JURNAL RISET INFORMATIKA **P-ISSN: 2656-1743 |E-ISSN: 2656-1735**
Vol. 5, No. 1. December 2022 DOI: https://doi.org/10.34288/jri. v5i1.169
Accredited rank 4 (SINTA 4), excerpts from the decision of the DITJEN DIKTIRISTEK No. 230/E/KPT/2023

Sentiment Analysis of Twitter's Opinion on The Russia and Ukraine War Using Bert

Muhammad Fahmi Julianto[-1], Yesni Malau[-2*)], Wahyutama Fitri Hidayat[-3]

Teknik Informatika Kampus Kota Pontianak[1], Sistem Informasi Kampus Kota Pontianak[3]
Universitas Bina Sarana Informatika
fahmi.fjl@bsi.ac.id, wahyutama.wfh@bsi.ac.id

Teknik Elektro[2]
Universitas Bina Sarana Informatika
[2*)] yesni.ymu@bsi.ac.id

(*) Corresponding Author

Abstract

News about the war between Russia and Ukraine can not be denied affecting various aspects of life worldwide. It affects the writings of every world citizen on various social media platforms, one of which is Twitter. Sentiment analysis is a process of identifying and making sentiment categories computationally. The sentiment analysis process is also intended to make computers understand the meaning of human sentences by processing algorithms. This research uses the deep learning method of the BERT (Bidirectional Encoder Representation Form Transform) model language to analyze the sentiments in the tweets written about the wars between Russia and Ukraine by Twitter social media users. The sentiment will be divided into positive, neutral, and hostile. The hyperparameters in this study used ten epochs, with a learning rate of 2e-5 and a batch size of 16. The test used in sentiment analysis was the BERTbase Multilingual-cased-model model, and the accuracy was 97%. Suggestions for further research are the need for a more balanced dataset between positive, neutral, and negative sentiments. They reward the dataset before training so that better results are expected.

Keywords: BERT; Russia-Ukraine; Sentiment Analysis; War

Abstrak

Berita mengenai perang yang terjadi antara Rusia dan Ukraina tidak dapat dipungkiri mempengaruhi berbagai aspek kehidupan di dunia. Hal tersebut mempengaruhi tulisan setiap warga dunia pada berbagai platform media sosial salah satunya twitter. Analisis sentimen merupakan proses identifikasi serta membuat kategori sentimen yang dilakukan secara komputasi. Proses Analisis sentiment dimaksudkan juga untuk membuat komputer memahami arti dari kalimat yang dituliskan oleh manusia dengan pemrosesan menggunakan algoritma. penelitian ini digunakan metode deep learning Bahasa model BERT (Bidirectional Encoder Representation form Transform) sebagai proses analisa sentimen yang ada pada twet yang dituliskan mengenai perang Rusia dan Ukraina oleh pengguna media sosial twetter. Sentimen akan dibagi kedalam tiga bagian yaitu positif, netral, serta negatif. Hyperparameters pada penelitian ini menggunakan 10 epoch, dengan learning rate 2e-5, serta batch size 16. Pengujian yang digunakan dalam analisis sentimen adalah model BERTbase Multilingual-cased-model serta hasil akurasi sebesar 97%. Saran untuk penelitian selanjutnya diperlukannya dataset yang lebih seimbang antara sentimen Positif, netral dan negatif. Melakukan imbalancing dataset sebelum melakukan training sehingga diharapkannya hasil yang lebih baik.

Kata kunci: BERT; Rusia-Ukraina; Analisis Sentimen; Perang

INTRODUCTION

Conflicts inevitably occur in everyday life, big or small. Many things can cause conflict, but conflict often arises because of differences in interests and can also be caused by domination and the desire to dominate.

The conflict between Russia and Ukraine occurred over Crimea, Eastern Ukraine. Crimea itself has been a struggle for centuries. It looks at its history. Crimea was formerly known as Tauris or as

Tavrida. By Russian society, the region is considered home to various ethnic groups. However, the area began to be contested after the Cimmerian tribes invaded Tauris, and this action also triggered Greek colonists to enter the Tauris region in the 6th century BC (BC)(Sudiq & Yustitianingtyas, 2022).

Russia's invasion began on February 24, 2022, to defend the country's security from the threat of Ukraine. Russia is trying to limit Ukraine's proximity to the European Union and NATO to safeguard its security from the threat of the western bloc and influence from the United States and maintain Russia's close relations with the former Soviet Union. It is related to the position of the territory of the Ukrainian state, which is directly adjacent to Russia, so if Ukraine joins NATO, there will no longer be a barrier between Russia and NATO, which is a threat to Russia. That is why Russia invaded Ukraine so that Ukraine could not contact NATO. Russia preventing Ukraine from joining NATO is an attitude that should not be done because it leads to intervention. As a sovereign country, Ukraine should have the freedom to make decisions about its membership in NATO, and Russia should not be able to affect that freedom with physical pressure in the form of an invasion (Sudiq & Yustitianingtyas, 2022).

