been developed and employed to analyse sentiments, including approaches such as support vector machines (SVM) [8], Naive Bayes (NB) [9]–[13], and Deep Learning (DL) [14], [15]. Currently, deep learning is a highly favoured technique for developing models, known for creating classification models with exceptional accuracy while eliminating the need for feature selection steps. The application of deep learning in sentiment analysis research has extended across various domains, such as analysing consumer sentiment in the telecommunications industry [16], evaluating reviews of cosmetic products [17], analysing the sentiment of academic students [18], and assessing sentiment within film reviews [19]. This method has gained popularity due to its capability to yield high-precision classification models without the intricacies associated with traditional feature selection procedures.

Various factors can affect the method's performance beyond its efficacy, with the quality of the data used playing a vital role. Textual data retrieved from social media typically lacks structure, provides less information, and contains numerous numerical elements and punctuation marks. Addressing the challenges posed by textual data processing, the initial stage of pre-processing significantly impacts the accuracy of results [20]–[23]. The principal aim of this study is to enhance the performance of sentiment analysis techniques via optimization of the pre-processing phase.

One crucial aspect of the pre-processing stage is stemming, which involves identifying essential words [24], [25]. For Indonesian text, the Sastrawi library is one of the stem models [25]. However, observations indicate that the Sastrawi library has limitations when processing sentences lacking spaces between words, such as "*bantupulihkanekonomiindonesia*". Sastrawi library is applied in previous research conducted by Prasetyo et al. and Handianti et al. on similar COVID-19 data. Their research showed that SVM and NB methods achieved accuracy rates of 82% [8], 54%, and 53% [13], respectively. Therefore, the aim of this study is to develop a word segmentation model that is suitable for instances where Sastrawi stemming is incapable of handling text data. The suggested method involves segmenting every word in a sentence based on the words listed in the dictionary. The effectiveness of proposed work in this research will be compared against benchmark methods of SVM, NB, and DL. The purpose of this research is to ascertain the degree to which the propose segmentation model enhances sentiment analysis results.

This research comprises four sections in addition to the introductory section. Section two, titled Methods and Materials, present the overview of the research process and explains the propose approach in detail before discussing the results in the third section. The fourth section comprises the conclusions of the study.

2. Methods and Materials

This study was conducted in several stages to attain analysis results from the proposed model. The stages included data preparation, data cleansing, data extraction, data selection, and sentiment analysis classification. Figure 1 illustrates the stages of the experimental design in this research.

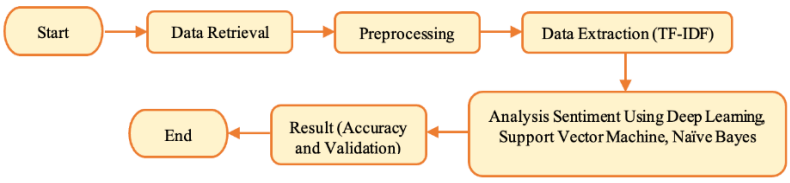


Figure 1: Research Flow

2.1 Data

The research starts with online data collection [26]. Then, a new pre-processing stage was carried out to classify the sentiment analysis. The data used to test the pre-processing combination model in this study were 1918 comments from Twitter, with the topic of dealing with the COVID-19 pandemic that comprises in two data partition that consists a negative class label of 1044 and a positive class label of 874.

The overall process starts from pre-processing, performance testing and visualization in this study by using Python. The tools used are Jupyter notebooks, and some libraries to support this research such as Pandas, NumPy, Sklearn, Scipy, Matplotlib, OS, and JSON. The dictionaries used are standard word dictionaries in Indonesian, stop words, slang dictionaries, and segmentation dictionaries. These tools were used to identify possible words that are connected without spaces.

2.2 Preprocessing

The following stages are carried out in data pre-processing:

Doi: https://doi.org/10.54216/FPA.150213
Received: January 16, 2024 Revised: April 25, 2024 Accepted: May 17, 2024

1) Case Folding: Changing all letters to lowercase. The results of this process can be shown in Table 1 below.

