

# Virality classification from Twitter data using pre-trained language model and multi-layer perceptron

Jeffrey Junior Tedjasulaksana, Abba Suganda Girsang

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

## Article Info

### Article history:

Received Feb 7, 2024

Revised Apr 19, 2024

Accepted May 12, 2024

### Keywords:

BERT

Pre-trained language model

Twitter

Virality classification

Virality features

## ABSTRACT

Twitter is one of the well-known text-based social media that is often used to disseminate content. According to Katadata, Indonesia ranked fifth in the world in 2023. So many people or organizations want to make tweets go viral. Therefore, this research aims to develop a model that uses tweet data from the Indonesian language Twitter social media to categorize the level of virality. There are several tasks in classifying the level of virality, such as upsampling data, predicting sentiment and emotion, and text embedding. Upsampling data was carried out because the dataset used was an imbalanced dataset. Data upsampling, emotions, and text embedding is carried out using the bidirectional encoder representation from transformers (BERT) model. Meanwhile, sentiment prediction uses the Ro-bustly optimized BERT pretraining approach (RoBERTa). The results of text embedding, sentiment, emotion, will be combined with Twitter metadata then all features will be fed into the multi-layer perceptron (MLP) model to classifying the level of virality which is divided into 3 classes based on the number of retweets, namely low, medium and high. The proposed method produces an F1-score of 49% and an accuracy of 95% and performs better than the baseline model.

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



## Corresponding Author:

Jeffrey Junior Tedjasulaksana

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

Jakarta, Indonesia, 11480

Email: jeffrey.tedjasulaksana@binus.ac.id

## 1. INTRODUCTION

Recently, social media has emerged as a popular medium for individuals and organizations to rapidly broadcast information to a broader audience [1]. Twitter, a widely used text based social media is frequently utilized for the rapid spread of viral information. According to Katadata, Indonesia is projected to be the fifth largest country in terms of Twitter users by 2023, with a total of 24 million users. Twitter offers a trending section that allows users to view the current topics of discussion. Twitter offers a trending section that allows users to view the current topics of discussion. Hence, numerous individuals and organizations are currently seeking to engage in hot topics in order to gain significant attention for their tweets. Gaining insight into the determinants that impact the level of virality of a tweet holds significant implications in various domains, such as marketing, political campaigns, journalism, and information distribution. Viral categorization study can utilize the analysis of specific characteristics or attributes of a tweet, such as its text or related metadata, to discover discernible patterns that might accurately predict the level of virality a tweet is likely to achieve [2].

Natural language processing (NLP) is a branch of artificial intelligence (AI) and linguistics that studies how computers interact with human language. This requires the creation of computational models and

algorithms that allow computers to understand and interpret so as to produce meaningful and useful output [3]. So NLP can be used in the viral level classification process from data in the form of text such as Twitter.

The transformer architecture is quickly emerging as the prevailing choice for NLP. This model's effectiveness is enhanced by the incorporation of larger datasets and architecture sizes, enabling parallel training and capturing longer sequence elements. Consequently, it paves the way for the development of more comprehensive and efficient language models [4]. Previous research on viral level classification utilized a model that has a transformer architecture by using RoBERTa methodology to conduct sentiment prediction on English language twitter data. Text from tweets is also utilized by embedding text using BERTweet. The analysis considered many factors such as the number of hashtags, mentions, followers, following, verified status, text length, text embeddings, and sentiment of tweets. Afterwards, these characteristics will be inputted into the multi-layer perceptron (MLP) model to facilitate the classification procedure [5]. Previous study has not adequately explored the parameters involved, such as the utilization of emotions detected from tweets as a criteria for categorizing levels of virality. Another previous research has established a correlation between emotion classes and degrees of virality [6]. Bidirectional encoder representations from transformers (BERT) is a pre-trained model that captures the contextual relationship of each word in a text in both forward and backward directions. The BERT model considers the words before and after each word in a sentence. In this case, BERT emotion is utilized to predict emotions from tweet data such as sadness, joy, love, anger, fear, or surprise as an additional feature in the MLP model [7].

In previous research, BERTweet was used to embed text [5], which is a pre-trained language model created especially for English tweets. It was trained using the RoBERTa pre-training technique and is based on the BERTbase architecture. An extensive dataset of 850 million English tweets was used to train BERTweet [8]. However, the research carried out will use Indonesian language which is pre-processed first and then translated into English, so that there are no English slang words in the data used. Therefore, Text embedding in the research was carried out using BERT because BERT trained on a BookCorpus which a large dataset of books and their corresponding titles and also using English wikipedia [9]. In the data used there is also imbalanced data, so data with minor classes will be upsampled using BERT. BERT insert performed next word prediction instead of next sentence prediction, which allowed the model to perform insertion by prediction, whereas previous word embedding models pick one word randomly for insertion [10], [11].

The model created in this research uses the outcomes of predicted emotions, sentiment, and text embedding will be merged with additional numerical features, including follower count, following count, likes count, responses count, account verification status, text length, number of hashtags, and number of mentions. Afterwards, all of these characteristics will serve as input for the MLP model in the process of categorizing the levels of virality. The reason why MLP is employed as a classifier is because to its ability to extract complex and abstract feature representations from data [12]. The level of virality will be categorized into three classes, which are determined by the number of retweets: low (0–1), medium (2–20), and high (more than 20). The study will use Twitter data in the Indonesian language due to a lack of research on classifying virality levels specifically in the Indonesian context.