The incident caused much discussion in the community. The response is widely expressed on social media. One of the social media used to conduct these conversations is Twitter. Twitter is considered suitable for conducting a discussion, we are on Twitter, and people can express their responses in the form of threads or reply to each other's tweets from other accounts.

In previous research related to this, (Andraini & Mahdiyah, 2022) researched Twitter users regarding the Russia-Ukraine War using the SVM method with various optimizations, including cross-validation. It split validation resulting in an accuracy of 86.0%. Meanwhile, the research (Aditya & Wibowo, 2022) uses the Naive Bayes method, and the sentiment analysis system is divided into 5 (five) parts, namely crawling, preprocessing, word weighting, model formation, and sentiment classification. Existing data will be grouped into 2 (two) categories, negative and positive, resulting in an accuracy value of 78.2%.

Based on this background, the author will use a deep learning method using a language model and bidirectional encoder representation Form transformers (BERT) to analyze the sentiments of Twitter users regarding the Russia-Ukraine war writings. The sentiments will be divided into positive, Neutral, and Negative.

RESEARCH METHODS

The method used in the dataset taken from Twitter this time uses the Bidirectional Encoder Representations from Transform method and has the following steps, as shown in Figure 1.

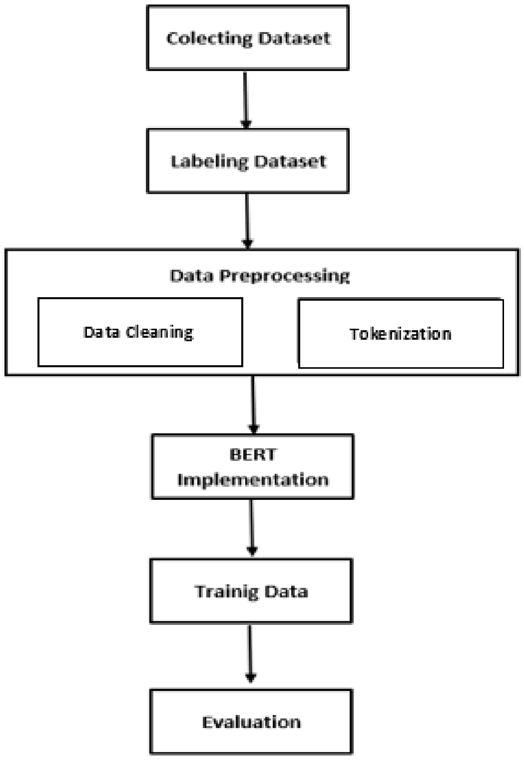


Figure 1. General Architecture

Sentiment analysis begins with collecting datasets, where the datasets are taken on the kaggle.com page. Then, the dataset will be labeled with positive, neutral, and negative names. Then after being labeled, it will enter the next stage, namely preprocessing the dataset, where this stage is the stage to prepare the dataset will later be processed by doing data cleaning and tokenization. The dataset that has gone through this process will then be trained to get a positive, neutral, and negative classification using the BERT model. The classification results will then be evaluated to obtain results.

Collecting dataset data is the least paid attention component of the ML process (Wiegreffe & Marasović, 2021). Selection of optimal features in determining sentiment where the content in the form of online text greatly influences the superior classification results. Selection of optimal features can also affect computational performance, which is very difficult and causes the need to design new

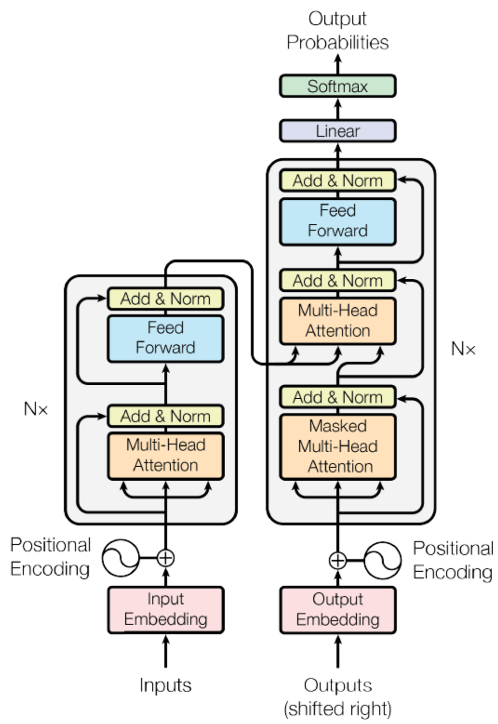
P-ISSN: 2656-1743 |E-ISSN: 2656-1735
DOI: https://doi.org/10.34288/jri. v5i1.169
Accredited rank 4 (SINTA 4), excerpts from the decision of the DITJEN DIKTIRISTEK No. 230/E/KPT/2023

techniques to improve classification performance (Kumar, Jaiswal, Garg, Verma, & Kumar, 2022). The dataset collected by Kaggle was adopted in this study to create a collection of tweets that match the discussion material.