Table 1. Case Folding

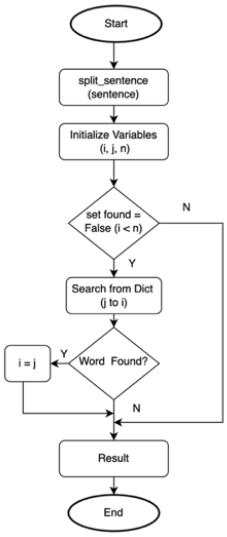
Text	Result
Yukk kawal Kebijakan Pemerintah jangan Sampe didalah Gunakan Oleh OKNUM	*yukk kawal kebijakan pemerintah jangan sampe didalah gunakan oleh oknum*

2) Cleansing : Cleansing by removing non-ASCII, URLs, Mentions, Hashtags, Symbols, and numbers and correcting duplication of characters.

Table 2: Cleansing

Text	Result
*Yg harus dikerjakan oleh Pemerintah..selain mengobati yg terkena Covid-19, juga Mencari para ahli ilmuwan Bikin vaksin. Dan kehidupan rakyat yg kena dampak..program lockdown. Secara palarel mengocorkan bantuan untuk slm di lockdown. Bgtulah kira" *Dari Rakyat untuk rakyat.**	*yg harus dikerjakan oleh pemerintah selain mengobati yg terkena covid juga mencari para ahli ilmuwan bikin vaksin dan kehidupan rakyat yg kena dampak program lockdown secara palarel mengocorkan bantuan untuk slm di lockdown bgtulah kira dari rakyat untuk rakyat*

3) Word Segmentation: According to the basic ideas algorithm from this study, the first step in preparation for segmentation is to gather dictionary data that includes basic words, standard words, hyphens, days of the week, city names, and states. The dictionary used is in the form of text, so the list can be added to as necessary. Text data will be split or tokenized and each word or term will be matched with the. If the word is in the dictionary, it will be separated with spaces. If it does not exist, then the word is considered correct and stored in a new collection. Figure 2 shows the design of the created segmentation algorithm.




```
Pre-Processing Text Algorithm
Function ProcessText(text, dictionary):
    i ← 0;
    n ← length of text;
    while i < n do
        found ← FALSE;
        for j ← n to i do
            if text[itoj] is in dictionary then
                Add text[itoj] to separate sentence;
                i ← j + 1;
                found ← TRUE;
                break;
            end
        end
        if not found then
            if last substring in separate sentence does not end with a
              space then
                Merge last substring with current character;
            end
            i ← i + 1;
        end
    end
```

Figure 2: Word Segmentation Flowchart and Algorithm

The result is text separated by words in each dictionary in Table 3.

Table 3: Segmentation

Doi: https://doi.org/10.54216/FPA.150213
Received: January 16, 2024 Revised: April 25, 2024 Accepted: May 17, 2024

Text	Result
▪ *antupulihkanekonomi*	▪ *antu pulihkan ekonomi*
▪ *emerintahindonesia*	▪ *emerintah indonesia*
▪ *ukungjokowibasmicorona*	▪ *ukung jokowi basmi corona*
▪ *ulihdengancepat*	▪ *ulih dengan cepat*
▪ *agajarakaman*	▪ *aga jarak aman*
▪ *anganlupacucitangan*	▪ *angan lupa cuci tangan*

4) Stop Words : Stop Words are common words that appear frequently in text and tend to have little informational value in text analysis [2], [27]. In Natural Language Processing (NLP), stop words are usually removed or ignored when performing text analysis, as these words do not make a significant contribution to understanding the context or meaning of the text. Examples of stop words in Indonesian include “yang”, ”dan”, “di”, “dari”, “dengan”, “itu”, “ini”, “pada”, “untuk, and “adalah. Table 4 shows Stop Words results.

Table 4: Stop Words

Text	Result
*semua negara **di** dunia sdng berjuang melawan covid pemerintah indonesia sdng brjuang **dan** kerja keras asal kita brsatu patuh taat **dan** percaya krn **itu yg** dibutuhkan pemerintah jgn percaya hoax **dan** cek sblm share kita yakin indonesia akan melewati nya dukung jokowi basmi corona*	*semua negara dunia sdng berjuang melawan covid pemerintah indonesia sdng brjuang kerja keras asal kita brsatu patuh taat percaya krn dibutuhkan pemerintah jgn percaya hoax cek sblm share kita yakin indonesia melewati dukung jokowi basmi corona*

