

Twitter-based classification for integrated source data of weather observations

Kartika Purwandari^{1,2}, Tjeng Wawan Cenggoro^{1,2}, Join Wan Chanlyn Sigalingging³,
Bens Pardamean^{1,4}

¹Bioinformatics and Data Science Research Center, Bina Nusantara University, Jakarta, Indonesia

²Department of Computer Science, School of Computer Science, Bina Nusantara University, Jakarta, Indonesia

³Database Center Division of BMKG, Meteorological, Climatological, and Geophysical Agency, Jakarta, Indonesia

⁴Department of Computer Science, BINUS Graduate Program-Master of Computer Science Program, Bina Nusantara University, Jakarta, Indonesia

Article Info

Article history:

Received Nov 15, 2021

Revised Jul 13, 2022

Accepted Aug 11, 2022

Keywords:

Classification

Deep learning

Geolocation

Machine learning

Natural language processing

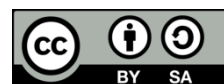
Transfer learning

Weather

ABSTRACT

Meteorology and weather forecasting are crucial for predicting future climate conditions. Forecasts can be helpful when they provide information that can assist people in making better decisions. People today use big data to analyze social media information accurately, including those who rely on the weather forecast. Recent years have seen the widespread use of machine learning and deep learning for managing messages on social media sites like Twitter. In this study, authors analyzed weather-related text in Indonesia based on the searches made on Twitter. A total of three machine learning algorithms were examined: support vector machine (SVM), multinomial logistic regression (MLR), and multinomial Naive Bayes (MNB), as well as the pretrained bidirectional encoder representations of transformers (BERT), which was fine-tuned over multiple layers to ensure effective classification. The accuracy of the BERT model, calculated using the F1-score of 99%, was higher than that of any other machine learning method. Those results have been incorporated into a web-based weather information system. The classification result was mapped using Esri Maps application programming interface (API) based on the geolocation of the data.

This is an open access article under the [CC BY-SA](https://creativecommons.org/licenses/by-sa/4.0/) license.



Corresponding Author:

Kartika Purwandari

Bioinformatics and Data Science Research Center, Bina Nusantara University

Jakarta, Indonesia

Email: kartika.purwandari@binus.edu

1. INTRODUCTION

Indonesia has a sea area of 6.22% of its relative area. The Indonesian territory is therefore characterized by a marine climate [1]. Global warming has led to the change of climate, especially during the dry and rainy seasons. The dry season lasts longer since it lasts longer, whereas the rainy season is shorter and occurs at a different time [2], [3]. The characteristics of multiple physical mechanisms and the dynamic nature of rainfall make it difficult to determine its consistency [4]. Intergovernmental Panel on Climate Change (IPCC) points out that climate change will require adaptation to environmental, social, and economic factors. Climate often changes in Indonesia because of its tropical location. Government mandates providing real-time weather data to support community activities [5], [6].

The advancements in technology have already led to progress in disseminating information; most of the information that the community receives comes from social media [7]. The government distributes publications in a variety of ways to meet the information needs of the public. Furthermore, the public aims to

stay informed about what happens around them, especially in relation to relevant events [8]. Twitter is used by people worldwide to access different types of information, including all kinds of information on Twitter. Monitoring topics and events is made easier with a structured combination of search parameters on a Twitter channel. We implemented the geolocation by using available application programming interface (APIs) and web services. Using existing APIs, location-specific terms were detected in a tweet. Social media platforms are continually generating and delivering information in real-time from various sources to users. Topics, hashtags, geographic location, language are extracted from tweets. In addition to scraping followers, likes, and retweets, a Python package called Twitter intelligence tools (Twint) allows users to identify their followers.

A variety of Twitter accounts, mostly related to information, have emerged all over the world in the last few years, most notably in Indonesia [9], [10]. The platform can be used to track public discussions about several issues that have been shared via Twitter. Data and information from Twitter have been used for classification tasks in a number of projects [11], [12]. Using the K-nearest neighbor (KNN) algorithm, a potential company's employees can be identified by their personalities. KNN identified the Myers-Briggs type indicator (MBTI) categories based on character classifications for potential employees from tweets [13].

Deep learning enhances the performance of various fields. Among the data types covered are images [14], [15], time series data [16], sounds [17], and text [18], [19]. Due to its time requirements and costs, bidirectional encoder representations from transformers (BERT) presents a challenge when used to classify large datasets, but it is generally still used because it is relatively inexpensive to train. Thus, the author used the BERT algorithm, which can only learn from datasets containing at least 256 characters [20]. In this study, we investigated whether sentiment analysis in texts can be classified using BERT-base. Using the Pontiki dataset, known as the laptop dataset [21], BERT, developed by Alexander Rietzler and fine-tuned with several layers, has been successful in detecting sentiment. A machine learning classification method was developed by the author using the support vector machine (SVM) technique before the BERT method, which provided 93% accuracy. As a result, other machine learning algorithms, such as multinomial Naive Bayes (MNB) and multinomial logistic regression (MLR), did not achieve highly accurate predictions when applied to Twitter data about weather conditions [22].

Data collected by the Meteorology, Climatology, and Geophysics Agency (BMKG) can be obtained from a number of sources. In Figure 1, it can be seen that the BMKG collects data and integrates them with each other to provide information on meteorology, climatology, and geophysics. The integration capabilities of BMKG can be enhanced by implementing a big data system that integrates multiple data sources. The first step to gathering weather data is to use automatic surface air instruments like automatic weather stations (AWS) and automatic rain gauges (ARG). An ARG is an instrument that measures rainfall. The two methods of recording data using this tool are manually (non-recording) and automatically (self-recording). In addition to rainfall information, weather forecasts also require data such as temperature, wind speed, and air humidity. Data can be obtained from AWS.

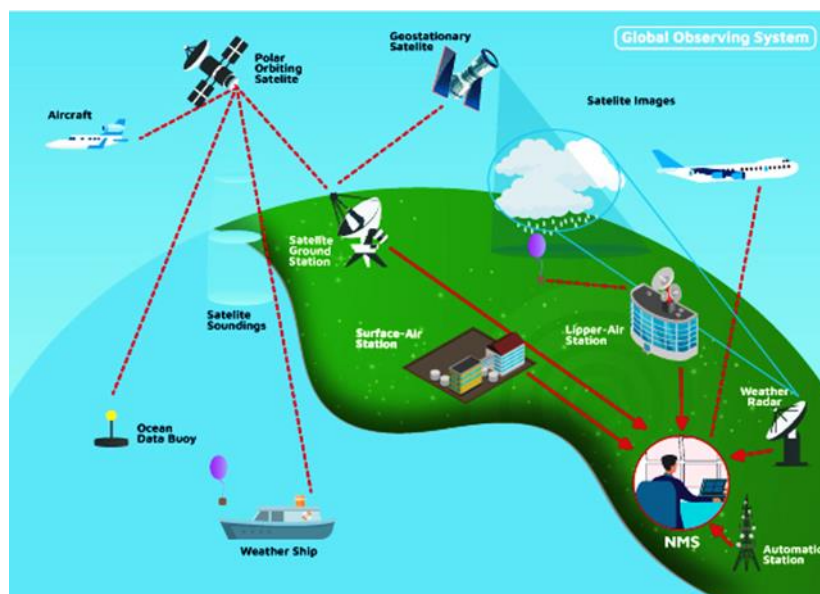


Figure 1. Global observing system on meteorology, climatology, and geophysics

The national weather service (NWS) forecasts and issues warnings for weather and hydrologic conditions in the United States, its territories, and adjacent waters and oceans, for the purpose of protecting lives and property and enhancing the nation's economy. As a complementary service, the NWS delivers Twitter feeds as a means of enhancing the reach of its information. In addition to disseminating environmental information, NWS will engage in outreach and education to increase awareness of weather conditions.