Dataset labeling marks specific texts or documents with appropriate labels, known as text categorization or classification (El Rifai, Al Qadi, & Elnagar, 2022). At this stage, text lines are extracted as an integration process of the pre-classification method with the segmentation and deep-learning methods so that each text is labeled with a label.

The next stage is data preprocessing, a crucial step before classification (Yutika, Adiwijaya, & Faraby, 2021). There are two preprocessing stages, namely data cleaning, and tokenization, as shown in Figure 1. Data cleaning can also be referred to as scrubbing, which functions as detection and removes incorrect or inconsistent words to improve the data quality (Lomet, 2001). Raw data requires processing first, which serves to improve data quality (Fatima, Nazir, & Khan, 2017). Tokenization is the process of separating the upward flow of text into parts, symbols, or other meaningful elements, while the goal of tokenization is the exploration of words in a sentence.

BERT was first introduced in 2018. BERT is a representation of the word bidirectional. The previous model only applies a one-way representation by looking at the word order from right to left or a mixture of right to left and left to right (Devlin et al., 2019). As the name implies, BERT uses to transform. Transform is a mechanism to determine the contextual relationship between several words in a text (Suresha, 2020). Transform can understand and transform the understanding obtained through a mechanism known as the self-attention mechanism. Transform's a way of translating the "understanding" of several related words into several words that are processed in the mechanism. The transformation has two mechanisms, namely, Encoder and the Decoder. An illustration of the Encoder and Decoder can be seen in Figure 2.

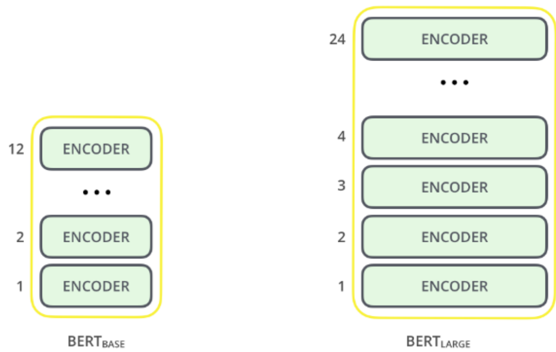


Source : (Vaswani et al., 2017)

Figure 2. Encoder (left) and Decoder (Right)

The steps involved in the encoder and decoder are:

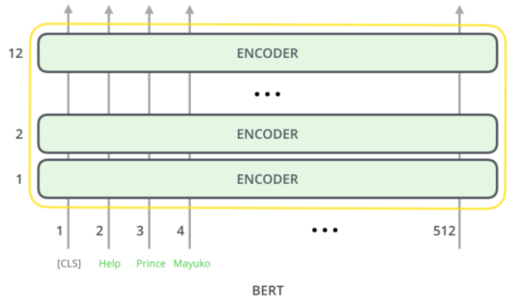
1. Every input word that goes to the *encoder is* converted into a vector list using embedding.
2. *Vector input* through the two existing layers in all encoders, the self-attention layer, and the forward-looking neural network.
3. When every process in the *encoder is* complete, the encoder output, key, and vector value are sent to the decoder. From the explanation of points one to point three, it can be illustrated based on Figure 3.



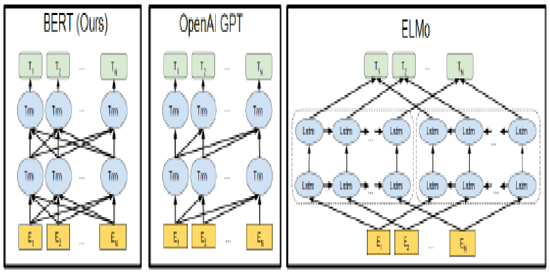
Source : (Alammar, 2019)

Figure 3. Differences in the size of BERTBASE and BERTLARGE

As the name suggests, BERT uses only one encoder. Therefore, the BERT architecture is similar to Figure 4. BERT differs from directional models in that it displays text strings in right-to-left, left-to-right, or right-to-left and left-to-right combinations. The model used in the two-way training language can be used for understanding in context so that it can be more profound than the one-dimensional language model. Figure 5 shows a comparison of the BERT architecture using OpenAIGPT and ELMo. Of the three architectural models, only BERT controls each level's left and proper context.



Source : (Alammar, 2019)
Figure 4 BERT Architecture

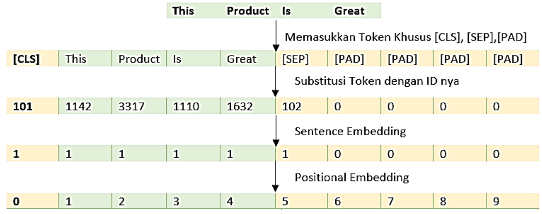


Source : (Devlin, Chang, Lee, & Toutanova, 2019).
Figure 5. Differences in the Architecture of BERT, OpenAI GP, and ELMO

Formula :

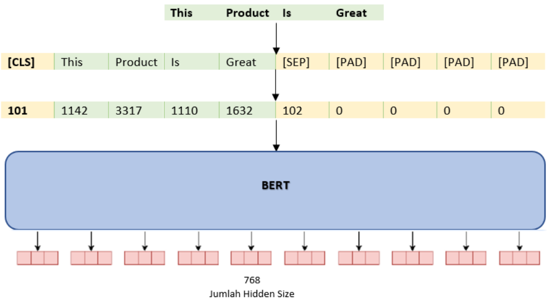
$$f_{baud} = \frac{2^{SMOD}}{64} x f_{osc} \quad (1)$$

Before the BERT method conducts training, the dataset needs to be adjusted based on the input representation that BERT will receive using Formula 1. That is why we need a tokenizer where the goal is to tokenize sentences so that the input can match. The tokenization process is illustrated based on Figure 6.