2.3 Word-Inverse Frequency Data Extraction

Document Word-Inverse Frequency (TF-IDF) method is used to fine the word relationships within documents [2], [28]. This method involves two approaches to weight calculation: Term Frequency (TF) and Document Frequency (DF). TF refers to the frequency of the word (*t*) appearing in a sentence (*d*), while DF refers to the number of sentences where the word (*t*) appears. The TF-IDF formula is:

$$W_{t,d} = tf_{t,d} * log\left(\frac{N}{df_t}\right) \quad (1)$$

Where $W_{t,d}$ is the weight of the word (term) t in the document d. This weight attempts to measure how important the word is in the context of the document. $tf_{t,d}$ is the frequency of the word t in the document d, that is, how often it appears in the document. N is the total number of documents in the collection. It reflects the total size of the document collection used in the analysis. df_t is the number of times the document says T in the entire document collection. It measures how many documents in the collection contain the word. $log\left(\frac{N}{df_t}\right)$ is the logarithm of the ratio between the total number of documents in the collection (N) and the frequency of documents of the word t (df_t). This section is used to lower the weight of words that appear in many documents because they are usually considered less informative in the context of analysis.

The formula facilitates an accurate evaluation of the significance of a word in a document by analyzing its frequency within the document as well as in other documents from the same collection. This fundamental technique is crucial for information retrieval and understanding the contextual relevance of words [29], [30].

2.4 Sentiment Analysis Using Deep Learning

DL for NLP is a neural network framework that captures the meaning and structure of text. DL uses convolution filters to analyze word sequences [31][32] and identify important features like phrases, word patterns, and sentence context. Convolution layers will construct the abstract hierarchical representations based on these features that can be applied to tasks such as text classification, sentiment analysis, and language comprehension. Technical term abbreviations will be explained upon first usage.

Doi: https://doi.org/10.54216/FPA.150213
Received: January 16, 2024 Revised: April 25, 2024 Accepted: May 17, 2024

The design of a DL for NLP may vary depending on the problem. However, the architecture for a DL neural network model in NLP [3][33] involves several crucial components such as:

- Input Layer: Accept sequences of words or characters in the form of tokens. It may require embedding layers to convert tokens into numerical vector representations. In this study, a TF-IDF vectorizer was used.
- Convolutional Layers: Run convolution filters to extract features from word or character sequences. Each filter can detect specific patterns; such as phrases or context. The model has several convolution layers with different filters. This study used the Conv1D design.
- Pooling Layers: Reduce the output dimension of the convolution layer. Max pooling or average pooling is often used to retrieve the most important features of any convolution result window. This study used Max pooling.
- Output Layer: Depending on the task, it can be either a layer with a single neuron for regression or a layer with neurons according to the number of classes for classification. This study uses Dense Layer output 2, which is with a value of 0.1 representing (Negative and Positive)
- Activation Functions: Each layer is usually followed by an activation function such as ReLU (Rectified Linear Activation) to introduce non-linearity.
- Padding: The use of padding can ensure that the output of the convolution layer has the appropriate dimensions.
- Regularization: Use techniques such as dropout or batch normalization to prevent overfitting.
- Optimization Algorithm: Choose an optimizer like SGD (Stochastic Gradient Descent) or Adam.
- Loss Function:This study used categorical cross-entropy for classification.

2.5 Performance Evaluation

The proposed technique is evaluated based on accuracy, precision, recall, and F1 score. The evaluation parameter is given below:

$$\text{Accuracy} = \frac{T_p + T_n}{T_p + T_n + F_p + F_n} \tag{3}$$

$$\text{Precision} = \frac{T_p}{T_p + F_p} \tag{4}$$

$$\text{Recall} = \frac{T_p}{T_p + T_n} \tag{5}$$

$$\text{F1} = 2 * \frac{\text{Precision} * \text{Recall}}{\text{Precision} + \text{Recall}} \tag{6}$$

3. Result and Discussion

3.1 Analysis of Results Word Segmentation Features in the Pre-processing

The effectiveness of the word segmentation algorithm is indicated by the comparison between the number of extraction features before and after segmentation. The utilization of segment features generates more comprehensive data compared to non-utilization. The data analysis of feature extraction is presented in Table 5.

Table 5: Analysis of Text Segmentation Results

Label Class	Count Features without Segmentation Text	Count Features using Segmentation Text
Positive	137.808	138.420
Negative	158.933	161.357
Total Count	296.741	299.777
Count Dataset	1.918	1.918

Table 5 shows the difference in the number of features before and after the word segmentation process. from a dataset of 1.918, without using the words segmentation process the results of the features is 296.741,

Doi: https://doi.org/10.54216/FPA.150213
Received: January 16, 2024 Revised: April 25, 2024 Accepted: May 17, 2024

while after adding the words segmentation process the features obtained are 299.777. There is an increase in the number of features by 3.036.