In this paper, we propose a machine learning method to integrate real-time weather data about Indonesia to support data diversity. Data from Twitter is used as the basis for this machine learning process. According to the Twitter location data, the crawled data is geolocated and entered into the database. The paper is organized firstly how weather information was collected using Twitter, followed by a description of the methodology used to analyze the data and summarizes the results and discusses them in detail.

2. METHOD

A Twitter framework for providing weather information can be seen in Figure 2. Figure 2 illustrates how data is stored in a database and reported in real-time to netizens through the android application. In the text preprocessing phase, uniform resource locators (URLs) are removed and unused words are eliminated, including stop words in the Indonesian language. Special characters are also removed. Authors determine the class based on the label generated during the training process and weather consultant in the classification phase. Geolocation filling is done based on the name of the district or city aforementioned inside a tweet.

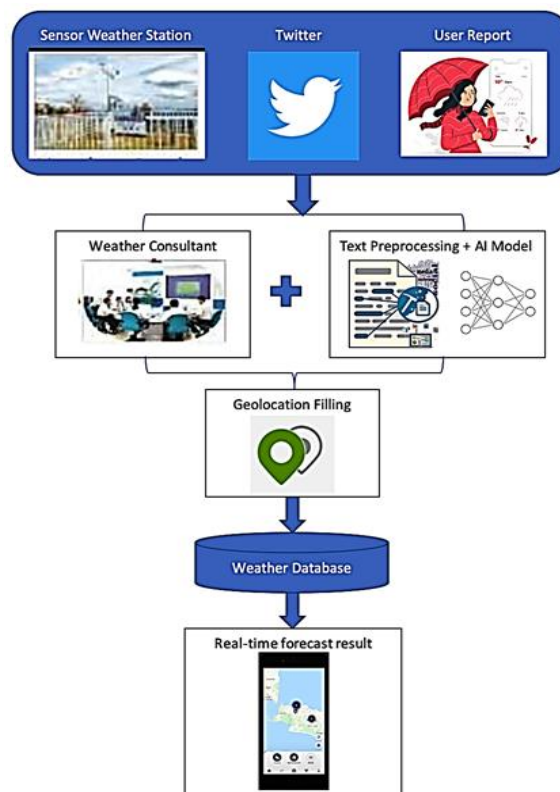


Figure 2. Integrated source data for the weather information system

2.1. Dataset

GetOldTweets3 was used to crawl tweets for the dataset. It is a Python 3 library for retrieving old tweets. According to Table 1, tweets were crawled from January to May 2019 that contained keywords derived from Indonesian (which were already translated into English). An Indonesian tweet is marked by a code of the language 'id'. A total of 506 tweets have been labeled. This experiment divided 20% of the total dataset into testing and control groups according to the Pareto ratio. We obtained 404 images for training and 102 images for testing from this process.

Table 1. Keywords for each class

Class	Keywords
Cloudy	Thick clouds, cloudy clouds, dark clouds
Sunny	The body gets wet of sweat, bright light, ill, clear, hot
Rainy	Rainy, rain, rainfall
Heavy rain	Lightning, thunderstorm, thunder, soaking wet
Light rain	Light rain, spatter, drizzle, spatter

In all, five classes of data were analyzed, namely "light rain", "heavy rain", "rainy", "sunny" and "cloudy". Figure 3 summarizes how these labels were distributed. Due to the similarity of the keywords for "rainy", "heavy rain" and "light rain", the amount of data in these classes is lower than the amount of data in the "sunny" class.

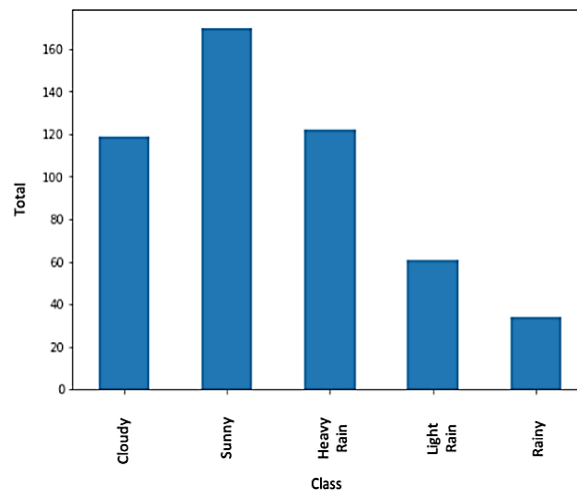


Figure 3. Data distribution

2.2. Pre-processing

Lowercase is applied to tweets. As shown in Figure 4, the following are removed from content: excessive newline characters and whitespace, URLs, Twitter and Instagram formatting, and non-American Standard Code for Information Interchange (ASCII) letters. Tweets containing emojis are translated using 116 emoji symbols formed in a .txt file, while tweets that contain slang words are transformed by 2,879 slang words written in a text file. Tokens are added to the beginning of each BERT text for classification [CLS].

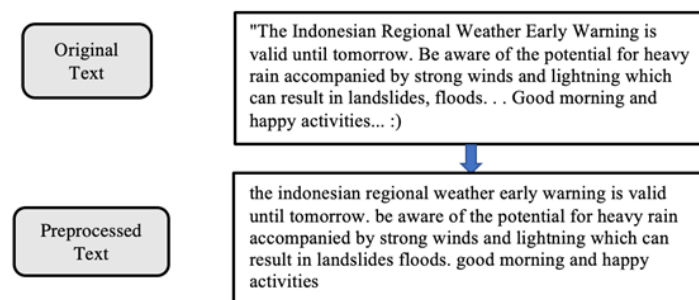


Figure 4. Comparison of tweet before and after text preprocessing

2.3. Feature extraction for machine learning algorithms

TfidfVectorizer is a machine learning algorithm that is based on term frequency–inverse document frequency (TF-IDF) and specifically processes words in a document [23]. By using this method, the inverse document frequency of a word (term) can be tracked [24]. The TF is calculated by counting how many words are in the word. The IDF method answers this question by determining which side of a document has more

weight. In other words, TF and IDF play a preliminary round match and determine the winner. To calculate the weight (W) of each document against keywords, the IDF TF algorithm uses the (1),

$$W_{dt} = Tf_{dt} * Idf_{dt} \quad (1)$$

W_{dt} = the weight of document d against word t.