There have been several previous studies related to the classification of virality levels, one of which is research conducted by (Rameez et al., 2022) [5]. The data used is data collected by the researchers themselves, namely 330,000 Twitter data. The sentiment features used are predicted using RoBERTa and BERTweet to carry out the text embedding process. The numerical features such as hashtags, mentions, followers, following, verified status, and text length are normalized using min-max scalar. The results of sentiment predictions and text embedding are combined with other numerical features. MLP model with 1 hidden layer and batch size of 32 as a classifier to classify virality levels. The virality level used is based on the number of retweets of each tweet. The performance results of the model in classifying virality levels were 49% for accuracy and 52% for F1-score.

The study conducted by [6] the factors that influence Twitter users' distribution of COVID-19 content, emotion analysis, text length, number of mentions, and number of hashtags. The model will undergo testing using a machine learning linear regression model to evaluate the impact of each utilized attribute. The data used is 57,000 tweets that mentioned COVID-19 and collected from 29<sup>th</sup> March 2020 till 29<sup>th</sup> April 2020 (one month). This research also uses named entity relationship (NER) by counting the number of people, number of locations, and number of organizations. Emotion prediction in this research using National Research Council Canada (NRC) word-emotion lexicon. A research study discovered a correlation between emotional features and the classification of virality levels.

Other research related to the classification of virality levels was also made by [13]. The data used in this research is Twitter data from April 2015 to June 2016 totaling 100 million data with a virality level based on the number of retweets on the tweet. The features used are the number of mentions, number of hashtags, number of likes, tweet date, number of days since the tweet was posted, number of tweets made, number of followers and following. Several models were tested for classification such as recursive partitioning, random

forest, gradient boosting, and logistic models with the best results using the random forest model with an accuracy value of 91.5% and an F1-score of 90.6%.

In research conducted by [14] predict the number of retweets from tweets by comparing 2 text pre-processing techniques, namely bag of words (TFIDF) and word embeddings (Doc2Vec). The preprocessing results will be tested using several machine learning techniques such as logistic regression, support vector machine (SVM), random forest, neural networks, and multinomial Naive Bayes. Apart from using text, the model created also uses the date the tweet was created and the number of likes as additional features in the model. The results show that the combination of Doc2Vec and random forest pre-processing produces an accuracy of 62.67%. Prasad *et al.* [15] did study on the classification of virality levels specifically in the context of mobile phones from twitter. The utilized features include the length of the text, the count of retweets and comments, the application of POS tagging, and the implementation of a decision tree as a classifier for categorizing levels of virality. The research carried out also carried out sentiment analysis as an additional feature in the model.

The aim of this paper is to categorize virality into three levels low, medium, and high utilizing various pre-trained language models and to assess the impact of emotion feature to the model. Pre-trained models that are used in this research are BERT for text embedding, RoBERTa for sentiment analysis, BERTinsert for data augmentation, and BERTemotion for emotion prediction. To assess the performance of the model, accuracy, precision, recall, and F1-score matrices will be used. The introduction will set the stage for categorizing virality levels using pre-trained language models, highlighting their importance in social media analytics. The method section will detail the research design, data collection, and evaluation metrics. Results and discussion will present findings, analyzing model performance and implications. The conclusion will summarize key insights and propose future research directions.

## 2. METHOD

Unlike earlier studies, this approach to assessing virality has not included parameter exploration through the addition of emotional feature. Then modifications to the text embedding process were also carried out using BERT [5]. So in this research, the proposed method will combine numerical features from tweets with the results of sentiment, emotion and text embedding predictions. The results of the combination of these features will be input for the MLP model to determine the level of virality, namely low, medium or high. Flowchart of this research could be seen in Figure 1.

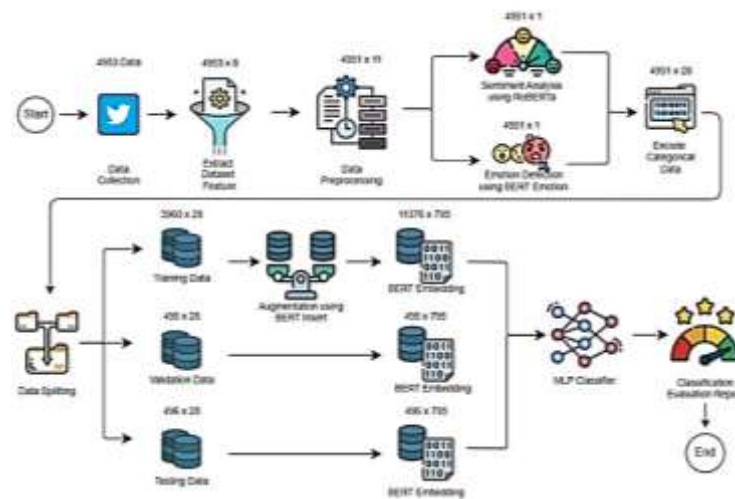


Figure 1. Proposed method architecture

### 2.1. Dataset

The dataset used is tweet data collected using the Apache Solr server on Twitter in 2021 in July, totaling 4,953 data using Indonesian. Data retrieval via the server is done by connecting JDBC (Java database connectivity driver) from Twitter to the Apache Solr server so that the Twitter database and server can be connected. Because there are too many metadata features in the data, some metadata will be taken as features that will be used in the process of classifying the virality level of tweets. The features used in the dataset are features that influence the level of virality. Detailed features of the tweet data used can be seen in Table 1.