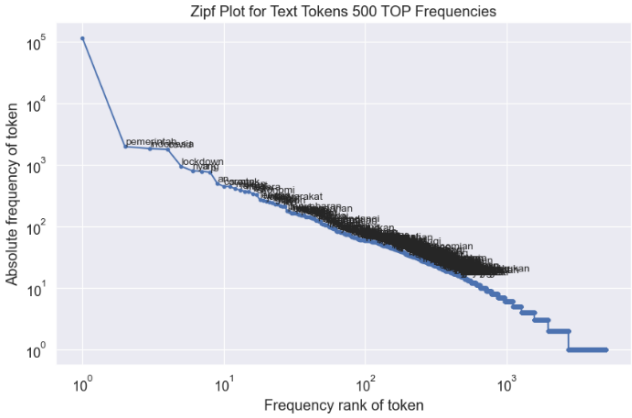


Figure 3: Frequency Top Tokens

Figure 3 shows the frequency distribution of words, and identifying the most frequent keywords helps to evaluate their importance in text analysis. A steeper *Zipf* line indicates a large number of frequent words in the text. The graph shows the top 100 words with the highest number of occurrences and the top 500 words with the highest frequencies.

3.2 TF-IDF Extraction Results

During this stage, the machine reads the annotation data by converting it from text to numbers through data extraction using TF-IDF [34]. TF-IDF is a method used to transform text into numerical values. This is necessary because computational intelligent approaches such as machine learning and deep learning models cannot analyze input data in the form of text or strings, requiring numerical input [35]. Figure 4 shows the process of data extraction using the TF-IDF method.

term	TF	TF-IDF
wabah	0.06666666666666667	0.14934765081783918
covid	0.06666666666666667	0.022609938900525508
sugesti	0.06666666666666667	0.4611803739890203
warga	0.06666666666666667	0.17939145763780837
indonesia	0.06666666666666667	0.02002017545738853
takut	0.13333333333333333	0.7077690263201607
perintah	0.06666666666666667	0.013205918197912503
media	0.06666666666666667	0.2955199306698203
over	0.06666666666666667	0.40009432519741
memberitahukan	0.06666666666666667	0.4611803739890203
rakyat	0.06666666666666667	0.12603606758585592
virus	0.06666666666666667	0.12691616837642497
corona	0.06666666666666667	0.14036808363085918
nakuti	0.06666666666666667	0.4611803739890203

Figure 4: TF-IDF Extraction Results

3.3 Performance evaluated on Sentiment Analysis

The word segmentation results show that this process is successful in increasing the number of features. To evaluate the effectiveness of this feature enhancement, we conducted tests using sentiment analysis classification. The benchmark for word segmentation performance is the increasing accuracy value produced in sentiment analysis. In this research, testing will be carried out using three methods, including DL, SVM, and NB. Table 6 displays the results obtained by the DL model, and Table 7 displays the results obtained by the SVM and NB methods.

Table 6: Result of sentiment analysis testing using DL

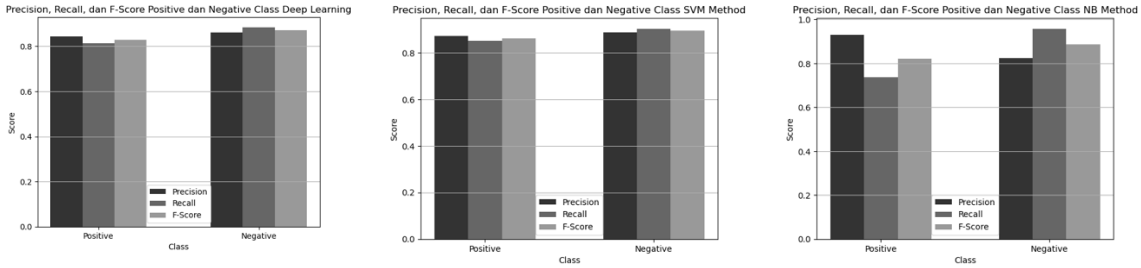
Train:Test	Epoch	Accuracy	Recall	F Score	Precision
80:20	10	86.46	86	86	86
70:30	10	84.11	84	84	84
60:40	10	85.16	85	85	85