Tf_{dt} = the frequency of occurrence of term i in document j divided by the total terms in document j , explained in the (2),

$$Tf_{dt} = \frac{f_{d(i)}}{f_{d(j)}} \quad (2)$$

Idf_{dt} is the function to reduce the weight of a term if its appearance is scattered throughout the document as spelled out in (3).

$$Idf_{dt} = \log \left(\frac{N}{df_t + 1} \right) \quad (3)$$

$df_t = |\{d \in D : t \in d\}|$ is the number of documents containing term t and N is the total number of documents in the corpus, $N = |D|$. Adding 1 to avoid dividing by 0 if df_t is not present in the corpus [25].

2.4. Classification method based on machine learning approaches

SVMs are supervised learning classification methods. In the SVM method, the original training data is mapped into a higher dimension using nonlinear mappings. The goal of this technique is to find the best separator function (hyperplane) to separate pairs of objects among all possible functions. In general, the best hyperplane can be defined as a line connecting two classes of objects. Using an SVM, the best equivalent hyperplane is constructed by maximizing the margins or distances between two different sets of classes [26], [27].

Naive Bayes with multinomial structures is a development of the Naive Bayes method which uses Naive Bayes to determine a probability value as to how often a word appears in a sentence. It affects the probability value according to the frequency with which that word appears in a sentence. However, there is a problem if a word is not included in any class in the Naive Bayes multinomial method. Probabilities 0 or zero are affected by this [28].

The scikit-learn Python package provides the Laplace smoothing method that avoids zero probabilities. As long as the α value is greater than 0, this method works by adding the α value. This value is set to 1.

$$P(c_j) = \frac{\text{count}(w_i, c_j) + \alpha}{\text{count}(c_j) + |V|} \quad (4)$$

where $P(c_j)$ is the probability value of word i against class j , $\text{count}(w_i, c_j)$ is the value of occurrence of word i in class j , and α is the value of Laplace smoothing (default as $\alpha = 1$). Then, $\text{count}(c_j)$ is the number of members of class j and $|V|$ is the number of members of the entire class without doubling.

In machine learning, MLR, also called softmax regression, is a method of separating classes of feature vectors from several classes. This method generalizes the logistic regression classification scheme for solving multiclass problems [29]. The main difference between the methods is the activation function. In MLR, sigmoid activation functions are used, while softmax activation functions are used in logistic regression. The scikit-learn logistic regression package in Python can be set up to use MLR by selecting multi-class as "multinomial".

2.5. BERT as a deep learning method

BERT is a two-way method based on a transformer architecture, replacing long short-term memory (LSTM) and gated recurrent units (GRU) in a sequential way with an attention approach that is faster. Additionally, the method was pre-trained to perform two unsupervised tasks, including modeling the masked language and predicting the next sentence. The pre-trained BERT method is utilized to perform downstream tasks like sentiment classification, intent detection, and question answering [30].

Documents may be classified according to multiple labels or classes simultaneously and independently, as indicated by multi-label classification. The multi-label classification has numerous real-world applications, such as categorizing businesses or assigning multiple genres to a film [31]. It can be used in customer service to determine multiple intentions for a customer email [32].

BERT-Base has a vocabulary of 30,522 words. Tokenization consists of splitting input text into tokens within a vocabulary. WordPiece tokenization is used by BERT for words that are not in its vocabulary. The outside words are gradually subdivided into sub-words and then represented by groups of sub-words [33].

2.6. Fine-tuning BERT

BERT is a network architecture that has been trained using large datasets from a wide variety of articles in multiple languages. Consequently, rather than train the BERT layer, which already has very good weights, researchers need to fine-tune the BERT layer for text classification [34]. Figure 5 depicts the input layer of the BERT method used to feed pre-processed tweets, followed by one dense layer employing tanh activation function, two dropout layers 0.5, and one output layer employing softmax activation function and cross-entropy loss. BERT pre-trains are fine-tuned by adding two dropouts (0.5), one dense layer, and one output layer. It is intended to stop overfitting by adding two dropout layers. Overfitting is when a model is too successful as a result of the training process, but it has the disadvantage that it is too dependent on training data, so the results are incorrect when new data is provided for classification [35]. In this model, 10 epochs were used with batch sizes of 5 and a sequence length based on the length of the dictionary from a tweet, which is the maximum for the previously trained model. AdamW optimizer was used with a learning rate of 3 e-5.