Table 1. Dataset features detail

No	Features	DataType	Unique	Unique sample	Unique translated sample
1	Topic	Object	12	<i>Bencanaalam, kuliner</i>	Natural disaster, culinary
2	Tweet	Object	4,952	Ryan Reynolds <i>menceritakan keahlian istrinya ...</i>	Ryan Reynolds shared his wife's expertise...
3	Follower	Int	4,952	17010332, 2246972	-
4	Following	Int	408	178, 105	-
5	Like	Int	432	1058, 2242	-
6	Retweet	Int	134	17, 0	-
7	Reply	Int	105	142, 6	-
8	Verified	Object	2	False, true	-

In the Table 1, there are 8 features selected based on previous research related to virality classification. Each feature obtained has no missing values. The unique column displays the number of unique values or classes in each feature, for example in the topic feature there are 12 unique values, meaning that there are 12 topic classes from the data obtained. Meanwhile, the unique sample column displays unique data samples from each feature. The DataType column displays the data type of the feature, where there are features with object or string data types so that the encoding process will be carried out using one hot encoding on the feature.

## 2.2. Data preprocessing

From the overall dataset, which is still in JSON form, tweet data containing text posted by users will be taken. The captured tweets will be pre-processed with the aim of converting unstructured data into structured data by cleaning the raw data so that it can be processed by the model and produce good performance. The following is the flow of pre-processing:

The first step in the preprocessing stage is checking tweets that have duplicates and deleting tweet data that has duplicates. The next stage is to calculate the length of the tweet or the number of words in the tweet, then the number of hashtags and mentions in the tweet is also calculated. The results of the calculations will later become additional features to be included in the MLP model. In tweets posted by users, there is usually the name of the account they want to target or what is usually called a mention. The account name does not provide any meaning to the tweet so it can reduce the performance of the model in carrying out the classification process so that the account name in the tweet is deleted. Likewise, the use of hashtags and URLs in tweets does not provide deep meaning in sentences so hashtags in tweets are also deleted.

ASCII is a computing standard on computers for representing numbers, letters, and punctuation marks in the form of integers from 0 to 127, so non-ascii is defined as a form of emoji or unusual symbols in text. Deletion of non-ascii is done because it does not really provide meaningful information in the text. Changing capital letters to lowercase letters makes it easier for the model to understand the meaning of words because every word that the model understands has lowercase letters. In a tweet that usually uses informal language, there will be many words that use 'Slang', namely language that is often used in everyday conversation that does not follow the rules of formal grammar or written language, so the tweet will be pre-processed so that the language is slang. Tweets will be converted into formal language according to the rules of grammar or written language. The next stage is labeling the virality level of tweets which will later be used as targets for the model. The labeling carried out will be divided into 3 classes, namely low class with number of retweets of 0 and 1, medium class with number of retweets of 2 to 20, and high class with number of retweets of more than 21. Data has 11 features after preprocessing stage.

## 2.3. Sentiment analysis

Sentiment analysis was done after preprocessing. Tweet feature was used for sentiment analysis. The results of sentiment analysis will be used as one of the parameters for classifying virality levels. Before conducting sentiment analysis, tweet data that has gone through the pre-processing stage will be translated into English first. This is done because the pre-trained model in RoBERTa uses English. During the pre-trained process RoBERTa uses a large data corpus called OpenWebText, which consists of 38 GB of text from the internet [16], [17].

The first process of sentiment analysis process with RoBERTa is tokenization. Tokenization is breaking down text into smaller levels, namely into words. This word breakdown or separation is intended so that the model can understand the meaning of each word in detail. The tokens will be converted into a numerical vector representation or what is usually called embedding so that it can be processed by the model. Added position encoding to the embedding to consider word order. The token results that have been encoded

and embedded will be entered into the RoBERTa model for a sentiment analysis process using the attention mechanism by examining the relationship between all tokens in the sentence so that the model can understand the context of each word. The results of sentiment analysis will produce 3 sentiment classes, namely positive, negative, and neutral which will be used as additional features in the MLP model.

#### 2.4. Emotion detection

Tweet feature was used for emotion detection. The results of emotion detection will also be one of the features used in classifying virality levels. Emotion detection uses BERT which also has a transformer architecture. BERT is pre-trained using BookCorpus data (800 M words) [18] and English Wikipedia (2,500 words) and uses masked language model (MLM) and next sentence prediction (NSP) mechanisms so that the model can understand the context of a sentence [9]. The BERT emotion model is also fine-tuned to be able to make predictions with emotion labels [19]. The input to the model is in the form of tweet text which has been pre-processed and translated into English. The results of the emotion detection process are 6 classes of emotions, namely sadness, joy, love, anger, fear, and surprise.

#### 2.5. Data encoding and data splitting

After sentiment analysis and emotion detection have done, data now has 13 features. Data still contains several categorical features. Features that still have categorical data types will be encoded using one hot encoding, except for the virality feature, the encoding process is carried out using ordinal encoding. Encoding categorical variables preserves the information contained within them while converting them into a format that algorithms can understand [20]. After the encoding process, normalization is carried out using min-max scalar because there are differences in the range of values of the features used, for example in the number of followers with text length. This ensures that all features contribute equally to the analysis, preventing features with larger scales from dominating the model's training process [21]. The next process is data splitting. Data is splitted into 80% training, 10% validation, and 10% testing.