Table 7: Result of sentiment analysis testing using SVM and NB

Train:Test	Method	Accuracy	Recall	F Score	Precision
80:20	SVM	88.02	88	88	88
80:20	NB	86.19	85	85	88

Doi: https://doi.org/10.54216/FPA.150213
Received: January 16, 2024 Revised: April 25, 2024 Accepted: May 17, 2024

Integrating word segmentation algorithms proved to be an effective method for pre-processing the extracted results data for sentiment classification, as concluded by the study. Remarkable accuracy rates were obtained through the application of three different methods, namely DL (86.46%), SVM (88.02%), and NB (86.19%). Tables 6 and 7, alongside Figure 5, provide detailed information on the outcomes of recall, precision, and f-score. Moreover, Figure 6 depicts graphical representations of accuracy for both training and validation, as well as validation loss.



(a) (b) (c)

Figure 5: Result Evaluated (a) Deep Learning Method, (b) SVM Method, (c) Naïve Bayes Method

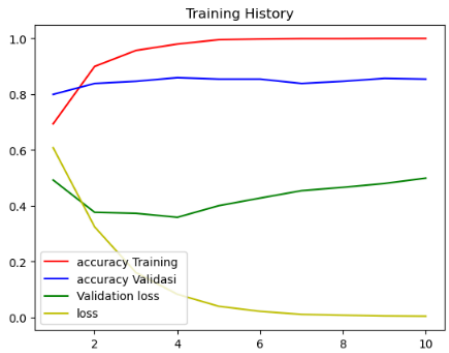
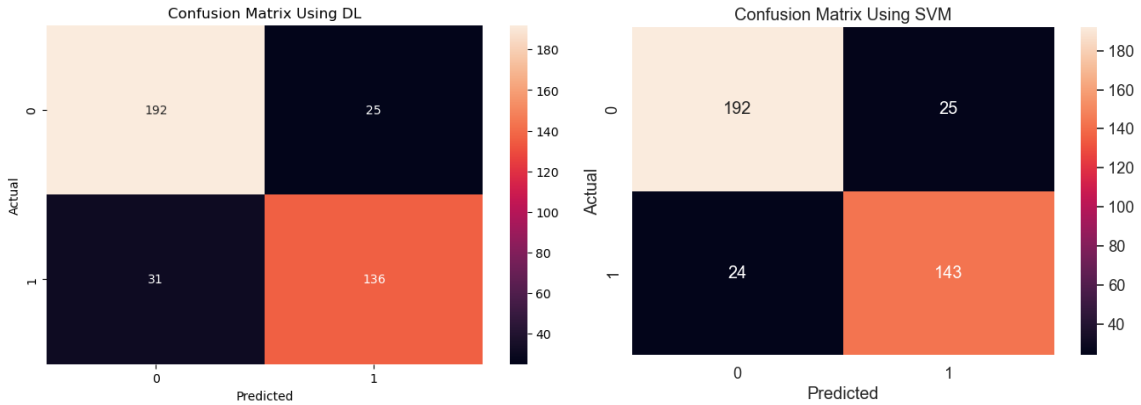


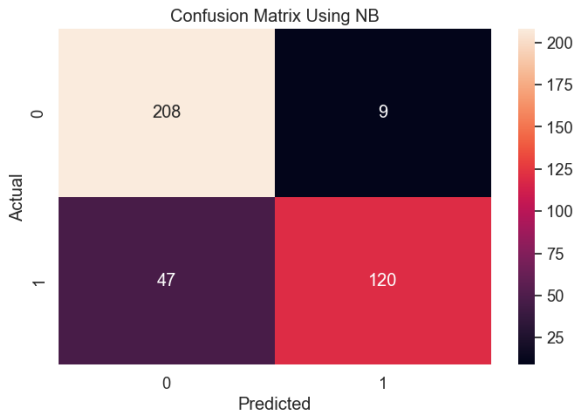
Figure 6: Accuracy Training dan Loss Validation

The results of the Confusion Matrix sentiment analysis using DL, SVM and Naïve Bayes Methods are shown in Figure 7.