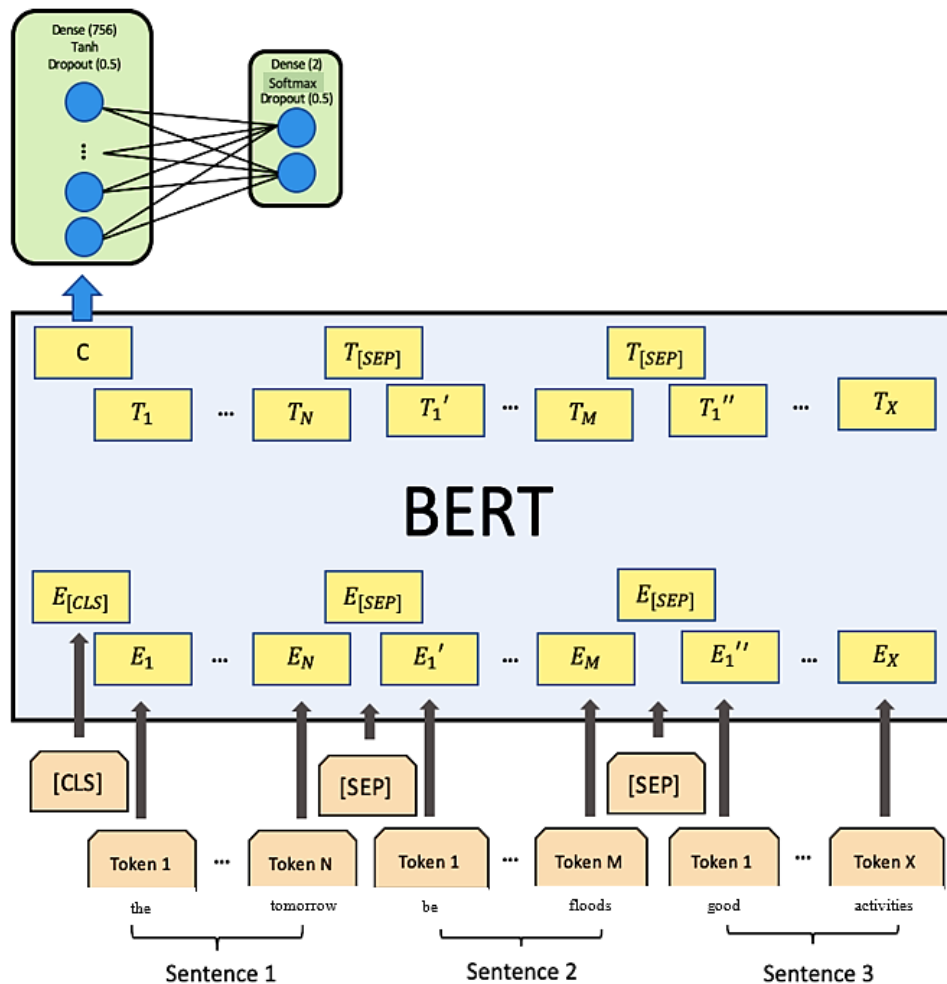


Figure 5. BERT fine-tuning model

2.7. Evaluation metrics

Confusion matrices are commonly used for calculating accuracy. The confusion matrix provides information on the comparison between the results generated by the model (system) and those actually

generated [36]. As shown in Table 2, there are 4 terms representing the classification results. TP, TN, FP, and FN represent true positive, true negative, false positive, and false negative, respectively. A test is conducted on F1-score, recall, and precision values in order to determine the accuracy of the results. Models are evaluated based on their F1-score, as they perform well on imbalanced datasets [37]. In (5) and (6), F1-score can be calculated for each class that offers the same weighting for recall and precision.

$$F1 - score = \frac{(2*(Precision*Recall))}{(Precision+Recall)} * 100\% \quad (5)$$

There is a weighted F1-score in which recall and precision can be assigned different weightings.

$$F_{\beta} = \frac{(1+\beta^2)*(Precision*Recall)}{((\beta^2*Precision)+Recall)} * 100\% \quad (6)$$

β reflects how much recall is more important than precision. The value of β is 2 if the recall is twice as significant as precision [38].

Table 2. Confusion matrix

		Actual class	
		Relevant	Non-Relevant
Predicted class	Retrieved	Correct result True positive (TP)	Unexpected result False positive (FP)
	Not retrieved	Missing result False negative (FN)	Correct absence of result True negative (TN)

The precision (7) indicates the system's ability to find the most relevant documents and is defined as the percentage of documents located and relevant to the query. A recall (8) measures the ability of the system to locate all relevant items from a document collection and is defined as the percentage of documents relevant to a query. The accuracy of the (9) is a comparison between correctly identified cases and the number of identified cases, compared to the error rate (10) on incorrectly identified cases.

TP = The number of correct predictions from relevant data.

FP = The number of incorrect predictions from irrelevant data.

FN = The number of incorrect predictions from irrelevant data.

TN = The number of correct predictions from relevant data.

$$Precision = \frac{TP}{(TP+FP)} * 100\% \quad (7)$$

$$Recall = \frac{TP}{(TP+FN)} * 100\% \quad (8)$$

$$Accuracy = \frac{(TP+TN)}{(TP+FP+TN+FN)} * 100\% \quad (9)$$

$$Error Rate = \frac{(FN+TN)}{(TP+FP+TN+FN)} * 100\% \quad (10)$$

2.8. Database management and geolocation filling

Purwandari *et al.*, developed a database management system that manages weather data for Indonesia. Three users are involved in this system: netizens, forecasters, and data engineers. A source of data from netizens is collected using tweets from Twitter, forecasters rely on data from BMKG sensors throughout Indonesia, and all data is analyzed by data engineers before being reported to the public. A data dictionary, entity-relationship diagrams, and use cases have been used to visualize all completed data [39].

In spite of this, less than 1% of the crawled tweet posts include geolocation information. Therefore, it is very important to ensure accurate predictions of the tweet posts for non-geo-tagged tweets when analyzing data in different domains. Moreover, we can alter it by modifying the city district database by adding district/city aliases to reflect the crawled tweets. Using this method, tweets from remote areas of Indonesia can still be displayed with the longitude and latitude even if the global positioning system (GPS) is not turned on. The content of tweets and metadata information can be used to identify a user's location even if Twitter has access to this information. In such a case, third parties will have to use other sources to identify the geolocation of a user or tweet.

3. RESULTS AND DISCUSSION

3.1. Evaluation of machine learning algorithms

This study compared three machine learning methods. The results are shown in Table 3. According to this table, SVM can successfully classify Twitter texts about the weather with a recall value of 87.3% and a precision value of 90.6%. The MLR method yields 83.3% recall and 90.3% precision. With the MNB method, the recall value is 73.6% and the precision is 86.3%. SVM provided the most accurate results, followed by MLR, then MNB, and also displayed the lowest error rate in comparison. Especially on Twitter about weather documents, SVM has proven to be effective in text classification. This is evident from the results of the test on the weather text classification, which showed that the recall value was lower than the precision value. Therefore, the precision level in this text classification was found to be effective. SVM became popular due to its accuracy and recall; this was confirmed during the test of the method.

Table 3. Classifier evaluation using machine learning approaches (%)

Model	Precision	Recall	F1-score	Accuracy	Error rate
SVM	90.6	87.3	88.1	87.3	12.7
MLR	90.3	83.3	85.6	83.3	16.7
MNB	86.3	73.6	77.5	73.5	26.5

An understanding of machine learning models requires a confusion matrix. The columns of the confusion matrix represent instances of the prediction class, whereas the rows represent instances of the actual class. The confusion matrix results illustrated in Figures 6(a) to 6(c) support the aforementioned results. Figure 6(a) illustrates the confusion matrix results from using SVM. Based on Figure 6(b), the method of MLR is also quite efficient for classifying weather-related tweets on Twitter. Figure 6(c) shows that the results of the MNB method are poor for the "light rain" class, since no TPs are generated in this class as indicated by the confusion matrix.

