Journal of Informatics, Information System, Software Engineering and Applications (INISTA)

Sentiment Analysis Using Transformer Method

Andi Aljabar [1], Ircham Ali[*2], Binti Mamluaatul Karomah [3]

[1,2,3]*Universitas Nahdlatul Ulama Indonesia*

Taman Amir Hamzah No.5, RT.8/RW.4, Pegangsaan, Kec. Menteng, Kota Jakarta Pusat, Daerah Khusus Ibukota Jakarta 10320,

Indonesia

[1] andialjabar@unusia.ac.id
[2] irchamali@unusia.ac.id
[3] mamluatul93@unusia.ac.id

Abstract

This research delves into sentiment analysis, employing the transformative capabilities of transformer method. Specifically, leveraging BERT Models, the study aims to advance sentiment classification accuracy by intricately capturing contextual nuances and positive or negative comment on IMDB movies reviews. The transformer architecture's distinctive attention mechanism proves pivotal in comprehending intricate relationships between words, facilitating a profound understanding of sentiment in textual data. Through extensive experimentation, the study establishes the transformative prowess of these methods, showcasing their effectiveness in achieving state-of-the-art results in sentiment analysis tasks. This investigation not only contributes to the evolving landscape of sentiment analysis but also underscores the significance of transformer-based approaches in deciphering the subtleties of human expression within textual data specially for Bert models. This research will predict sentiment analysis on comment of IMDB movies and shows some results which are 3% of loss, 60% off loss validation, 98% of accuracy and 90% of validation accuracy

Keywords: Sentiment analysis, BERT models, Transformer Methods.

Corresponding Author:
*Andi Aljabar
Informatics Engineering Department, University of Nahdlatul Ulama Indonesia
Taman Amir Hamzah No. 5 RT.8/RW4. Pegangsaan, Kec. Menteng, Kota Jakarta Pusat, DKI Jakarta 10320
Email: andialjabar@unusia.ac.id

I. INTRODUCTION

PEOPLE'S innate attitudes toward a certain subject, person, or thing are known as their sentiments. It is beneficial for us to communicate, learn, and make decisions when we are aware of people's attitudes [1] – [5]. For instance, a business or retailer can adjust its operations accordingly depending on how consumers view its name or line of goods. Public opinion can be influenced by government entities through the assessment of online voting. over this reason, researchers in artificial intelligence have been working to give machines the cognitive capacity to identify, understand, and communicate emotions over the past 20 years. On the other hand, Companies utilize these materials to understand consumer perceptions of their products in real time and adjust their strategy accordingly. Prior to the release of new products, this data can also be utilized to track public opinion and resolve product limits [6].

The explosion of online social networking on daily basis in turn requires the necessity of high Cloud computing Graphics Processing Unit (GPU) and Tensor Processing Unit (TPU) processors with a particular purpose and architecture to store, process and analyze them. These have proliferated, and advances in Natural Language Processing (NLP) have allowed researchers to address a greater number of engineering language difficulties pertaining to people's feelings and beliefs. Exploiting people's thoughts, intentions for

conversations, feelings, and any latent patterns or hidden information in the form of any naturally occurring, human-to-human accessible canonical language is crucial. Additionally, disorganized, insufficient, and unstructured instant messages cover useful info [6]. To meet the data needs of clients, some researchers suggest organizing many these messages or microblogs into groups with prominent bunch marks. This will provide an overview of the content. For big and heavily tailed corpus texts, statistical topic models are unable to correctly extract these patterns.