Source : (McCormick & Ryan, 2019)
Figure 6. Tokenization Process

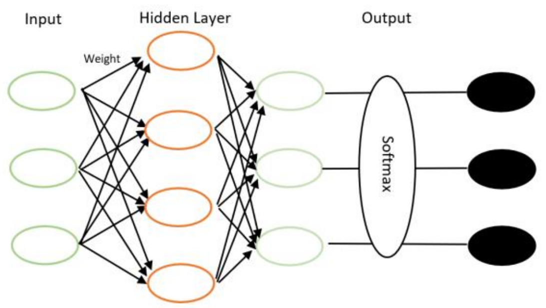
BERT will later receive a series of words or sentences used as input and follow up on the encoder stack. Self-attention will be sent to each encoder to provide results through a feed-forward network, which is then continued by the next encoder. The selected model in this study is BERTBASE so this process will be repeated 12 times. After passing through all the encoders, tokens at each position will generate a vector value with a hidden size of 768 in BERTBASE, as shown in Figure 7. For the sentiment analysis process, the output considered is the first position, the [CLS] token. This vector is used as the input of the classifier.



Source : (McCormick & Ryan, 2019)
Figure 7. Input and Output on BERT

BERT can achieve good results using only one neural network as a classifier(Devlin et al., 2019). The neural network layer is used for classification with the softmax function shown in Figure 10. Therefore, the BERT output used for classification is derived from the [CLS] token vector. It is because the [CLS] token is supposed to collect the average of the word tokens and take the sentence vector. The last level of the classification level generates a logit. Logit is generated as a result of rough probability prediction of rank sentences. Softmax takes the exponent of each logit value and converts this logit into a probability so that the sum of the probabilities is precisely 1. Therefore, the probability value can be either 0 or a positive number. The illustration used to perform sentiment analysis can be seen in Figure 8.

P-ISSN: 2656-1743 |E-ISSN: 2656-1735
DOI: https://doi.org/10.34288/jri. v5i1.169
Accredited rank 4 (SINTA 4), excerpts from the decision of the DITJEN DIKTIRISTEK No. 230/E/KPT/2023



Source : (McCormick & Ryan, 2019)
Figure 8. Layer illustration for sentiment analysis

After that, it enters the training process, where in this process, the dataset will be divided into three parts, namely the dataset for training, testing, and validation. This process aims to reduce the lengthy process due to large datasets. At this stage, a data loader is also needed to help reduce the memory used so that it can speed up the training process. This data loader is a function of PyTorch which will act as an iterator. The data loader will also create three parts: training, validation, and testing. At this stage, the AdamW optimizer will also be used to correct the weight of the sentence.

In the training phase, BERT will perform fine-tuning using predefined hyperparameters. Refers to(Devlin et al., 2019)hyperparameters include Batch size: 16, 32, Learning Rate (Adam): 5e-5, 3e-5, 2e-5, Epoch: 2, 3, 4

RESULTS AND DISCUSSION

Collecting Dataset

The author of this study uses public datasets that all parties can access. The dataset can use EDA ON the Data, Sentimental Analysis on the Data, and Build a Model that can predict the Impact on the Different Nations, To which people are Supporting More, and What People are thinking about the war. The dataset can be accessed via https://www.kaggle.com/datasets/vanamayaswanth/russia-vs-ukraine-tweets.

Labelling Dataset

Sentiment analysis uses supervised learning methods, so it requires a dataset that already has a label. Labeling aims to determine reviews into neutral, negative, and positive categories. This labeling is needed because the supervised learning method requires examples to generate generalizations until the model obtained is a prediction that matches the existing label(Goldberg, 2017). The model can see and understand which comments have positive, neutral, and negative labels.

The next stage is labeling the dataset. Labels are divided into 3, namely positive, neutral and negative. Where to get positive, neutral, and negative sentiments are grouped first into 0.1 and 2. The cleaned dataset is then labeled. The results of labeling data using 0, 1, and 2 can be seen in Figure 9. While the results of labeling using negative, neutral, and positive can be seen in Figure 10.