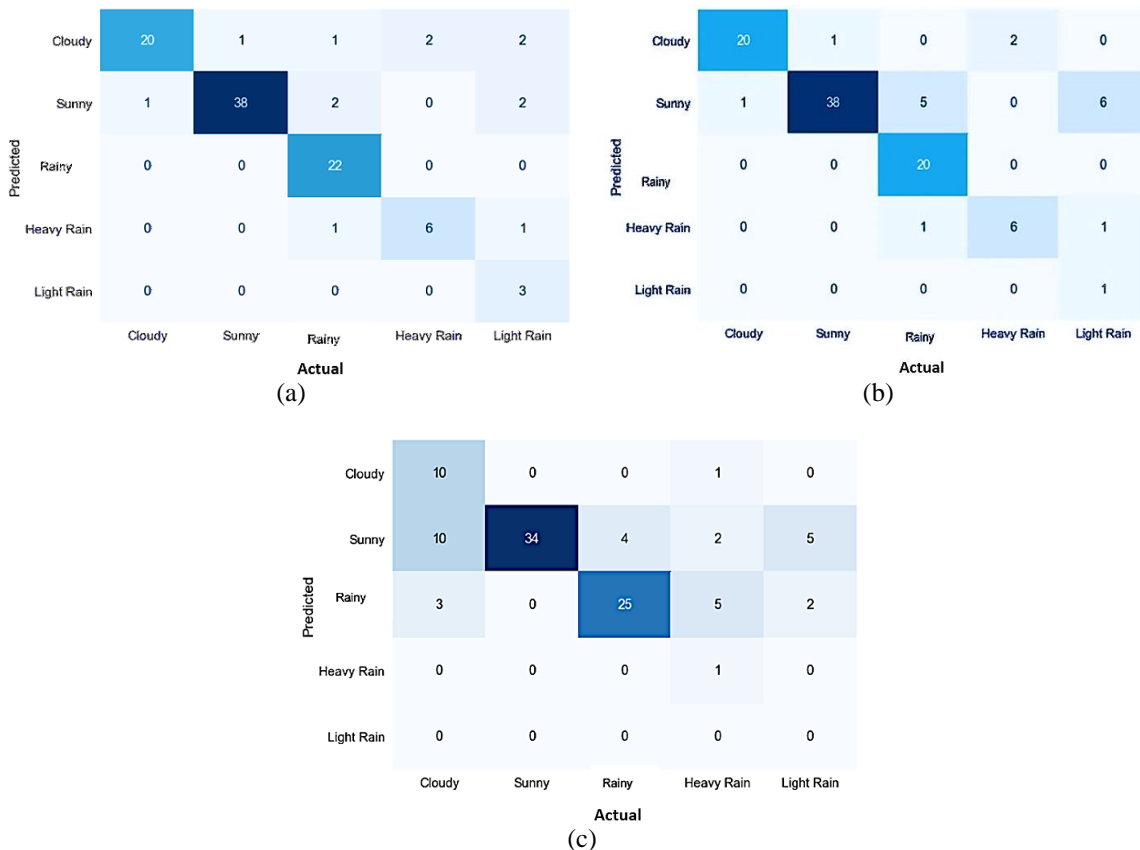


Figure 6. The confusion matrix results illustrated in (a) SVM, (b) MLR, and (c) MNB

3.2. Evaluation of BERT method

BERT method confusion matrix is depicted in Figure 7. It was concluded that cloudy, sunny, and light rain classes are able to perform classification well, meaning that the three classes have the exact same number of TP results as the actual number of sentences. There was one data point predicted as 'heavy rain' (FN). As can be seen, there is 1 prediction in the "rainy" class FP for the "heavy rain" class. Correct predictions are located in the diagonal figures, so visually it is obvious that unexpected predictions lie outside the diagonal confusion matrix. As shown in Table 4, the results of precision, recall, F1-score, accuracy, and error rate using the BERT method are 99.1%, 99%, 99%, 99%, and 1% respectively.

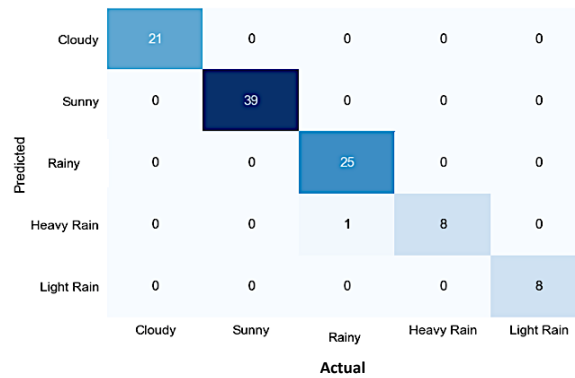


Figure 7. Confusion matrix of BERT model

Table 4. Classifier evaluation using BERT method (%)

Model	Precision	Recall	F1-score	Accuracy	Error rate
BERT	99.1	99	99	99	1

Table 5 provides precision, recall, and F1-score from each class demonstrating the results. BERT model generated a maximum F1-Score for "cloudy", "sunny", and "light rain" classes, and it was worked on as well for "rainy" and "heavy rain" classes. A model's output results can be ensured by using experimental results that have been used to analyze training data loss and validation. In Figure 8, it can be seen that loss from training data is often constant, increasing from epoch 4 to epoch 5, whereas loss from data validation is more unstable, increasing from epoch 4 to epoch 5. Validation losses tend to produce unreliable results due to the random input data each epoch receives. It can be said that the BERT model is very robust and stable. Unfortunately, due to the imbalanced distribution of data in Figure 3, overfitting occurred as a result of training data.

Table 5. Evaluation metrics for individual classes using BERT model (%)

Class	Precision	Recall	F1-score
Cloudy	100	100	100
Sunny	100	100	100
Rainy	100	96.2	98.1
Heavy rain	89	100	94.2
Light rain	100	100	100

Additionally, Figure 9 displays the F1-Scores for each epoch along with Figure 8. In spite of the decrease in yields from epoch 8 to epoch 9, it can be shown that the yield increases with each succeeding epoch. Every epoch contains five batches, and each batch must complete its task before the weight is changed. Weights are updated based on the estimated sum of losses. Using the convolutional output with BERT layers, the loss function is computed. Weights with the best quality will be saved for testing after the epoch has ended.

3.3. Web-based weather report

Following the BERT model, the next step would be to fill out the empty geolocations in the tweet. Geolocations are determined based on latitude and longitude coordinates of the cities and districts from

Central Agency on Statistics (BPS) which has been integrated into the database. This is a necessary step before integrating weather information on a website. Once the geolocation point has been filled, the latitude and longitude points are plotted into Esri Maps. Example of plotting weather reports submitted by netizens into Esri Maps, shown in Figure 10. As can be seen in the report, the first geolocation shows the word "South Jakarta". Consequently, the tweet is positioned at coordinates 6.2615° S, 106.8106° E, which means South Jakarta coordinates.