In recent years, sentiment analysis has gained significant traction as a vital component in understanding public opinion and user feedback. One style of neural network architectures is the Transformer model. It is made to handle sequential data, like language [7]. The Transformer employs self-attention mechanisms to determine the relative importance of each word in a sequence, in contrast to previous models that depend on recurrent or convolutional layers. Because of this parallelization, it is very effective for training and enables the capture of intricate connections in data. Modern natural language processing models such as BERT, GPT, and others have their roots in the Transformer architecture. This study delves into the application of the Bidirectional Encoder Representations from Transformers (BERT) transformer method to advance the field of sentiment analysis [8] – [11]. BERT, renowned for its ability to grasp contextualized language nuances, emerges as a promising tool for enhancing sentiment classification accuracy [12]. As we navigate through the intricacies of human expression within textual data, the unique architecture of BERT allows for a more nuanced comprehension of language, capturing subtle dependencies and relationships. This introduction sets the stage for exploring how the utilization of BERT transformer can elevate sentiment analysis to new levels of precision and effectiveness. This research will address how the transformer works and can predict sentiment analysis by using some comment of IMDB movies.

II.   RESEARCH METHOD

The initial action is the study starts with a thorough literature review with the goal of comprehending the sentiment analysis field as it stands today and identifying the best transformer topologies for tasks involving natural language processing. In the early stages of sentiment analysis, text was the primary focus, and sentiment was determined solely by looking at the links between words and sentences. Nonetheless, depending just on textual data is inadequate for deriving human sentiments since non-verbal cues frequently cause a speaker's meaning to shift in real-time. The word "great," for instance, is analyzed by the model as generally positive; yet, if an exaggerated expression or sarcastic laughter is included, the expression may become negative. It has been suggested that multimodal sentiment analysis, which refers to the various modalities (text, audio, and video) via which people express and communicate their emotions, can help with this issue [2].

Extensive studies and research conducted over the years have demonstrated that multimodal systems outperform unimodal systems in detecting the emotions of speakers. According to a 2015 multimodal sentiment analysis survey, multimodal systems often outperformed their best unimodal counterparts in terms of accuracy. Since social media has grown so rapidly, a lot of films featuring people's personal thoughts have been posted on sites like Facebook and YouTube. These movies provide great resource support for multimodal sentiment analysis [2]. Typically, these films are reviews of movies, products, policies, etc. Videos offer rich visual and auditory information in addition to text, and feature fusion analysis of both modalities creates a multimodal sentiment analysis system. Moreover, doing data collection for supporting model. Following that, a diversified dataset comprising labeled sentiments is gathered, guaranteeing compliance with the study goals, and covering a range of sources and sentiments.

Preprocessing operations like tokenization and noise reduction are carried out after data gathering. The training step is preceded by the selection of a suitable BERT model, considering parameters such as model size and computer resources. The model is adjusted as needed once the dataset is divided into training, validation, and test sets. Evaluation metrics are defined for evaluating the performance of the model, such as accuracy and loss. This research method also is explained in Fig. 1.

Fig 1 explains about how to predict comment of IMDB by using transformer method start from literature review to result and analysis. In the literature review the researcher doing literary studies about some prediction of sentiment analysis by using some method of sentiment prediction. Sentiment analysis by using *GloVe-DCNN* on twitter by 85.63% of accuracy [13], Sentiment analysis of tweet before the election in Indonesia by using *IndoBERT* 83.50% of accuracy [14], *ABCDM* model for tweet sentiment prediction by 93% of accuracy [15]. Sentiment analysis on twitter by using Hybrid of SC, EC, (SentiWordNet), and IDSC classifier with 81% of accuracy [16]. Sentiment analysis using twitter data by using LSTM model [3].
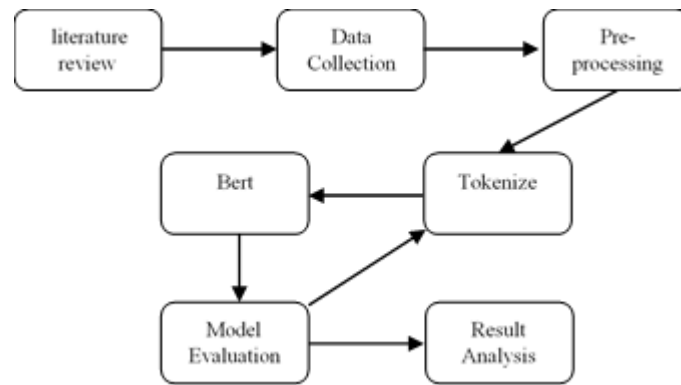
Fig. 1.   Research Methodology