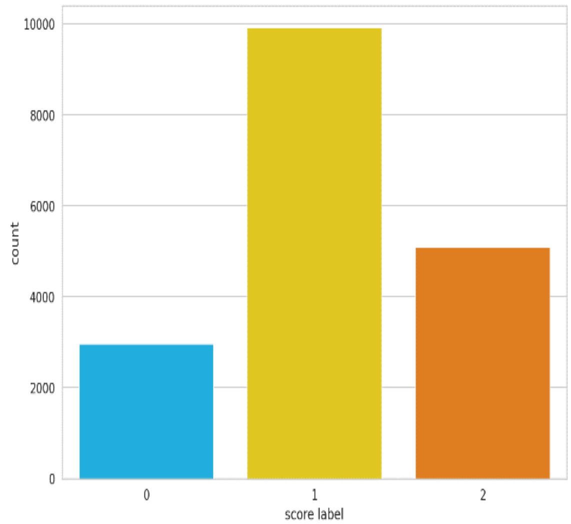


Figure 9. Data are labeled 0, 1 and 2

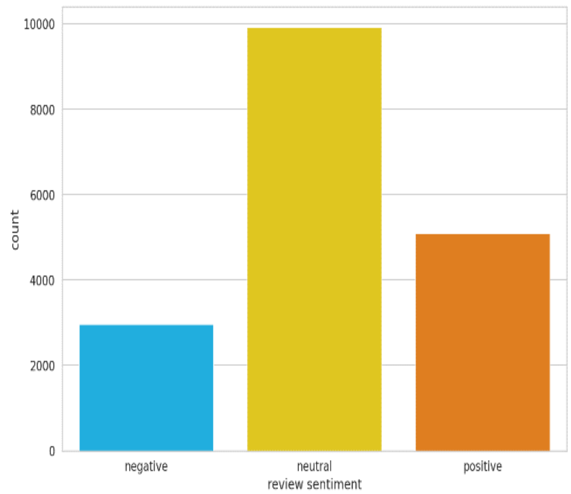
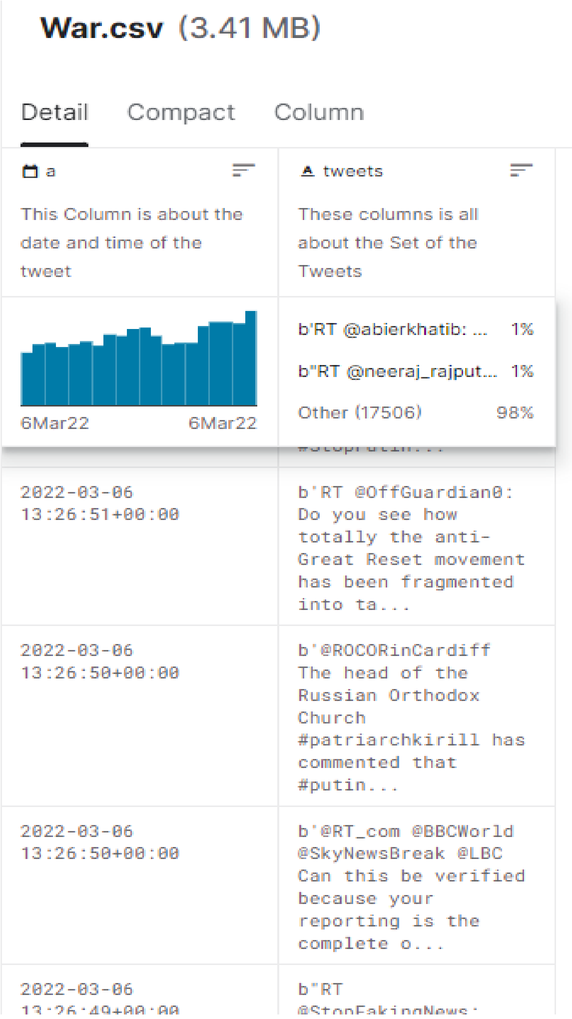


Figure 10. data that has been labeled

At the dataset analysis stage, the process of understanding the data used as research material is carried out. This study, in particular, uses the tweets column as the label. Where tweets are the contents of the opinion of Twitter citizens, below are some examples of tweets in the dataset in Figure 11.

P-ISSN: 2656-1743 | E-ISSN: 2656-1735
DOI: https://doi.org/10.34288/jri.v5i1.169
JURNAL RISET INFORMATIKA
Vol. 5, No. 1. December 2022
Accredited rank 4 (SINTA 4), excerpts from the decision of the DITJEN DIKTIRISTEK No. 230/E/KPT/2023



Source : (Yaswanth, 2022)
Figure 11. Dataset

Data Preprocessing

The next stage is cleaning the data, which removes symbols and numbers so that only text will be processed later. The following is the dataset-cleaning process in Figure 12:

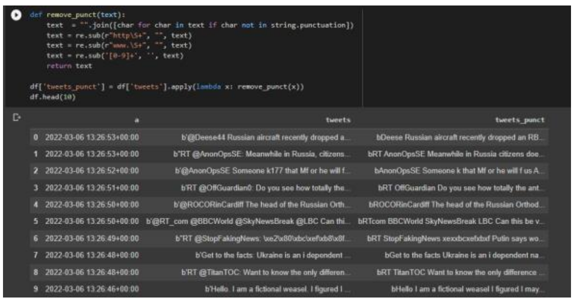


Figure 12. Cleaning Data

The preprocessing stage at this stage has several processes, namely, tokenization, unique tokens, and choosing a sequence length.