#### 2.6. Data augmentation

The BERT model is utilized to upsample minority data in the medium and high classes by replacing or adding words according to the overall context of the sentence. During the text embedding process, BERT employs NSP, while during augmentation, it uses next word prediction and leverages MLM to mask words to be replaced or added. The augmentation process will be conducted on minority classes. Table 2 shows augmentation data example and Table 3 shows the comparison of the data counts before and after augmentation in each class.

Table 2. Augmentation data example

Input	Features
Original text	Oxygen is empty at the Putussibau hospital, West Kalimantan
Augmentation text 1	Terminal oxygen is empty at of the nearby Putussibau central hospital, West East Kalimantan
Augmentation text 2	All oxygen tank is empty at nearby the Putussibau hospital, of West Kalimantan

Table 3. Augmentation data result

Virality class	Before augmentation	After augmentation
Low	4,772	4,772
Medium	52	3,832
High	129	3,759

#### 2.7. Text embedding

BERT, which has 12 transformer blocks, 768 hidden units, and 12 self-attention heads, is used to carry out the text embedding process. It results in a 768 dimension [CLS] embedding [9]. Text embedding has two stages, that is tokenization and convert the token to numerical vector. Tokenization is breaking down text into smaller levels, namely into words. This word breakdown or separation is intended so that the model can understand the meaning of each word in detail. The tokens will be converted into a numerical vector representation or what is usually called embedding so that it can be processed by the model.

#### 2.8. Classification

The results of sentiment analysis, emotion prediction, and text embedding will be combined with other numerical features such as follower count, following count, likes count, responses count, account

verification status, text length, number of hashtags, and number of mentions. All 795 features will be input to the MLP model for classification of virality levels, namely low, medium, or high. For details of the features used in the MLP model, see Table 4.

Table 4. Features for MLP model

Input	Features
1	Topic
2	Follower
3	Following
4	Like
5	Reply
6	Verified
7	Number of hashtag
8	Number of mention
9	Text length
10	Sentiment
11	Emotion
12	Virality
13	BERT embedding 1
...	...
...	...
...	...
780	BERT embedding 768

The loss function used in the MLP model is sparse categorical cross entropy. The loss function is used to measure how far the prediction results are from the expected target [22]. Sparse categorical cross entropy is used because it is suitable for classification with multiclass targets [23].

Then the optimizer used in the MLP model is adaptive moment estimation or usually called Adam. The main goal of the optimizer is to help the model produce fast and efficient convergence during the training process so that it can understand the pattern of each feature in each given class. Adam has the unique ability of learning from previous gradients for each parameter, then Adam also helps the model to get out of the local minimum and accelerates convergence, as well as good bias correction [24]. From the proposed model, settings or adjustments will be made to the parameters contained in the model or what is usually called hyperparameters tuning. The hyperparameters tuning process will be carried out by testing all hyperparameter combinations using training data and testing data. Later the best parameters will be taken from the results of the hyperparameters tuning process. Table 5 shown the details of the hyperparameters that will be used in creating the MLP model architecture to get the best performance results.

Table 5. Hyperparameter tuning MLP

Hyperparameter	Hyperparameter value
Hidden layer sizes	[1, 3]
Batch size	[4, 8, 16, 32]
Learning rate	[0.00001, 0.001, 0.01]
Loss function	Sparse categorical crossentropy
Optimizer	Adam
Epoch	100
Activation	[ReLU, Tanh]

**2.9. Evaluation**

After getting the best hyperparameters from the model in the training and testing process, then evaluate the performance of the model using testing data. Model evaluation will use four metrics, namely accuracy, precision, recall, and F1-score. The metric evaluation results are obtained based on the results of the confusion matrix. Table 6 shown the multiclass confusion matrix table that will be used in the research.

Table 6. Multiclass confusion matrix

Multiclass confusion matrix		Predicted		
		Low	Medium	High
Actual	Low	A	B	C
	Medium	D	E	F
	High	G	H	I

The multiclass confusion matrix is a tabular representation that assesses the performance of a classification model in the context of a problem with three classes: low, medium, and high [25]. This matrix is organized such that each row corresponds to the actual class of instances, and each column represents the class predicted by the model. The elements within the matrix indicate the count or probability of instances falling into specific combinations of actual and predicted classes.