TABLE I.   COMPARISON OF LITERATURE REVIEW

Model	Topic	Accuracy	Year
GloVe-DCNN	Sentiment analysis by using *GloVe-DCNN* on twitter (202)	85.63%	2018
IndoBert	Sentiment analysis of tweet before the election in Indonesia	83.50%	2024
ABCDM	*ABCDM* model for tweet sentiment prediction	93%	2021
SentiWordNet	Sentiment analysis on twitter by using Hybrid of SC, EC, (SentiWordNet)	81%	2022
LSTM	Sentiment analysis using twitter data by using LSTM model	97%	2021

For next step, research doing data collecting from IMDB and web scraping form IMDB website. After that, cleaning the special character of data which is downloaded from IMDB website. Next step is doing repeating for training and testing data by using transformer model. Firstly, the word will be given token (*tokenize method*). Every single word must be addressed as a token. After that using BERT method as a part of *transformer model.* The researcher repeats this step until found the best accuracy by modify epoch process and also evaluate model. The last process is result analysis.

TABLE II.   EPOCH

Epoch	Accuracy	Loss
2	87%	7%
5	90%	3%
7	93%	15%
8	95%	17%
9	98%	20%

A. *Tokenize*

A crucial step is tokenization, which entails dividing the input text into smaller pieces called tokens. breaks down words into *sub word* as number of token units using a *Word Piece* tokenizer, enabling a finer-grained representation of language. Tokenization is a basic preprocessing step in BERT, or Bidirectional Encoder Representations from Transformers, that includes breaking down input text into smaller linguistic units known as tokens. BERT includes a tokenization technique known as *WordPiece*, which allows the model to handle a wide range of words, including uncommon or out-of-vocabulary terms, by breaking them down into understandable subword units. For example, the word "running" may be tokenized into "run" and "ning," capturing the important subword components.

The BERT model uses a predefined vocabulary of sub words that includes entire words as well as sub word units seen during training. This vocabulary serves as the foundation for encoding any input text as a series of tokens. BERT adds unique tokens to address specific features of language representation in addition to entire words and sub word units. These tokens include [CLS] (classification), [SEP] (separator), and [MASK] (mask). To indicate the outcome of the classification problem, the [CLS] token is added at the beginning of the input sequence, whereas [SEP] tokens are used to separate different segments, particularly in tasks involving pairs of sentences.

Following tokenization, each token is assigned a unique numeric ID based on the model's vocabulary. This yields a series of token IDs that act as the BERT model's input representation. The [CLS] token, which is frequently used in classification tasks, denotes the start of the input, and [SEP] tokens demarcate different segments [17]. BERT uses segment IDs to discriminate between segments in tasks involving pairs of sentences. Furthermore, BERT employs attention masks to distinguish between tokens that correspond to actual words and those that are padding tokens. During training, this attention mask is required for the model to focus on meaningful sections of the input and ignore padding tokens.

Tokenization in BERT is essentially a complicated procedure that uses *WordPiece tokenization* to break down text into sub word units, maps these units to integer IDs using a set vocabulary, and integrates additional tokens and markers to aid in certain language representation tasks. The tokenized input is then fed into the BERT model for additional processing, allowing contextualized word embeddings to be learned and supporting a range of natural language processing applications. Every single word which has correlation will be given token sequentially. Fig 2 shows that how the tokenize works.
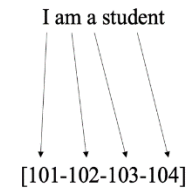
I am a student

[101-102-103-104]

Fig. 2.   Tokenize

B.   Bert Model

Bidirectional Encoder Representations from Transformers, or BERT [12], works by radically changing the way natural language context is interpreted. BERT's pre-training phase allows it to capture complex linkages and dependencies by learning to predict missing words inside sentences by considering each word's bidirectional context. The architecture of the model uses self-attention processes, which enable it to assess a word's value in both the left and right contexts of a sequence.