Figure 8. The plot of model loss on training and validation datasets

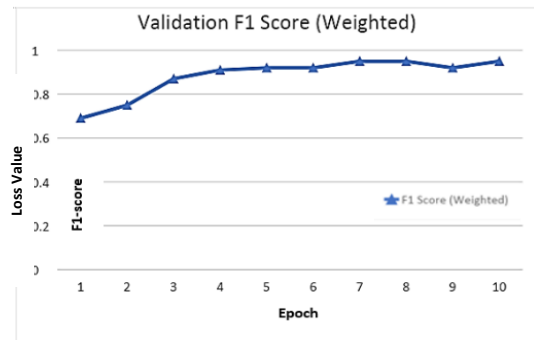


Figure 9. The plot of F1-score result on validation datasets

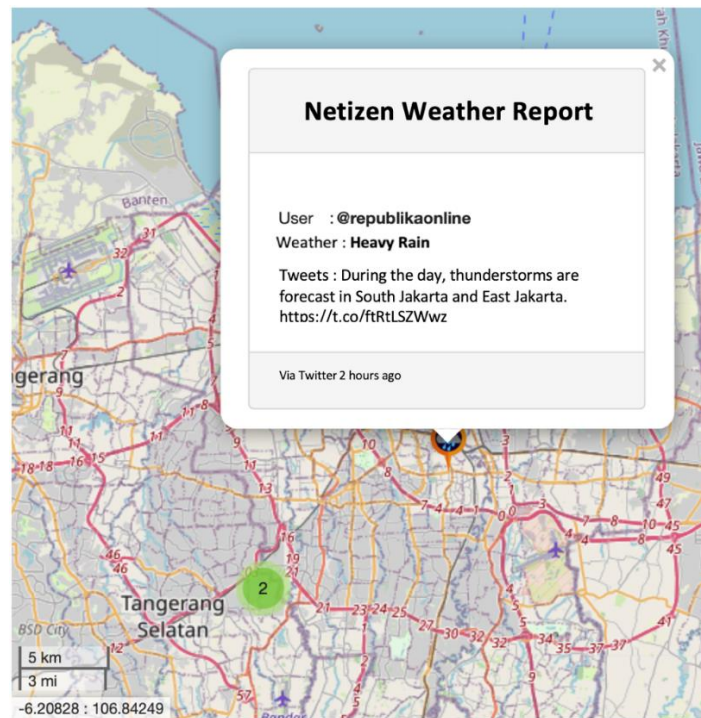


Figure 10. Example of tweets integrated into a geographic information system (GIS) with weather classification and geolocation plotting

3.4. Discussion

This study focused on the comparison of basic machine learning models (SVM, MLR, and MNB) and deep learning models (BERT) for classification texts. The best classification results aim to be applied to a website-based information system. By using a machine learning model, maximum results have been given, especially in the SVM model. Classification results are compared primarily to advanced models like the BERT transformer and classical natural language processing. In recent years, BERT has achieved state-of-

the-art results in a wide range of natural language processing (NLP) tasks [40]. The application of BERT transfer learning using a text dataset on weather has proven to be able to provide good results. BERT is a big neural network architecture, with a huge number of parameters, that can range from 100 million to over 300 million. So, training a BERT model from scratch on a small dataset would result in overfitting [41].

A result of loss validation and training shown in Figure 8 shows evidence of overfitting. Obviously, this can happen when the model used for training is too focused on one training dataset, and so it cannot predict correctly if given another similar dataset [40], [42]. Figure 3 shows the distribution of data for certain training datasets. Twitter's data regarding sunny weather has the highest number in the period between January and May 2019. The dry season begins in April and May. Therefore, March is the transitional period between the rainy and dry seasons. After that, cloudy weather and heavy rain almost equal each other. According to BMKG data, Indonesia enters its rainy season only in January or early February of 2019. Despite the similarity in words between heavy rain, light rain, and rain in the tweets, heavy rain, light rain, and rainy show a small comparison. Because of the ambiguity in the labels, it is difficult to determine which category the tweets belong to. For example: "There is a high probability that rain will drench the entire DKI Jakarta area today. We are expecting light rain to heavy rain in the morning".

In filling out the geolocation, the ambiguity of mentioning the name of the district/city in the sentence tweet also affects the plotting results on Esri Maps. The diversity of ethnic groups in Indonesia causes the use of regional languages to be used in everyday language. According to data from the BPS, in Indonesia there are 1,340 tribes or ethnic groups. Meanwhile, according to the language development and development agency, the number of regional languages in Indonesia was 646 at the beginning of 2017. The similarity between regional languages and regional names in a place affects geolocation filling. This causes the plotting on Esri Maps to not match the area names mentioned in the tweet. For example, the word "karo" can be translated as a regional language from the Central Java Region, and also there is name of district in North Sumatra called "Karo". In this case, the text will be plotted at coordinates 3.1053° N, 98.2651° E on Esri Maps which shows the "Karo" district location.

4. CONCLUSION

The use of Twitter has been proved an effective tool for opinion mining and polling, especially in predicting weather conditions. BERT-based pretrained model is effective for classifying texts from Twitter, based on the dataset used. Identifying data sets before modeling algorithms for different classifications or scenarios is imperative. In addition to categorizing short sentences, BERT-base is useful for other purposes. This model has a yield of 99%. In comparison to automatic classification algorithms (SVM, MNB, and MLR), this accuracy proves to be very good. Based on it, the sentences after the BERT model have been used for geolocation filling tasks from mentioning the name of the district/city in tweets. Tweets are mapped into Esri Maps according to the geolocation points. For future works, the authors will continue mining and analyzing more Twitter data using smart crawling to get a more accurate prediction about weather conditions in Indonesia.

ACKNOWLEDGEMENTS

Author thanks Faiz Ayyas Munawwar for providing us with the illustration in this paper.

REFERENCES




- [1] Y. Marini and K. T. Setiawan, "Indonesia sea surface temperature from TRMM Microwave Imaging (TMI) sensor," *IOP Conference Series: Earth and Environmental Science*, vol. 149, no. 1, 2018, doi: 10.1088/1755-1315/149/1/012055.
- [2] K. E. Trenberth, "Changes in precipitation with climate change," *Climate Research*, vol. 47, no. 1–2, pp. 123–138, 2011, doi: 10.3354/cr00953.
- [3] M. R. Mozell and L. Thachn, "The impact of climate change on the global wine industry: Challenges & solutions," *Wine Economics and Policy*, vol. 3, no. 2, pp. 81–89, 2014, doi: 10.1016/j.wep.2014.08.001.
- [4] R. E. Caraka, S. A. Bakar, M. Tahmid, H. Yasin, and I. D. Kurniawan, "Neurocomputing fundamental climate analysis," *Telkomnika (Telecommunication Computing Electronics and Control)*, vol. 17, no. 4, pp. 1818–1827, 2019, doi: 10.12928/TELKOMNIKA.v17i4.11788.
- [5] R. Caraka, R. C. Chen, T. Toharudin, M. Tahmid, B. Pardamean, and R. M. Putra, "Evaluation performance of SVR genetic algorithm and hybrid PSO in rainfall forecasting," *ICIC Express Letters, Part B: Applications*, vol. 11, no. 7, pp. 631–639, 2020, doi: 10.24507/iceiclb.11.07.631.
- [6] M. G. De Giorgi, A. Ficarella, and M. Tarantino, "Assessment of the benefits of numerical weather predictions in wind power forecasting based on statistical methods," *Energy*, vol. 36, no. 7, pp. 3968–3978, 2011, doi: 10.1016/j.energy.2011.05.006.
- [7] J. Zhuang, T. Mei, S. C. H. Hoi, X. S. Hua, and S. Li, "Modeling social strength in social media community via kernel-based learning," *MM'11 - Proceedings of the 2011 ACM Multimedia Conference and Co-located Workshops*, pp. 113–122, 2011, doi: 10.1145/2072298.2072315.
- [8] S. Z. Jannah, "Clustering and Visualizing Surabaya Citizen Aspirations by Using Text Mining. Case Study: Media Center