In the matrix, the entry labeled 'A' represents the instances where the actual class is low, and the model correctly predicted them as low. This is a measure of the true positives for the low class. The entry labeled 'B' signifies instances where the actual class is low, but the model predicted them as medium. This represents false positives for the medium class in relation to the instances of the low class. The entry labeled 'C' corresponds to instances where the actual class is low, but the model predicted them as high, indicating false positives for the high class concerning the instances of the low class. This pattern continues for the entries 'D' through 'I,' capturing instances of correct predictions (true positives) and instances where the model made errors in predicting different classes (false positives). Based on Table 2, when the low class is a positive class then labeled 'A' will be a true positive value, then labeled 'D' and 'G' will be a false positive value. Meanwhile, labeled 'B' and 'C' are false negative values and labeled 'E', 'F', 'H', 'I' are true negative values. Likewise for other virality classes. Next, carry out precision calculations based on the results of the confusion matrix. Precision is an evaluation metric to see the performance of a classification model by measuring the proportion of positive predictions whether they correspond to the actual class, namely positive [26]. So, in the case of this research to avoid data that is not viral but is predicted to be in the viral class. In (1) is an equation of precision.

$$Precision = \frac{TP}{TP+FP} \quad (1)$$

Recall is an evaluation metric used in classification tasks, to measure the proportion of actual positive cases that were correctly identified by the model. In other words, recall calculates the ratio of true positives to the sum of true positives and false negatives [27]. The following (2) is the formula for recall.

$$Recall = \frac{TP}{TP+FN} \quad (2)$$

F1-score is a combination of the harmonic mean values of precision and recall, thus providing a more complete picture of model performance in classification tasks [28]. In (3) are the equations of the F1-score process.

$$F1\ Score = 2 \times \frac{Precision \times Recall}{Precision+Recall} \quad (3)$$

Accuracy at the model evaluation stage is a measure that describes the extent to which the classification model can correctly identify or predict the class of a given data sample. Accuracy is the percentage of correct predictions compared to the total amount of data evaluated [29]. The formula for accuracy can be seen in (4).

$$Accuracy = \frac{TP+TN}{TP+TN+FP+FN} \quad (4)$$

The results of calculating model performance from precision, recall, F1-score and accuracy values will be compared without using the attributes resulting from sentiment analysis and emotion detection. Then it is compared with the model performance when using attributes from sentiment analysis results and emotion detection results.

### 3. RESULTS AND DISCUSSION

The results of the research conducted found that if tweets containing negative sentiments tend to get high virality results which can be seen in Figure 2, while tweets that have anger and joy emotions tend to have medium and high virality levels which can be seen in Figure 3. Then the addition of emotion features improves the performance of the proposed model with an increase in performance of 4% for precision, 1% for recall, 3% for F1-score, and 1% for accuracy which can be seen in Table 7. The pre-trained language model BERT is also better than BERTweet in performing text embedding because the pre-trained process carried out by BERT is more in accordance with the data used than BERTweet.

**3.1. Exploratory data analysis**

Exploratory data analysis was carried out to see the characteristics of the data. The data analysis carried out looked at the distribution of sentiment and emotion data at each level of virality. Figure 2 shown the distribution of sentiment data in each virality class and Figure 3 shown distribution of emotional data predicted using BERT emotion. The emotional feature is an additional feature made from previous research. According to Figure 2, in the low virality class where the number of retweets ranges from 0 to 1, tweets with a neutral sentiment constitute a significant percentage of 64.19%. This suggests that tweets created with a neutral sentiment have a high likelihood of not gaining virality. The medium virality class (2 to 20 retweets), there is an increase in the percentage of tweets with negative sentiment compared to the low virality class. In the high virality class (above 20 retweets), the percentage of tweets with negative sentiment is notably high at 47.2%. This implies that when users create tweets with a negative sentiment, there is a greater chance of the tweets going viral.

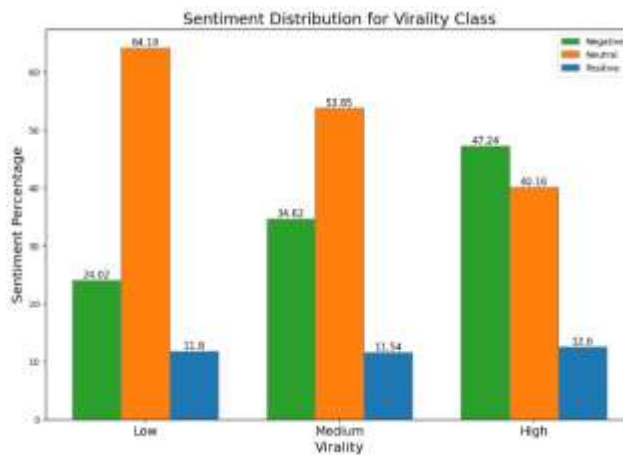


Figure 2. Sentiment distribution

According to Figure 3, the analysis reveals that Joy emerges as the predominant emotion in low-virality tweets, constituting almost half of the emotional content. While Anger and Fear also make notable contributions, their presence is relatively less pronounced compared to Joy. Sadness, surprise, and love exhibit lower percentages, indicating a more balanced emotional mix, with Joy taking the lead. In the medium virality class, the emotional landscape resembles that of low-virality tweets, with Joy and Anger being the dominant emotions. The absence of surprise and love suggests a more focused emotional content, with a substantial emphasis on Joy and Anger. High-virality tweets exhibit a similar pattern, where Joy and Anger stand out as the dominant emotions. Although fear, sadness, surprise, and love contribute to a lesser extent, their inclusion adds emotional diversity to high-virality tweets.