1. Tokenization

BERT's task is to classify two sentences. Tokenization is simple in this graph because BERT uses WordPieces as tokens rather than words --- so some words are broken down into simpler parts. (Alammar, 2021). Some basic operations can convert the text to tokens and tokens to unique integers (ids) shown in Figure 13:

```
 Sentence: stop this war
   Tokens: ['stop', 'this', 'war']
Token IDs: [1831, 1142, 1594]
```

Figure 13. Process tokenization

2. Special Tokens

[SEP] - at the end of the sentence, [CLS] - so a token must be added at the beginning so that BERT can run the classification. (Valkov, 2020) The results of removing unique signs are shown in Figure 14, namely cleaning special characters with sep_token of 102 characters, cls_token of 101 characters, pad_token of 0 characters, and unk_token of 100 characters:

```
Special Tokens
[SEP] - marker for ending of a sentence

[ ] tokenizer.sep_token, tokenizer.sep_token_id
    ('[SEP]', 102)

[ ] tokenizer.cls_token, tokenizer.cls_token_id
    ('[CLS]', 101)

[ ] tokenizer.pad_token, tokenizer.pad_token_id
    ('[PAD]', 0)

[ ] tokenizer.unk_token, tokenizer.unk_token_id
    ('[UNK]', 100)
```

Figure 14. result of special tokens

P-ISSN: 2656-1743 |E-ISSN: 2656-1735
DOI: https://doi.org/10.34288/jri. v5i1.169
Accredited rank 4 (SINTA 4), excerpts from the decision of the DITJEN DIKTIRISTEK No. 230/E/KPT/2023

3. Choosing Sequence length

Sequence length is one of the methods used to choose sequence length based on data that has been cleaned to produce a comparison of the taken count and density, as shown in Figure 15.

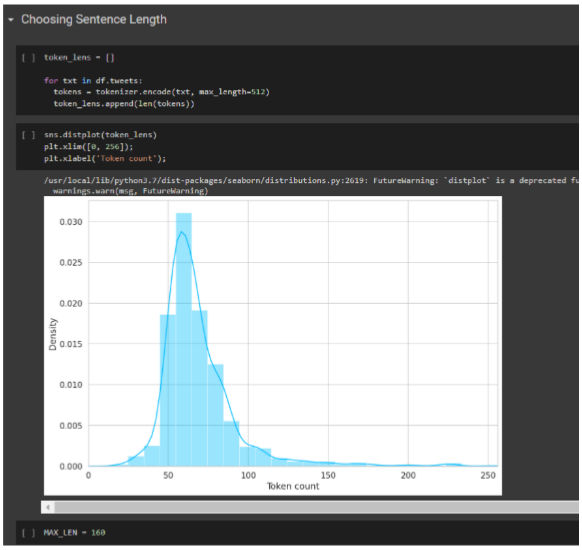


Figure 15. result of choosing sequence length

BERT Implementation

To be able to do a test, it takes a data loader for each dataset to be able to iterate. The goal is to maintain memory during training so that all datasets do not need to be entered into memory simultaneously. We also need a helpful block for creating a data loader to generate tokenized comments later. Sentiments and comments can be as long as 160 words. Researchers do fine-tune using hyperparameters using the recommendations from BERT as follows: Batch Size: 16, Epoch: 10, Learning Rate 2e-5 (0.0005)

The hyperparameters used are determined for several reasons, so batch 16 is chosen. The bigger the batch size, the longer it will take (Osinga, 2018). Then use epochs as much as 10(Song, Wang, Liang, Liu, & Jiang, 2020). In addition, the learning rate of 2e-5 was chosen because it can overcome the problem of forgetting disasters (Sun, Qiu, Xu, & Huang, 2019). The problem of understanding obtained from pre-training is removed while examining new data or information.

After determining the Hyperparameters, testing will be carried out using ten epochs. The dataset used in each training, validation, and testing is done without specifying a random seed. So that every time the test is carried out, the data is different. The results of the epochs test are shown in Figure 16.

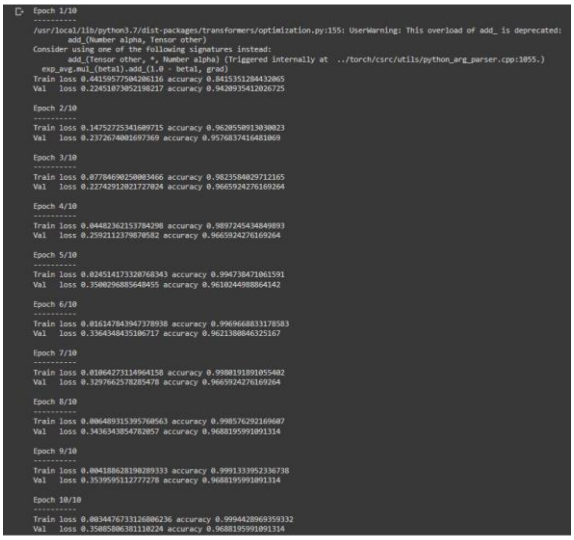


Figure 16. The training process and dataset evaluation.