The architecture is based on transformers with many layers. Relevant information is highlighted by dynamic token weighting made possible by BERT's attention mechanism. When the model is fine-tuned for a particular job, it produces outputs that are specific to that task by improving its learnt representations. Because of its attention mechanisms and bidirectional approach, BERT can understand complex language patterns, which makes it an essential model.

How the *encoder* works? After doing tokenize BERT model will do prediction by counting the accumulation of tokenize. Those are token in which close then each number of tokens like 1 close with 2 or 3, the data training will predict positive, and also when they are opposite sentences will predict negative. Figure 3 explain how the encoder translate the token then doing prediction.
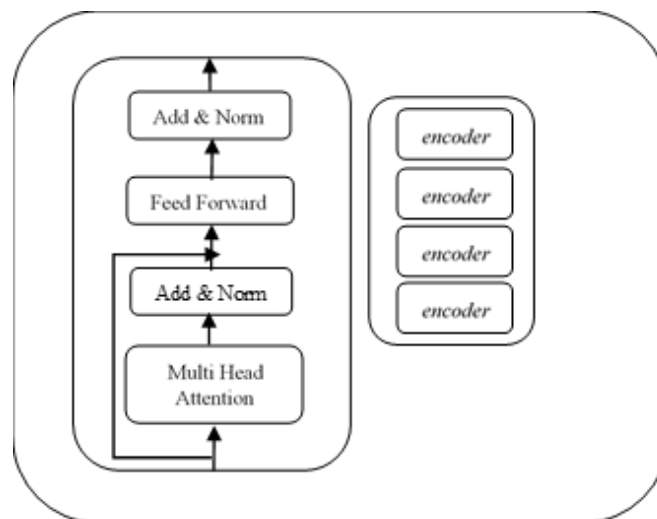
Fig. 3.   Bert Model

A key element of the Transformer model, a ground-breaking design in natural language processing, is *multihead attention*. The idea of *multihead attention* is incorporated into the Transformer's attention mechanism to improve the model's capacity to recognize intricate relationships in input sequences. This strategy involves deploying numerous attention heads in parallel.

The model may focus on distinct elements of the data because each attention head separately projects the input sequence into query, key, and value representations. The computation of attention scores for each

attention head involves assessing the dot product of the query and key vectors. Subsequently, attention weights are obtained using scaling and a *softmax* operation.

The final *multihead attention* output is obtained by concatenating and linearly transforming the results from each attention head after these weights are utilized to compute a weighted sum of the values. The model can simultaneously listen to many aspects in the input thanks to the use of multiple attention heads, which makes it easier for the model to learn complex patterns and dependencies within sequences. This mechanism's ability to capture long-range dependencies has made it a fundamental component of many state-of-the-art natural language processing models.

Another process is "*Add & Norm*" function is essential for helping deep neural networks be trained since it solves issues with gradient vanishing or exploding. Every sub-layer in the model has its output subjected to this process. Initially, a residual connection is used to add the sub-layer's output—for example, the outcome of a feedforward neural network layer or a *multihead attention* mechanism to its input. This additional step helps to mitigate possible problems during backpropagation and guarantees that information flows over the network without interruption.

The addition's outcome is then subjected to layer normalization. By normalizing the activations of the neurons within a layer, layer normalization improves stability and speeds up the training process. Scaling and shifting are then applied to the normalized output using learnable parameters. In Equation (1) the Transformer architecture, the *"Add & Norm"* operation mathematically expressed.

$$Norm(Add(x, SubLayer(x))) \qquad (1)$$

This formula is applicable to normalization and residual connection within each sub-layer of the model. This technique is critical in overcoming issues associated with deep neural network training, promoting robust and efficient learning. Also, carried out once for every sub-layer. This design decision is crucial to the Transformer model's performance in a range of natural language processing tasks since it enables the model to efficiently capture complex patterns in input. "*SubLayer(x)*" denotes the result of a sub-layer action, such as the multihead self-attention mechanism or the feedforward layer, applied to the input sequence marked by "*x.*" The "*Add(x, SubLayer(x))*" method adds the original input "x" to the sub-layer's output. During the training process, this residual link provides for the uninterrupted flow of information over the network.