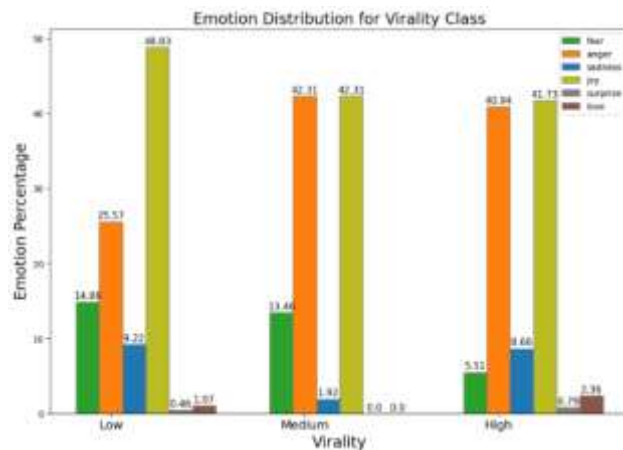


Figure 3. Emotion distribution



### 3.2. Hyperparameter

Hyperparameter tuning is carried out using the grid search method, where the grid search method will search for all possible combinations of the given hyperparameters. Table 7 shows the hyperparameters used in the MLP model based on the results of the grid search. The outcomes of the hidden layer hyperparameter with a count of 1 exhibit superior performance compared to having 3 hidden layers. This is attributed to the relatively lower complexity of patterns within the dataset, evident in the significantly smaller dataset size of 4,953 data rows, as opposed to the baseline with 330,000 data rows. The dataset size used in this research aligns well with a batch size of 4 and a learning rate of 0.0001, as it enables the model to evade convergence errors and facilitates more frequent weight updates.

Table 7. MLP selected hyperparameter

Hyperparameter	Hyperparameter value
Hidden layer sizes	1
Batch size	4
Learning rate	0.0001
Loss function	Sparse categorical crossentropy
Optimizer	Adam
Epoch	100
Activation	Tanh

### 3.3. Evaluation metric

The tuned hyperparameters will be used in the proposed MLP model. Table 8 shows the performance result of a model with several different feature selection schemes. Experimental results show that the proposed model performs better than the baseline model in classifying virality levels using Indonesian language data. Then look at the influence of other features such as the influence of text embedding features and emotion features on the model when deleting these features.

Table 8. Evaluation metric

Method	Main features			Precision	Model performance		
	Sentiment	Emotion	Text Embedding		Recall	F1-score	Accuracy
Baseline [5]	RoBERTa	-	BERTweet	0.49	0.35	0.37	0.35
Proposed model (without emotion)	RoBERTa	-	BERT	0.44	0.50	0.46	0.94
Proposed model	RoBERTa	BERT emotion	BERT	0.48	0.51	0.49	0.95

In Table 8, it is evident that the proposed model outperforms the baseline model in terms of recall, F1-score, and accuracy. This improvement can be attributed to the choice of text embedding using BERT, which proves to be more suitable than BERTweet. The dataset used is in Indonesian, pre-processed, and translated into English, resulting in a more standardized language. BERT's pre-trained dataset with standard language is advantageous compared to BERTweet, which relies on Twitter data containing numerous English slang terms. Additionally, the enhanced performance of the proposed model is attributed to the incorporation of emotional features. The addition of emotion features improves the performance of the proposed model with an increase in performance of 4% for precision, 1% for recall, 3% for F1-score, and 1% for accuracy. Tweets with specific emotions tend to go viral, aiding the model in generalizing during training. The significance of these emotional features is highlighted by the observed performance decline when they are omitted from the model. The results of the research conducted show that the proposed model has shown excellent performance results because research related to virality classification using Indonesian tweets has never been done before. However, this research also has several limitations that need to be considered, such as the use of tweets Indonesian language from 2021, virality labelling based on the number of retweets, and a limited amount of data, totaling 4,953 data.

The modeling carried out in this study by adding emotion features, selecting BERT embedding text, and using MLP as a classifier provides much better results when using Indonesian twitter data than the baseline model in classifying virality levels. Feature selection in the proposed model can be tested more deeply because when the data is first crawled from social media there are many features. But in the model proposed in this study, the features used are the same as the baseline model by adding emotion features. So that in future research, deeper feature selection testing can be carried out. Future research can also add the amount of data and use data from other social media.

#### 4. CONCLUSION

In this study, the proposed model incorporates text embedding, sentiment, and emotion derived from tweets, along with metadata such as the number of followers and likes. The dataset comprises 4,953 Indonesian language Twitter data, and imbalanced handling is implemented using BERT insert. Sentiment prediction is executed using RoBERTa, while emotion prediction and text embedding utilize BERT. The MLP classifier yields commendable performance, surpassing the baseline with precision values of 0.48, recall of 0.51, F1-score of 0.49, and accuracy of 0.95. The research underscores the efficacy of BERT-based text embedding on Indonesian data and demonstrates the informative value of emotional features, indicating that tweets with specific emotions increase the likelihood of virality. This research also found that emotion features have impact to the model. The research that has been conducted can help organizations in conducting marketing and promotions so that campaigns are more effective and help journalists or information media in identifying content that has the potential to go viral through Twitter social media, especially in Indonesia.

Future research could extend the model's applicability to classify virality levels on other platforms such as Facebook or news articles. The model's capabilities may be further enhanced by exploring ensemble models, combining text embedding from BERT, RoBERTa, XLnet, or other models. Additional developments may involve investigating other metadata, such as posting date and total tweets made by users. Testing the classifier with alternative models like long short-term memory (LSTM), one-dimensional convolutional neural networks (1D CNN), and expanding the dataset with better class balance can significantly contribute to improving the model's virality level classification.