After repeating the epoch on the model, the result that gets the best value will be used. Figure 18 compares the training and validation of the test shown in Figure 17.

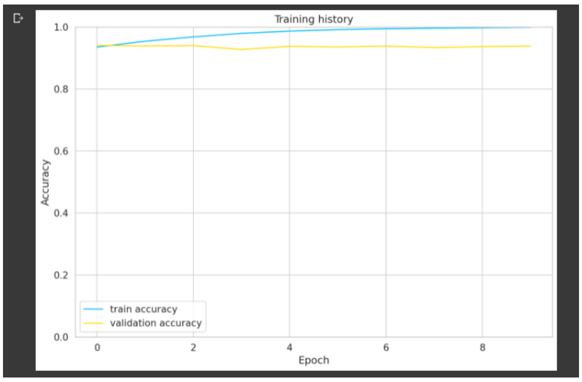


Figure 17. The curve of training and validation

From the experimental observations, it can be concluded that the training data has better results than the validation data. Training has a stable upward curve, while validation has a slightly up-and-down curve.

Training Dataset

The dataset is divided into a training, validation, and testing dataset before classification. Train the model using the training dataset. Meanwhile, to minimize overfitting, a validation dataset is used. As a final test to obtain network accuracy that has been trained with the training dataset, dataset testing is used. The results of the dataset training process in the form of a split dataset can be seen in Figure 18.

P-ISSN: 2656-1743 | E-ISSN: 2656-1735
DOI: https://doi.org/10.34288/jri.v5i1.169

Accredited rank 4 (SINTA 4), excerpts from the decision of the DITJEN DIKTIRISTEK No. 230/E/KPT/2023

```
[ ] df_train, df_test = train_test_split(df, test_size=0.1, random_state=RANDOM_SEED)
    df_val, df_test = train_test_split(df_test, test_size=0.5, random_state=RANDOM_SEED)

[ ] df_train.shape, df_val.shape, df_test.shape

    ((16155, 6), (898, 6), (898, 6))
```

Figure 18. Result of Splitting Dataset

The experimental results show the testing of the Twitter opinion dataset with the BERT model. The stages that will be carried out in the experiment are testing using a dataset divided into training, validation, and testing, which will later get the classification results from the desired sentiment.

Evaluation

BERT is used for sentiment analysis, using a dataset to test it based on the best accuracy value. After the accuracy value is obtained using dataset testing shown in Figure 19, the BERT model of the test data set is used to make predictions resulting in an accuracy value of 97%, as shown in Figure 20.

```
Evaluation

[ ] test_acc, _ = eval_model(
      model,
      test_data_loader,
      loss_fn,
      device,
      len(df_test)
    )

    test_acc.item()

    0.9743875278396437
```

Figure 19. Results using the testing dataset

	precision	recall	f1-score	support
negative	0.96	0.95	0.96	154
neutral	0.98	0.98	0.98	500
positive	0.96	0.97	0.97	244
accuracy			0.97	898
macro avg	0.97	0.97	0.97	898
weighted avg	0.97	0.97	0.97	898

Figure 20. Accuracy Results

Based on the confusion matrix diagram in Figure 21, it can be concluded that the neutral detecting system has a value that tends to be more significant.

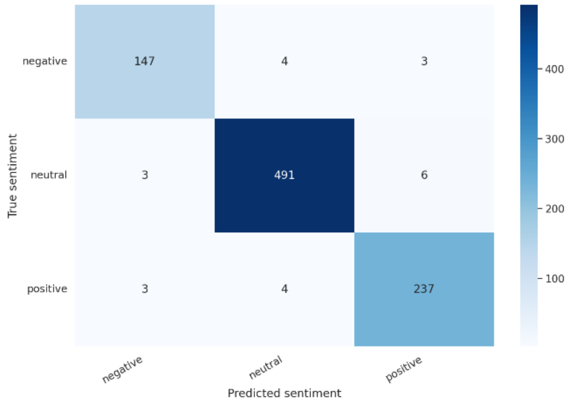


Figure 21. Confusion Matrix

CONCLUSIONS AND SUGGESTIONS

Conclusion

Sentiment analysis using Bidirectional Encoder Representations Transform (BERT) produces an accuracy of 97% with batch size 16 selection of hyperparameters, learning rate 2e-5, and epoch 10. These results can be said to be good results. The following conclusions are obtained based on the testing implementation of the BERT model in sentiment analysis. The results show more neutral sentiments than positive and negative sentiments.

Suggestion

The authors propose several suggestions for further research from the results, including a more balanced dataset between Positive, neutral, and negative sentiment. They are balancing the dataset before training so that better results are expected.