After the addition operation, the result is normalized using the "Norm" function. This is typically layering normalization in the Transformer model, which standardizes the activations of the neurons in the layer. Normalization is critical for training process stability by preventing vanishing or ballooning gradients, which are prevalent in deep networks.

The expression "*Norm(Add(x, SubLayer(x))*" describes the process of applying a sub-layer operation to the input sequence, adding the original input via a residual link, and normalizing the combined output. This strategy, which is repeated across numerous levels of the Transformer, contributes to the model's depth and capacity to capture complex patterns and dependencies within input sequences, allowing it to be more effective in a variety of natural language processing tasks.

The *feed-forward* layer in the Transformer model is essential to the sequence processing carried out by the encoder and decoder layers. The feedforward layer oversees further modifying and honing the data that was taken from the input sequence after the self-attention process. Starting with a linear transformation, every location in the sequence goes through a separate operation. Then, a non-linear activation function typically a *Rectified Linear Unit (ReLU)* passes through the linearly transformed output, adding critical non-linearity to the model, and making it possible to capture complex patterns.

A further linear transformation is then used to project the data back into a space with more dimensions. Following these procedures, the feedforward layer's final output is obtained. The parameters of Equation (2) is the feedforward layer, represented mathematically.

$$FFN(x) = ReLU(Linear2(ReLU(Linear1(x)))) \qquad (2)$$

The feedforward layer within the Transformer model is represented by the expression "*FFN(x) = ReLU(Linear2(ReLU(Linear1(x)))*", which is a vital component that follows the *multihead self-attention process*. This configuration is used independently for each location in the input sequence, which helps the model catch detailed patterns and relationships in the data [6] .

"Linear1(x)" represents the outcome of the first linear transformation applied to the input sequence *"x"* in the feedforward layer. This linear transformation comprises multiplying the input by a weight matrix and adding a bias term, resulting in the input being projected into a higher-dimensional space. Following the application of the *ReLU* activation function, the model gains non-linearity, allowing it to learn complicated patterns and relationships.

By following *ReLU* activation, the output of the first linear transformation is transferred through the second linear transformation, designated as "*Linear2*." This operation, like the first linear transformation, involves a new set of learnable weights and biases. The *ReLU* activation function is used once more to introduce nonlinearity. The final outcome of this sequence of operations, described by the formula "*ReLU(Linear2(ReLU(Linear1(x))))*" is the feedforward layer's output for a given place in the input sequence.

The objective of the feedforward layer is to process and refine the information captured by the self-attention mechanism. The addition of non-linear activation functions, such as *ReLU*, enables the model to learn and reflect complicated data associations. During the training process, the parameters of the linear transformations (weights and biases) are learned, allowing the model to adapt to the specific properties of the input data [18]. Together with the self-attention mechanism and other Transformer components, the feedforward layer forms a powerful architecture for natural language processing tasks, demonstrating its effectiveness in capturing contextualized representations of words within sequences. There are acquired during training, enabling the model to adjust and recognize intricate correlations within the input sequences. Together with the self-attention mechanism, this part helps the Transformer perform better on a range of natural language processing tasks.

III. RESULTS AND DISCUSSION

The BERT model research on sentiment analysis has produced interesting findings that demonstrate the potency of the suggested strategy. By using 5003 rows of dataset, the sentiment analysis model performed robustly in accurately classifying sentiments in unseen data, as seen by its astounding 98% accuracy on the test set. Moreover, the 90% validation accuracy confirms the model's ability to generalize, demonstrating its effectiveness outside of the training set.

The model's capacity to reduce mistakes and converge during training is demonstrated by its low training loss of 3%. This implies that the sentiment data's underlying patterns were effectively learned using the transformer technique. A 60% loss on the validation side could mean that the validation set contains more difficult cases or that there is a certain amount of overfitting. Investigating if model modifications or new training techniques could improve generalization on the validation set is crucial.