#### REFERENCES

- [1] J. A. Obar and S. Wildman, "Social media definition and the governance challenge: an introduction to the special issue," *Telecomm Policy*, vol. 39, no. 9, pp. 745–750, 2015, doi: 10.1016/j.telpol.2015.07.014.
- [2] A. M. Kaplan and M. Haenlein, "Users of the world, unite! the challenges and opportunities of social media," *Business Horizons*, vol. 53, no. 1, pp. 59–68, Jan. 2010, doi: 10.1016/j.bushor.2009.09.003.
- [3] K. R. Chowdhary, "Fundamentals of artificial intelligence," *Springer India*, 2020, doi: 10.1007/978-81-322-3972-7.
- [4] A. Vaswani *et al.*, "Attention is all you need," in *31st Conference on Neural Information Processing Systems (NIPS 2017)*, 2017, pp. 5999–6009.
- [5] R. Rameez, H. A. Rahmani, and E. Yilmaz, "ViralBERT: a user focused BERT-based approach to virality prediction," in *UMAP2022 - Adjunct Proceedings of the 30th ACM Conference on User Modeling, Adaptation and Personalization*, Association for Computing Machinery, Inc, Jul. 2022, pp. 85–89, doi: 10.1145/3511047.3536415.
- [6] K. Nanath and G. Joy, "Leveraging Twitter data to analyze the virality of COVID-19 tweets: a text mining approach," *Behaviour and Information Technology*, vol. 42, no. 2, pp. 196–214, 2023, doi: 10.1080/0144929X.2021.1941259.
- [7] H. Choi, J. Kim, S. Joe, and Y. Gwon, "Evaluation of BERT and Albert sentence embedding performance on downstream NLP tasks," in *Proceedings - International Conference on Pattern Recognition*, Institute of Electrical and Electronics Engineers Inc., 2020, pp. 5482–5487, doi: 10.1109/ICPR48806.2021.9412102.
- [8] D. Q. Nguyen, T. Vu, A. T. Nguyen, and V. Research, "BERTweet: a pre-trained language model for English tweets," in *Proceedings of the 2020 EMNLP (Systems Demonstrations)*, Nov. 2020, pp. 9–14. [Online]. Available: <https://pyip.org/project/emoji>
- [9] J. Devlin, M. Chang, K. Lee, K. Toutanova, "BERT: pre-training of deep bidirectional transformers for language understanding," 2019, *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, vol. 1, pp. 4171–4186, doi: 10.18653/v1/N19-1423.
- [10] A. J. Keya, M. A. H. Wadud, M. F. Mridha, M. Alatiyyah, and M. A. Hamid, "AugFake-BERT: handling imbalance through augmentation of fake news using BERT to Enhance the Performance of Fake News Classification," *Applied Sciences (Switzerland)*, vol. 12, no. 17, Sep. 2022, doi: 10.3390/app12178398.
- [11] M. F. Hasani, K. Jingga, K. M. Suryaningrum and M. Caleb, "Comparison of pretrained BERT Embedding and NLTK approach for easy data augmentation in research title document classification task," *2022 IEEE Creative Communication and Innovative Technology (ICCIT)*, Tangerang, Indonesia, 2022, pp. 1-6, doi: 10.1109/ICCIT55355.2022.10118992.
- [12] W. Samek, G. Montavon, S. Lapuschkin, C. J. Anders, and K. R. Müller, "Explaining deep neural networks and beyond: a review of methods and applications," *Proceedings of the IEEE*, vol. 109, no. 3, pp. 247–278, Mar. 2021, doi: 10.1109/JPROC.2021.3060483.
- [13] P. Nesi, G. Pantaleo, I. Paoli, and I. Zaza, "Assessing the reTweet proneness of tweets: predictive models for retweeting," *Multimedia tools and applications*, vol. 77, no. 20, pp. 26371–26396, Oct. 2018, doi: 10.1007/s11042-018-5865-0.
- [14] I. Daga, A. Gupta, R. Vardhan, and P. Mukherjee, "Prediction of likes and retweets using text information retrieval," in *Procedia Computer Science*, Elsevier B.V., 2020, pp. 123–128, doi: 10.1016/j.procs.2020.02.273.
- [15] B. S. P. Prasad, R. S. Punith, R. Aravindhan, R. Kulkarni and A. R. Choudhury, "Survey on prediction of smartphone virality using twitter analytics," *2019 IEEE International Conference on System, Computation, Automation and Networking (ICSCAN)*, Pondicherry, India, 2019, pp. 1-5, doi: 10.1109/ICSCAN.2019.8878839.
- [16] Y. Liu *et al.*, "RoBERTa: a robustly optimized BERT pretraining approach," Jul. 2019, [Online]. Available: <http://arxiv.org/abs/1907.11692> (Accessed: Feb. 15, 2024)
- [17] W. Suwarningsih, R. A. Pratama, F. Y. Rahadika, and M. H. A. Purnomo, "RoBERTa: language modelling in building Indonesian question-answering systems," *Telkonnika (Telecommunication Computing Electronics and Control)*, vol. 20, no. 6, pp. 1248–1255, Dec. 2022, doi: 10.12928/TELKOMNIKA.v20i6.24248.
- [18] Y. Zhu *et al.*, "Aligning books and movies: towards story-like visual explanations by watching movies and reading books." [Online]. Available: <http://www.cs.utoronto.ca/> (Accessed: Jan. 11, 2024)
- [19] L. Khalili, Y. You, and J. Bohannon, "BabyBear: cheap inference triage for expensive language models," May 2022, [Online]. Available: <http://arxiv.org/abs/2205.11747>