REFERENCES

Aditya, A., & Wibowo, A. (2022). Analisis Sentimen Menggunakan Metode Naïve Bayes Berdasarkan Opini Masyarakat Dari Twitter Terhadap Perang Rusia dan Ukraina. *Seminar Nasional Mahasiswa Fakultas Teknologi Informasi (SENAFTI) Universitas Budi Luhur*, (September), 138–145. Jakarta: Universitas Budi Luhur.

Alammar, J. (2021). The Illustrated BERT, ELMo, and co. (How NLP Cracked Transfer Learning).

Andraini, F., & Mahdiyah, E. (2022). Analisis Sentimen Twitter Terhadap Peperangan Rusia Dan Ukraina Menggunakan Algoritma

Support Vector Machine. *Jurnal Aplikasi Komputer*, *2*(1), 46–58.

Devlin, J., Chang, M. W., Lee, K., & Toutanova, K. (2019). BERT: Pre-training of deep bidirectional transformers for language understanding. *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, 4171–4186. Minneapolis, Minnesota: Association for Computational Linguistics. https://doi.org/10.18653/v1/N19-1423

El Rifai, H., Al Qadi, L., & Elnagar, A. (2022). Arabic text classification: the need for multi-labeling systems. *Neural Computing and Applications*, *34*(2), 1135–1159. https://doi.org/10.1007/s00521-021-06390-z

Fatima, A., Nazir, N., & Khan, M. G. (2017). Data Cleaning In Data Warehouse: A Survey of Data Preprocessing Techniques and Tools. *International Journal of Information Technology and Computer Science*, *9*(3), 50–61. https://doi.org/10.5815/ijitcs.2017.03.06

Kumar, A., Jaiswal, A., Garg, S., Verma, S., & Kumar, S. (2022). Sentiment Analysis Using Cuckoo Search for Optimized Feature Selection on Kaggle Tweets. *IGI Global Publisher of Timely Knowledge*, 19. https://doi.org/10.4018/978-1-6684-6303-1.ch062

Lomet, D. B. (2001). Bulletin of the Technical Committee on Data Engineering. *Bulletin of the Technical Committee on Data Engineering*, *24*(4), 1–56. Retrieved from papers2://publication/uuid/30073F7F-1B7C-4496-ADA4-94FF4E6EE8F7

McCormick, C., & Ryan, N. (2019). BERT Fine-Tuning Tutorial with PyTorch.

Osinga, D. (2018). *Deep Learning Cookbook* (1st ed.). Sebastopol, California: O'Reilly Media.

Song, Y., Wang, J., Liang, Z., Liu, Z., & Jiang, T. (2020). *Utilizing BERT intermediate layers for aspect-based sentiment analysis and natural language inference*. arXiv. https://doi.org/10.48550/arXiv.2002.04815

Sudiq, R. dwinanda, & Yustitianingtyas, L. (2022). Intervensi rusia terhadap ukraina pada tahun 2022 sebagai Pelanggaran Berat HAM. *Jurnal Pendidikan Kewarganegaraan Undiksha*, *10*(3), 101–117.

Sun, C., Qiu, X., Xu, Y., & Huang, X. (2019). How to Fine-Tune BERT for Text Classification? *Lecture Notes in Computer Science (Including Subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*. https://doi.org/10.1007/978-3-030-32381-3_16

Suresha, A. M. (2020). Sentiment Analysis on Amazon Product Reviews with Stacked Neural Networks. *Https://Www.Researchgate.Net/Publication/344518660_Sentiment_Analysis_on_Amazon_Product_Reviews_with_Stacked_Neural_Networks*, (October). https://doi.org/10.13140/RG.2.2.14746.67524

Valkov, V. (2020). Sentiment Analysis with BERT and Transformers by Hugging Face using PyTorch and Python.

Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A. N., … Polosukhin, I. (2017). Attention is all you need. *31st Conference on Neural Information Processing Systems (NIPS 2017)*, 1–11. Long Beach: NeurIPS.

Wiegreffe, S., & Marasović, A. (2021). *Teach Me to Explain: A Review of Datasets for Explainable Natural Language Processing*. (NeurIPS). Retrieved from http://arxiv.org/abs/2102.12060

Yaswanth, V. (2022). Russia Vs Ukraine Tweets: Tweets of the war that is happening between the Nations.

Yutika, C. H., Adiwijaya, A., & Faraby, S. Al. (2021). Analisis Sentimen Berbasis Aspek pada Review Female Daily Menggunakan TF-IDF dan Naïve Bayes. *Jurnal Media Informatika Budidarma*, *5*(2), 422. https://doi.org/10.30865/mib.v5i2.2845

P-ISSN: 2656-1743 | E-ISSN: 2656-1735
DOI: https://doi.org/10.34288/jri.v5i1.169

Accredited rank 4 (SINTA 4), excerpts from the decision of the DITJEN DIKTIRISTEK No. 230/E/KPT/2023

Blank Page