More broadly, these findings add to the increasing amount of data demonstrating the effectiveness of transformer-based techniques in sentiment research. The model appears to have effectively captured complex patterns in sentiment expressions, as seen by its high accuracy and comparatively minimal training loss. On the other hand, the high validation loss demands a close look at possible enhancements to improve the model's performance on untested data. Subsequent research endeavors may encompass refining hyperparameters, investigating other transformer configurations, or augmenting the dataset to tackle plausible obstacles and enhance the model's performance.
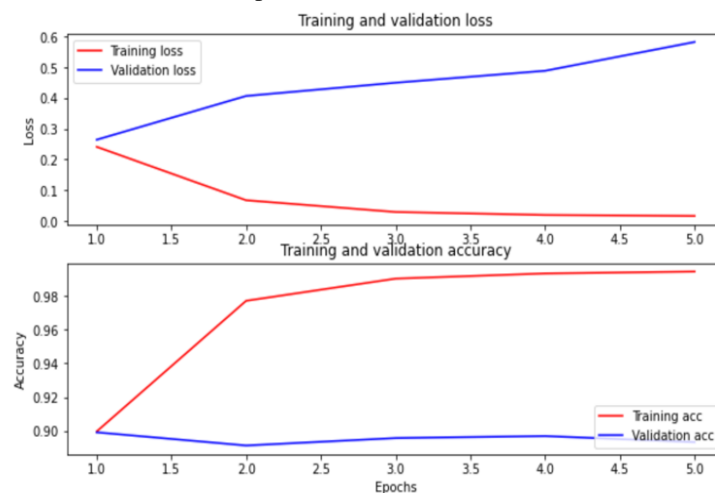

Fig. 4.   Accuracy, loss, and validation

IV.   CONCLUSION

In conclusion, findings add to the increasing amount of data demonstrating the effectiveness of transformer-based techniques in sentiment research. The model appears to have effectively captured complex patterns in sentiment expressions, as seen by its high accuracy and comparatively minimal training loss. On the other hand, the high validation loss demands a close look at possible enhancements to improve the model's performance on untested data. Subsequent research endeavors may encompass refining hyperparameters, investigating other transformer configurations, or augmenting the dataset to tackle plausible obstacles and enhance the model's performance.

ACKNOWLEDGMENT

REFERENCES

[1]   J. Cheng, I. Fostiropoulos, B. Boehm, and M. Soleymani, "Multimodal Phased Transformer for Sentiment Analysis," Association for Computational Linguistics. [Online]. Available: https://github.com/chengjunyan1/

[2]   L. Zhu, Z. Zhu, C. Zhang, Y. Xu, and X. Kong, "Multimodal sentiment analysis based on fusion methods: A survey," *Information Fusion*, vol. 95. Elsevier B.V., pp. 306–325, Jul. 01, 2023. doi: 10.1016/j.inffus.2023.02.028.

[3]   A. Aljabar and A. A. Abd Karim, "ANALISIS SENTIMEN MENGGUNAKAN ALGORITMA LSTM PADA MEDIA SOSIAL," *JUPIKOM*, vol. 1, no. 3, 2022, [Online]. Available: http://ejurnal.stie-trianandra.ac.id/index.php/jupkomHalamanUTAMAJurnal:http://ejurnal.stie-trianandra.ac.id/index.php

[4]   D. Tri Hermanto, A. Setyanto, and E. T. Luthfi, "Algoritma LSTM-CNN untuk Sentimen Klasifikasi dengan Word2vec pada Media Online LSTM-CNN Algorithm for Sentiment Clasification with Word2vec On Online Media".