- [20] R. Karthiga, G. Usha, N. Raju, and K. Narasimhan, "Transfer learning based breast cancer classification using one-hot encoding technique," in *2021 International Conference on Artificial Intelligence and Smart Systems (ICAIS)*, IEEE, Mar. 2021, pp. 115–120. doi: 10.1109/ICAIS50930.2021.9395930.
- [21] D. Singh and B. Singh, "Investigating the impact of data normalization on classification performance," *Appl Soft Comput*, vol. 97, p. 105524, Dec. 2020, doi: 10.1016/j.asoc.2019.105524.
- [22] Q. Wang, Y. Ma, K. Zhao, and Y. Tian, "A Comprehensive survey of loss functions in machine learning," *Annals of Data Science*, vol. 9, no. 2, pp. 187–212, Apr. 2022, doi: 10.1007/s40745-020-00253-5.
- [23] J. Kakarla, B. V. Isunuri, K. S. Doppalapudi, and K. S. R. Bylapudi, "Three-class classification of brain magnetic resonance images using average-pooling convolutional neural network," *Int J Imaging Syst Technol*, vol. 31, no. 3, pp. 1731–1740, Sep. 2021, doi: 10.1002/ima.22554.
- [24] S. Mehta, C. Paunwala, and B. Vaidya, "CNN based traffic sign classification using adam optimizer," in *2019 International Conference on Intelligent Computing and Control Systems (ICCS)*, IEEE, May 2019, pp. 1293–1298. doi: 10.1109/ICCS45141.2019.9065537.
- [25] A. Luque, A. Carrasco, A. Martin, and A. de las Heras, "The impact of class imbalance in classification performance metrics based on the binary confusion matrix," *Pattern Recognit*, vol. 91, pp. 216–231, Jul. 2019, doi: 10.1016/j.patcog.2019.02.023.
- [26] G. S. Handelman *et al.*, "Peering into the black box of artificial intelligence: evaluation metrics of machine learning methods," *American Journal of Roentgenology*, vol. 212, no. 1, pp. 38–43, Jan. 01, 2019, doi: 10.2214/AJR.18.20224.
- [27] M. S. M. Sajjadi, O. Bachem, M. Lucic, O. Bousquet, and S. Gelly, "Assessing generative models via precision and recall," May 2018, [Online]. Available: <http://arxiv.org/abs/1806.00035> (Accessed: Feb. 12, 2024)
- [28] N. W. S. Wardhani, M. Y. Rochayani, A. Iriany, A. D. Sulistyono, and P. Lestantyo, "Cross-validation metrics for evaluating classification performance on imbalanced data," in *2019 International Conference on Computer, Control, Informatics and its Applications (IC3INA)*, IEEE, Oct. 2019, pp. 14–18. doi: 10.1109/IC3INA48034.2019.8949568.
- [29] M. Hossin and S. M.N., "A Review on evaluation metrics for data classification evaluations," *International Journal of Data Mining & Knowledge Management Process*, vol. 5, no. 2, pp. 01–11, Mar. 2015, doi: 10.5121/ijdkp.2015.5201.

## BIOGRAPHIES OF AUTHORS



**Jeffrey Junior Tedjasulaksana**     enrolled as a student majoring in computer science at Bina Nusantara University. Jeffrey is building upon the foundational knowledge acquired during his undergraduate studies, where he earned a bachelor's degree in computer science from Brawijaya University. His professional journey includes pivotal roles such as a data analyst at PT Adhi Commuter Property from 2022 to 2023. Furthermore, Jeffrey embarked on internships to deepen his practical understanding, serving as an IT Sales Operation intern at Eurokars Mazda from 2021 to 2022 and gaining hands-on experience as a software engineer intern at the Faculty of Computer Science at Brawijaya University for a period of three months. For further communication. He can be contacted at email: [jeffrey.tedjasulaksana@binus.ac.id](mailto:jeffrey.tedjasulaksana@binus.ac.id).



**Abba Suganda Girsang**     is presently employed as a lecturer in the Master of Information Technology program at Bina Nusantara University in Jakarta. He completed his Ph.D. at the Institute of Computer and Communication Engineering within the Department of Electrical Engineering at National Cheng Kung University in Tainan, Taiwan, in 2014. His undergraduate studies were undertaken at the Department of Electrical Engineering at Gadjah Mada University (UGM) in Yogyakarta, Indonesia, graduating in 2000. Following this, he pursued his master's degree in the Department of Computer Science at the same institution from 2006 to 2008. Over the course of his career, he has amassed experience as a staff consultant programmer at Bethesda Hospital in Yogyakarta in 2001, as well as working as a web developer from 2002 to 2003. Subsequently, he transitioned to the Faculty of the Department of Informatics Engineering at Janabadra University, where he served as a lecturer from 2003 to 2015. Additionally, he has taught various subjects at different universities from 2006 to 2008. Abba Suganda Girsang's research interests include swarm intelligence, combinatorial optimization, and decision support systems. He can be contacted at email: [agirsang@binus.edu](mailto:agirsang@binus.edu).