- Surabaya,” *Institut Teknologi Sepuluh Nopember*, 2018, [Online]. Available: <https://repository.its.ac.id/58200/>.
- [9] C. Castillo, M. Mendoza, and B. Poblete, “Information credibility on Twitter,” *Proceedings of the 20th International Conference Companion on World Wide Web, WWW 2011*, pp. 675–684, 2011, doi: 10.1145/1963405.1963500.
 - [10] A. Lenhart, K. Purcell, A. Smith, and K. Zickuhr, “Social Media & Mobile Internet Use among Teens and Young Adults. Millennials,” *Pew Internet & American Life Project*, vol. 01, pp. 1–16, 2010, [Online]. Available: <http://eric.ed.gov/?id=ED525056>.
 - [11] R. Rahutomo, A. Budiarto, K. Purwandari, A. S. Perbangsa, T. W. Cenggoro, and B. Pardamean, “Ten-year compilation of #savekpk twitter dataset,” *Proceedings of 2020 International Conference on Information Management and Technology, ICIMTech 2020*, pp. 185–190, 2020, doi: 10.1109/ICIMTech50083.2020.9211246.
 - [12] A. Budiarto, R. Rahutomo, H. N. Putra, T. W. Cenggoro, M. F. Kacamarga, and B. Pardamean, “Unsupervised News Topic Modelling with Doc2Vec and Spherical Clustering,” *Procedia Computer Science*, vol. 179, pp. 40–46, 2021, doi: 10.1016/j.procs.2020.12.007.
 - [13] B. Y. Pratama and R. Sarno, “Personality classification based on Twitter text using Naive Bayes, KNN and SVM,” *Proceedings of 2015 International Conference on Data and Software Engineering, ICODSE 2015*, pp. 170–174, 2016, doi: 10.1109/ICODSE.2015.7436992.
 - [14] L. Yung-Hui, Y. Nai-Ning, K. Purwandari, and L. N. Harfiya, “Clinically applicable deep learning for diagnosis of diabetic retinopathy,” *Proceedings - 2019 12th International Conference on Ubi-Media Computing, Ubi-Media 2019*, pp. 124–129, 2019, doi: 10.1109/Ubi-Media.2019.00032.
 - [15] T. B. Pramono *et al.*, “A Model of Visual Intelligent System for Genus Identification of Fish in the Siluriformes Order,” *IOP Conference Series: Earth and Environmental Science*, vol. 794, no. 1, 2021, doi: 10.1088/1755-1315/794/1/012114.
 - [16] F. E. Gunawan *et al.*, “Multivariate Time-Series Deep Learning for Joint Prediction of Temperature and Relative Humidity in a Closed Space,” *Conference: 2021 International Conference on Computer Science and Computational Intelligence*, 2021.
 - [17] A. A. Hidayat, T. W. Cenggoro, and B. Pardamean, “Convolutional Neural Networks for Scops Owl Sound Classification,” *Procedia Computer Science*, vol. 179, pp. 81–87, 2021, doi: 10.1016/j.procs.2020.12.010.
 - [18] S. Minaee, N. Kalchbrenner, E. Cambria, N. Nikzad, M. Chenaghlu, and J. Gao, “Deep Learning Based Text Classification: A Comprehensive Review,” Apr. 2020, [Online]. Available: <http://arxiv.org/abs/2004.03705>.
 - [19] I. Nurlaila, R. Rahutomo, K. Purwandari, and B. Pardamean, “Provoking Tweets by Indonesia Media Twitter in the Initial Month of Coronavirus Disease Hit,” in *2020 International Conference on Information Management and Technology (ICIMTech)*, Aug. 2020, pp. 409–414, doi: 10.1109/ICIMTech50083.2020.9211179.
 - [20] A. Rietzler, S. Stabinger, P. Opitz, and S. Engl, “Adapt or Get Left Behind: Domain Adaptation through BERT Language Model Finetuning for Aspect-Target Sentiment Classification,” Aug. 2019, [Online]. Available: <http://arxiv.org/abs/1908.11860>.
 - [21] M. Pontiki, D. Galanis, H. Papageorgiou, S. Manandhar, and I. Androustopoulos, “SemEval-2015 Task 12: Aspect Based Sentiment Analysis,” *SemEval 2015 - 9th International Workshop on Semantic Evaluation, co-located with the 2015 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, NAACL-HLT 2015 - Proceedings*, pp. 486–495, 2015, doi: 10.18653/v1/s15-2082.
 - [22] K. Purwandari, J. W. C. Sigalingging, T. W. Cenggoro, and B. Pardamean, “Multi-class Weather Forecasting from Twitter Using Machine Learning Approaches,” *Procedia Computer Science*, vol. 179, pp. 47–54, 2021, doi: 10.1016/j.procs.2020.12.006.
 - [23] B. Komer, J. Bergstra, and C. Eliasmith, “Hyperopt-Sklearn: Automatic Hyperparameter Configuration for Scikit-Learn,” *Proceedings of the 13th Python in Science Conference*, pp. 32–37, 2014, doi: 10.25080/majora-14bd3278-006.
 - [24] S. Robertson, “Understanding inverse document frequency: On theoretical arguments for IDF,” *Journal of Documentation*, vol. 60, no. 5, pp. 503–520, 2004, doi: 10.1108/00220410410560582.
 - [25] S. Sintia, S. Defit, and G. W. Nurcahyo, “Product Codefication Accuracy With Cosine Similarity And Weighted Term Frequency And Inverse Document Frequency (TF-IDF),” *Journal of Applied Engineering and Technological Science (JAETS)*, vol. 2, no. 2, pp. 62–69, 2021, doi: 10.37385/jaets.v2i2.210.
 - [26] I. Aljarah, A. M. Al-Zoubi, H. Faris, M. A. Hassonah, S. Mirjalili, and H. Saadeh, “Simultaneous Feature Selection and Support Vector Machine Optimization Using the Grasshopper Optimization Algorithm,” *Cognitive Computation*, vol. 10, no. 3, pp. 478–495, 2018, doi: 10.1007/s12559-017-9542-9.
 - [27] M. Galar, A. Fernández, E. Barrenechea, H. Bustince, and F. Herrera, “An overview of ensemble methods for binary classifiers in multi-class problems: Experimental study on one-vs-one and one-vs-all schemes,” *Pattern Recognition*, vol. 44, no. 8, pp. 1761–1776, 2011, doi: 10.1016/j.patcog.2011.01.017.
 - [28] B. Heap, M. Bain, W. Wobcke, A. Krzywicki, and S. Schmeidl, “Word Vector Enrichment of Low Frequency Words in the Bag-of-Words Model for Short Text Multi-class Classification Problems,” 2017, [Online]. Available: <http://arxiv.org/abs/1709.05778>.
 - [29] A. Zeggada, F. Melgani, and Y. Bazi, “A Deep Learning Approach to UAV Image Multilabeling,” *IEEE Geoscience and Remote Sensing Letters*, vol. 14, no. 5, pp. 694–698, 2017, doi: 10.1109/LGRS.2017.2671922.
 - [30] Y. Liu *et al.*, “RoBERTa: A Robustly Optimized BERT Pretraining Approach,” 2019, [Online]. Available: <http://arxiv.org/abs/1907.11692>.
 - [31] R. B. Mangolin *et al.*, “A multimodal approach for multi-label movie genre classification,” *Multimedia Tools and Applications*, vol. 81, no. 14, pp. 19071–19096, 2022, doi: 10.1007/s11042-020-10086-2.
 - [32] N. Kampani and D. Jhamb, “Analyzing the role of E-CRM in managing customer relations: A critical review of the literature,” *Journal of Critical Reviews*, vol. 7, no. 4, pp. 221–226, 2020, doi: 10.31838/jcr.07.04.41.
 - [33] A. K. B. Singh, M. Guntu, A. R. Bhimireddy, J. W. Gichoya, and S. Purkayastha, “Multi-label natural language processing to identify diagnosis and procedure codes from MIMIC-III inpatient notes,” *Syria Studies*, vol. 7, no. 1, pp. 37–72, Mar. 2020, doi: 2003.07507v1.
 - [34] C. Sun, X. Qiu, Y. Xu, and X. Huang, “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)*, vol. 11856 LNAI, pp. 194–206, 2019, doi: 10.1007/978-3-030-32381-3_16.
 - [35] J. V. Tu, “Advantages and disadvantages of using artificial neural networks versus logistic regression for predicting medical outcomes,” *Journal of Clinical Epidemiology*, vol. 49, no. 11, pp. 1225–1231, 1996, doi: 10.1016/S0895-4356(96)00002-9.
 - [36] X. Deng, Q. Liu, Y. Deng, and S. Mahadevan, “An improved method to construct basic probability assignment based on the confusion matrix for classification problem,” *Information Sciences*, vol. 340–341, pp. 250–261, 2016, doi: 10.1016/j.ins.2016.01.033.
 - [37] Y. S. Aurelio, G. M. de Almeida, C. L. de Castro, and A. P. Braga, “Learning from mbalanced data sets with weighted cross-entropy function,” *Neural Processing Letters*, vol. 50, no. 2, pp. 1937–1949, 2019, doi: 10.1007/s11063-018-09977-1.
 - [38] D. Tran, H. Mac, V. Tong, H. A. Tran, and L. G. Nguyen, “A LSTM based framework for handling multiclass imbalance in DGA