(a) (b)

Doi: https://doi.org/10.54216/FPA.150213
Received: January 16, 2024 Revised: April 25, 2024 Accepted: May 17, 2024



(c)

Figure 7: Confusion Matrix Analysis Sentiment Using (a) Deep Learning Method, (b) SVM Method, (c) Naïve Bayes Method

The sentiment analysis results for the test data showed 217 instances of negative class and 167 instances of positive class. The deep learning model, equipped with previous segmentation features during pre-processing, attained an accuracy of 86.46%. The model predicted negative sentiment value with 192 instances of correct classification and 25 false classifications. Although the sentiment for the positive class was confirmed, the accuracy of 136 true and 31 false classifications did not align with the class. Although SVM and NB methods produce varying results for each class, SVM outperforms NB in the negative class. Out of 167 test data, SVM accurately predicts 143 while NB correctly predicts 120.

3.4 Comparing with previous study

The results of the research have a better level of accuracy compared to previous researchers [8], [13]. The following is a comparison table of the proposed model with previous research.

Table 8: Comparing Result of Sentiment Analysis Testing

Author	Model	Accuracy %
Proposed Model	SVM + Segmentation Words	88.02
	NB + Segmentation Words	86.20
	Deep Learning + Segmentation Words	86.46
	Deep Learning	84.37
Prasetyo, et. al[8]	SVM	82
Hadianti, et. al[13]	SVM	54
Hadianti, et. al[13]	NB	53

The DL model showed an increase in accuracy of 2.09% with and without word segmentation. The SVM + word segmentation method achieved a 34% improvement in accuracy from previous research [8][13]. Additionally, the NB + word segmentation approach yielded a 33% increase in accuracy compared with previous research [13].

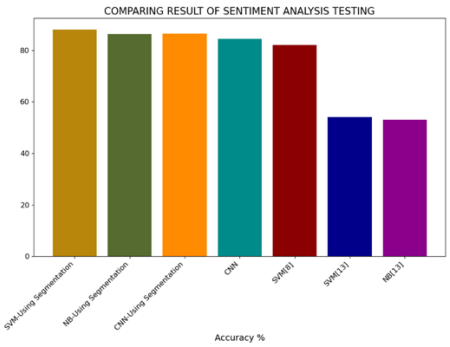


Figure 8: Comparing Result of Sentiment Analysis Test

Doi: https://doi.org/10.54216/FPA.150213
Received: January 16, 2024 Revised: April 25, 2024 Accepted: May 17, 2024

The study revealed that the SVM method outperformed both Naive Bayes and deep learning models in sentiment analysis.

4. Conclusion

The designed and implemented word segmentation process has shown excellent performance. This can be seen from the feature extraction results which increased by 3,036 features after the implementation of word segmentation. This research uses 1-layer Conv1D and Max pooling parameters for 1-dimensional Deep Learning Neural Network. The activation function used is soft max, and the optimizer is Adam with a batch size of 64 and 10 epochs. This configuration resulted in 86.46% accuracy, 86% recall, 86% F-score, and 86% precision. In addition, this study also evaluated the SVM and Naive Bayes approaches, with accuracy rates of 88.02% and 86.20%, respectively. So it can be concluded that segmenting words can improve the performance of sentiment analysis in Indonesian.

Conflicts of Interest: "The authors declare no conflict of interest."

References

[1] D. Chaffey, "Global social media statistics research summary 2023," https://www.smartinsights.com/social-media-marketing/social-media-strategy/new-global-social-media-research/.

[2] R. Feldman and J. Sanger, *The Text Mining Handbook*. Cambridge: Cambridge University Press, 2006. doi: 10.1017/cbo9780511546914.

[3] "Mining Text Data."

[4] F. A. Pozzi, E. Fersini, E. Messina, and B. Liu, *Sentiment Analysis in Social Networks*. 2016.

[5] Matthew A.Russell, "Mining the Social Web: Analyzing Data from Facebook," *Twitter, LinkedIn, and Other Social Media Sites*, p. 428, 2019.

[6] F. A. Nugraha, N. H. Harani, R. Habibi, and Rd. N. S. Fatonah, "Sentiment Analysis on Social Distancing and Physical Distancing on Twitter Social Media using Recurrent Neural Network (RNN) Algorithm," *Jurnal Online Informatika*, vol. 5, no. 2, 2020, doi: 10.15575/join.v5i2.632.