[5]   W. E. Nurjanah, R. Setya Perdana, and M. A. Fauzi, "Analisis Sentimen Terhadap Tayangan Televisi Berdasarkan Opini Masyarakat pada Media Sosial Twitter menggunakan Metode K-Nearest Neighbor dan Pembobotan Jumlah Retweet," 2017. [Online]. Available: http://j-ptiik.ub.ac.id

[6]   S. Palani, P. Rajagopal, and S. Pancholi, "T-BERT -- Model for Sentiment Analysis of Micro-blogs Integrating Topic Model and BERT," Jun. 2021, [Online]. Available: http://arxiv.org/abs/2106.01097

[7]   A. E. Yüksel, Y. A. Türkmen, A. Özgür, and A. B. Altınel, "Turkish tweet classification with transformer encoder," in *International Conference Recent Advances in Natural Language Processing, RANLP*, Incoma Ltd, 2019, pp. 1380–1387. doi: 10.26615/978-954-452-056-4_158.

[8]   J.-B. Delbrouck, N. Tits, M. Brousmiche, and S. Dupont, "A Transformer-based joint-encoding for Emotion Recognition and Sentiment Analysis," Jun. 2020, doi: 10.18653/v1/2020.challengehml-1.1.

[9]   Z. Yuan, W. Li, H. Xu, and W. Yu, "Transformer-based Feature Reconstruction Network for Robust Multimodal Sentiment Analysis," in *MM 2021 - Proceedings of the 29th ACM International Conference on Multimedia*, Association for Computing Machinery, Inc, Oct. 2021, pp. 4400–4407. doi: 10.1145/3474085.3475585.

[10]   Z. Wang, Z. Wan, and X. Wan, "TransModality: An End2End Fusion Method with Transformer for Multimodal Sentiment Analysis," in *The Web Conference 2020 - Proceedings of the World Wide Web Conference, WWW 2020*, Association for Computing Machinery, Inc, Apr. 2020, pp. 2514–2520. doi: 10.1145/3366423.3380000.

[11]   S. Tabinda Kokab, S. Asghar, and S. Naz, "Transformer-based deep learning models for the sentiment analysis of social media data," *Array*, vol. 14, Jul. 2022, doi: 10.1016/j.array.2022.100157.

[12]   S. Alaparthi and M. Mishra, "Bidirectional Encoder Representations from Transformers (BERT): A sentiment analysis odyssey."

[13]   Z. Jianqiang, G. Xiaolin, and Z. Xuejun, "Deep Convolution Neural Networks for Twitter Sentiment Analysis," *IEEE Access*, vol. 6, pp. 23253–23260, Jan. 2018, doi: 10.1109/ACCESS.2017.2776930.

[14]   L. Geni, E. Yulianti, and D. I. Sensuse, "Sentiment Analysis of Tweets Before the 2024 Elections in Indonesia Using IndoBERT Language Models," *Jurnal Ilmiah Teknik Elektro Komputer dan Informatika (JITEKI)*, vol. 9, no. 3, pp. 746–757, 2023, doi: 10.26555/jiteki.v9i3.26490.

[15]   M. E. Basiri, S. Nemati, M. Abdar, E. Cambria, and U. R. Acharya, "ABCDM: An Attention-based Bidirectional CNN-RNN Deep Model for sentiment analysis," *Future Generation Computer Systems*, vol. 115, pp. 279–294, Feb. 2021, doi: 10.1016/j.future.2020.08.005.

[16]   A. Alsaeedi and M. Z. Khan, "A study on sentiment analysis techniques of Twitter data," *International Journal of Advanced Computer Science and Applications*, vol. 10, no. 2, pp. 361–374, 2019, doi: 10.14569/ijacsa.2019.0100248.

[17]   M. Boukabous and M. Azizi, "Crime prediction using a hybrid sentiment analysis approach based on the bidirectional encoder representations from transformers," *Indonesian Journal of Electrical Engineering and Computer Science*, vol. 25, no. 2, pp. 1131–1139, Feb. 2022, doi: 10.11591/ijeecs.v25.i2.pp1131-1139.

[18]   L. B. Hutama and D. Suhartono, "Indonesian Hoax News Classification with Multilingual Transformer Model and BERTopic," *Informatica (Slovenia)*, vol. 46, no. 8, pp. 81–90, 2022, doi: 10.31449/inf.v46i8.4336.