- botnet detection,” *Neurocomputing*, vol. 275, pp. 2401–2413, 2018, doi: 10.1016/j.neucom.2017.11.018.
- [39] K. Purwandari, A. S. Perbangsa, J. W. C. Sigalingging, A. A. Krisna, S. Anggrayani, and B. Pardamean, “Database management system design for automatic weather information with twitter data collection,” *Proceedings of 2021 International Conference on Information Management and Technology, ICIMTech 2021*, pp. 326–330, 2021, doi: 10.1109/ICIMTech53080.2021.9535009.
- [40] S. González-Carvajal and E. C. Garrido-Merchán, “Comparing BERT against traditional machine learning text classification,” 2020, [Online]. Available: <http://arxiv.org/abs/2005.13012>.
- [41] L. Gong, D. He, Z. Li, T. Qin, L. Wang, and T. Y. Liu, “Efficient training of BERT by progressively stacking,” *36th International Conference on Machine Learning, ICML 2019*, vol. 2019-June, pp. 4202–4211, 2019.
- [42] R. Shokri, M. Stronati, C. Song, and V. Shmatikov, “Membership Inference Attacks Against Machine Learning Models,” *Proceedings - IEEE Symposium on Security and Privacy*, pp. 3–18, 2017, doi: 10.1109/SP.2017.41.

BIOGRAPHIES OF AUTHORS






Kartika Purwandari    received the bachelor’s degree in Information Technology from Brawijaya University and the master’s degree in Computer Science from National Central University Taiwan. She is currently a lecturer at Computer Science Department in Bina Nusantara University, Jakarta, Indonesia. She is also a lecturer specialist S2 of basic programming in Bina Nusantara University since December 2021. In the past 2 years ago, she was become a research assistant at Bioinformatics and Data Science Research Center (BDSRC) Bina Nusantara University. She has developed programming based on AI and bioinformatics by joining the colorectal cancer project since she joined BDSRC. Furthermore, she is also active in participating with AI projects in BDSRC to help in processing data about lidar, air quality, crowd counting, fishery image, text, and pap smear. She can be contacted at email: kartika.purwandari@binus.edu.






Tjeng Wawan Cenggoro    is an AI researcher whose focus is in the development of deep learning algorithms for application in computer vision, natural language processing, and bioinformatics. He has led several research projects that utilize deep learning for computer vision, which is applied to indoor video analytics and plant phenotyping. He has published over 20 peer-reviewed publications and reviewed for prestigious journals such as Scientific Reports and IEEE Access. He also holds 2 copyrights for AI-based video analytics software. He received his master’s degree in Information Technology from Bina Nusantara University as well as bachelor’s degree in Information Technology from STMIK Widya Cipta Dharma. He is also certified instructor at NVIDIA Deep Learning Institute. He can be contacted at email: wcenggoro@binus.edu.



Join Wan Chanlyn Sigalingging    holds a Bachelor of Engineering in Electrical and Electronics Engineering from Sumatera Utara University, Master of Science in Computer Science and Information Engineering from National Central University. His research on master degree has focused on He is currently working with the meteorology, climatology, and geophysics agency at Indonesia. He is a member of the database department. His research areas of interest include data engineer, NLP, image processing, artificial intelligent, and digital signal processing. He can be contacted at email: join.wan.chanlyn@bmkg.go.id.



Dr. Bens Pardamean    has over thirty years of global experience in information technology, bioinformatics, and education, including a strong background in database systems, computer networks, and quantitative research. His professional experience includes being a practitioner, researcher, consultant, entrepreneur, and lecturer. His current research interests are in developing and analyzing genetic data in cancer studies and genome-wide association studies (GWAS) for agriculture genetic research. After successfully leading the Bioinformatics Research Interest Group, he currently holds a dual appointment as the Director of Bioinformatics & Data Science Research Center (BDSRC) and as a Professor of Computer Science at the University of Bina Nusantara (BINUS) in Jakarta, Indonesia. He earned a doctoral degree in informative research from the University of Southern California (USC), as well as a master’s degree in computer education and a bachelor’s degree in computer science from California State University, Los Angeles. He can be contacted at email: bpardamean@binus.edu.