[7] S. Makinist, İ. R. Hallaç, B. Ay Karakuş, and G. Aydın, "Preparation of Improved Turkish DataSet for Sentiment Analysis in Social Media," *ITM Web of Conferences*, vol. 13, 2017, doi: 10.1051/itmconf/20171301030.

[8] P. H. Prastyo, A. S. Sumi, A. W. Dian, and A. E. Permanasari, "Tweets Responding to the Indonesian Government's Handling of COVID-19: Sentiment Analysis Using SVM with Normalized Poly Kernel," *Journal of Information Systems Engineering and Business Intelligence*, vol. 6, no. 2, 2020, doi: 10.20473/jisebi.6.2.112-122.

[9] M. B. Ressan and R. F. Hassan, "Naïve-Bayes family for sentiment analysis during COVID-19 pandemic and classification tweets," *Indonesian Journal of Electrical Engineering and Computer Science*, vol. 28, no. 1, 2022, doi: 10.11591/ijeecs.v28.i1.pp375-383.

[10] P. Arsi, B. A. Kusuma, and A. Nurhakim, "Analisis Sentimen Pindah Ibu Kota Berbasis Naive Bayes Classifier," *Jurnal Informatika Upgris*, vol. 7, no. 1, 2021, doi: 10.26877/jiu.v7i1.7636.

[11] A. Perdana, A. Hermawan, and D. Avianto, "Analisis Sentimen Terhadap Isu Penundaan Pemilu di Twitter Menggunakan Naive Bayes Clasifier," *Jurnal Sisfokom (Sistem Informasi dan Komputer)*, vol. 11, no. 2, pp. 195–200, Jul. 2022, doi: 10.32736/sisfokom.v11i2.1412.

[12] A. Erfina and M. Rifki Nurul, "Implementation of Naive Bayes classification algorithm for Twitter user sentiment analysis on ChatGPT using Python programming language," *Data & Metadata*, vol. 2, p. 45, Jun. 2023, doi: 10.56294/dm202345.

[13] S. Hadianti *et al.*, "ANALISIS SENTIMENT COVID-19 DI TWITTER MENGGUNAKAN METODE NAIVE BAYES DAN SVM," *Jurnal Teknologi Informasi)*, vol. 6, no. 1, [Online]. Available: www.Kaggle.com.

[14] P. Cen, K. Zhang, and D. Zheng, "Sentiment Analysis Using Deep Learning Approach," vol. 2, no. 1, pp. 17–27, 2020, doi: 10.32604/jai.2020.010132.

[15] D. Tang *et al.*, "Sentiment analysis using deep learning architectures: a review," *Artif Intell Rev*, vol. 9, no. 2, pp. 4335–4385, 2020, doi: 10.1007/s10462-019-09794-5.

[16] Z. Amalia, M. Irfan, D. S. A. Maylawati, A. Wahana, W. B. Zulfikar, and M. A. Ramdhani, "Sentiment Analysis of the Use of Telecommunication Providers on Twitter Social Media using Convolutional Neural Network," in *2022 IEEE 8th International Conference on Computing, Engineering and Design, ICCED 2022*, 2022. doi: 10.1109/ICCED56140.2022.10010357.

Doi: https://doi.org/10.54216/FPA.150213
Received: January 16, 2024 Revised: April 25, 2024 Accepted: May 17, 2024

[17] E. Y. Hidayat and D. Handayani, "Penerapan 1D-CNN untuk Analisis Sentimen Ulasan Produk Kosmetik Berdasar Female Daily Review," *Jurnal Nasional Teknologi dan Sistem Informasi*, vol. 8, no. 3, pp. 153–163, Jan. 2023, doi: 10.25077/teknosi.v8i3.2022.153-163.
[18] A. Yunita, H. B. Santoso, and Z. A. Hasibuan, "Deep Learning for Predicting Students' Academic Performance," *Proceedings of 2019 4th International Conference on Informatics and Computing, ICIC 2019*, p. 8985721, Oct. 2019, doi: 10.1109/ICIC47613.2019.8985721.
[19] R. Ganda and A. Mahmood, "Deep Learning approach for sentiment analysis of short texts," no. February 2018, 2017, doi: 10.1109/ICCAR.2017.7942788.
[20] "The effects of Pre-Processing Techniques on Arabic Text Classification," *International Journal of Advanced Trends in Computer Science and Engineering*, vol. 10, no. 1, 2021, doi: 10.30534/ijatcse/2021/061012021.
[21] R. Duwairi and M. El-Orfali, "A study of the effects of preprocessing strategies on sentiment analysis for Arabic text," *J Inf Sci*, vol. 40, no. 4, pp. 501–513, 2014, doi: 10.1177/0165551514534143.
[22] H. M. Zin, N. Mustapha, M. A. A. Murad, and N. M. Sharef, "The effects of pre-processing strategies in sentiment analysis of online movie reviews," *AIP Conf Proc*, vol. 1891, no. October 2017, 2017, doi: 10.1063/1.5005422.
[23] Y. S. Mehanna and M. Mahmuddin, "The Effect of Pre-processing Techniques on the Accuracy of Sentiment Analysis Using Bag-of-Concepts Text Representation," *SN Comput Sci*, vol. 2, no. 4, 2021, doi: 10.1007/s42979-021-00453-7.
[24] M. U. Albab, Y. Karuniawati P, and M. N. Fawaiq, "Optimization of the Stemming Technique on Text preprocessing President 3 Periods Topic," vol. 20, no. 2, pp. 1–10, 2023, doi: 10.26623/transformatika.v20i2.5374.
[25] Rianto, A. B. Mutiara, E. P. Wibowo, and P. I. Santosa, "Improving the accuracy of text classification using stemming method, a case of non-formal Indonesian conversation," *J Big Data*, vol. 8, no. 1, 2021, doi: 10.1186/s40537-021-00413-1.
[26] Lamia Mohamed Ahmed, Gawaher Soliman Hussein, Abdel Nasser Hessin Zaied, A Survey on Sentiment Analysis Algorithms and Techniques For Arabic Textual Data, Journal of Fusion: Practice and Applications, Vol. 2 , No. 2 , (2020) : 74-87 (Doi : https://doi.org/10.54216/FPA.020205)
[27] R. Feldman and J. Sanger, "Text Mining Preprocessing Techniques," in *The Text Mining Handbook*, Cambridge University Press, 2006, pp. 57–63. doi: 10.1017/CBO9780511546914.004.
[28] D. Rao and B. Mcmahan, "Natural Language Processing with PyTorch Build Intelligent Language Applications Using Deep Learning," 2019.
[29] A. S. Alammary, "Arabic Questions Classification Using Modified TF-IDF," *IEEE Access*, vol. 9, 2021, doi: 10.1109/ACCESS.2021.3094115.
[30] Vijay K, Collaborating The Textual Reviews Of The Merchandise and Foretelling The Rating Supported Social Sentiment, Journal of Journal of Cognitive Human-Computer Interaction, Vol. 1 , No. 2 , (2021) : 63 - 72 (Doi : DOI: https://doi.org/10.54216/JCHCI.010203)
[31] Moch. A. Nasichuddin, T. B. Adji, and W. Widyawan, "Performance Improvement Using CNN for Sentiment Analysis," *IJITEE (International Journal of Information Technology and Electrical Engineering)*, vol. 2, no. 1, 2018, doi: 10.22146/ijitee.36642.
[32] Praloy Biswas, A. Daniel, Subhrendu Guha Neogi, Spider Monkey Optimization with Deep Learning-based Hindi Short Text Sentiment Analysis, Journal of Journal of Intelligent Systems and Internet of Things, Vol. 12 , No. 1 , (2024) : 97-109 (Doi : https://doi.org/10.54216/JISIoT.120108)
[33] M. M. Khalid, & O. Karan, Deep Learning for Plant Disease Detection. International Journal of Mathematics, Statistics, and Computer Science, 2023, v. 2, 75–84.
[34] D. A. Prabowo, M. Fhadli, M. A. Najib, H. A. Fauzi, and I. Cholissodin, "TF-IDF-Enhanced Genetic Algorithm Untuk Extractive Automatic Text Summarization," *Jurnal Teknologi Informasi dan Ilmu Komputer*, vol. 3, no. 3, 2016, doi: 10.25126/jtiik.201633217.
[35] M. Chiny, M. Chihab, Y. Chihab, and O. Bencharef, "LSTM, VADER and TF-IDF based Hybrid Sentiment Analysis Model," *International Journal of Advanced Computer Science and Applications*, vol. 12, no. 7, 2021, doi: 10.14569/IJACSA.2021.0120730.

Doi: https://doi.org/10.54216/FPA.150213
Received: January 16, 2024 Revised: April 25, 2024 Accepted: May 17, 2024