Sundanese Twitter Dataset for Emotion Classification

Oddy Virgantara Putra
Department of Informatics
Universitas Darussalam Gontor
Ponorogo, Indonesia
oddy@unida.gontor.ac.id

Fathin Muhammad Wasmanson
Department of Informatics
Universitas Darussalam Gontor
Ponorogo, Indonesia
fathin@mhs.unida.gontor.ac.id

Triana Harmini
Department of Informatics
Universitas Darussalam Gontor
Ponorogo, Indonesia
triana@unida.gontor.ac.id

Shoffin Nahwa Utama
Department of Informatics
Universitas Darussalam Gontor
Ponorogo, Indonesia
shoffin@unida.gontor.ac.id

***Abstract*—Sundanese is the second-largest tribe in Indonesia which possesses many dialects. This condition has gained attention for many researchers to analyze emotion especially on social media. However, with barely available Sundanese dataset, this condition makes understanding sundanese emotion is a challenging task. In this research, we proposed a dataset for emotion classification of Sundanese text. The preprocessing includes case folding, stopwords removal, stemming, tokenizing, and text representation. Prior to classification, for the feature generation, we utilize term frequency-inverse document frequency (TFIDF). We evaluated our dataset using k-Fold Cross Validation. Our experiments with the proposed method exhibit an effective result for machine learning classification. Furthermore, as far as we know, this is the first Sundanese emotion dataset available for public.**

***Keywords*—emotion classification, dataset, sundanese, support vector machine, text mining**

I. Introduction

Nowadays, social media has been widely known as a new way of communication. People have been using it for many purposes, such as promoting products, introducing health protocols, and researches. One of the generally used social media is Twitter. In Indonesia, Twitter has been used in many different local languages. The second-largest local language is Sundanese. Sundanese is quite active as their favorite football club, PERSIB Bandung has more than 3 million followers on Twitter. Twitter is prevalent in many types of research, especially in emotion analysis [1]–[5].

When people travel to a country that varies in ethnicity, such as Indonesia, they must pay attention to local customs prior to communicating with each other. It is easy to understand someone's expression from their face, even a subtle movement [6]. On the other hand, interpreting expression through text without emojis is burdensome. [4].

In a world of machine learning, emotion extraction is a challenging task. Many algorithms have been proposed in this field, from video-based [7] to text-based recognition [5], [8], [9]. Some of them are used to gather knowledge from customer satisfaction and business trends [2] from Twitter. Therefore, utilizing tweets is promising for data analysis.

Recently, there are many emerging issues on Twitter, especially West Java, from which Sundanese originate. It has considerable potential for sentiment and emotion study. On this day, there are many Indonesian Twitter datasets available for the public [2], [10]. However, it is yet to find public Sundanese Twitter dataset. Hence, in this work, we build a public Sundanese Twitter dataset for emotion classification.

In here, we provide a public dataset from Twitter and propose a model for emotion classification. Furthermore, we perform the classification with K-Nearest Neighbor (KNN), Random Forest (RF), Naive Bayes (NB), Logistic Regression (LR), and Support Vector Machine (SVM). We also evaluate our model using F1-Score, Precision, and Recall from the confusion matrix.

II. Related Works

Research in emotion analysis has been conducted for more than a decade, which results in many useful datasets available for the public. A heterogeneous annotated database has been publicly available by the contribution of [11]. This dataset was a combination of headlines, fairy tales, and blogs. Here, several fundamental emotions, such as fear, disgust, anger, sadness, happiness, and surprise, employed in order to analyze. The final task shows that this dataset has an incredible performance by using SVM compared to other classifiers.

A two-stage method proposed by [10] for Indonesian emotion detection from the Twitter dataset. The proposed method a couple of stages: emotion extraction and emotion classification. Here, emotions are grouped into five outstanding classes: joy, anger, sadness, fear, and love. Some various components were devised, such as semantic, linguistic, and orthographic, to classify the emotion. This work demonstrated superior results and tackled challenging issues in emotion analysis

A handy work by [2] proposed a public Indonesian emotion dataset. This incredible work gathered data from Twitter for about two weeks. This dataset contains five distinctive emotions: anger, sadness, fear, joy, and love. In the learning process, this dataset was classified using Logistic Regression (LR), RF, and SVM. 10-fold Cross-Validation was used to split the data between test and training. Finally, the results gained for precision, recall, F1-score are 70%, 68%, and 68%, respectively.

In the next few months, a practical text classifier using a pre-trained model proposed [1]. This work can be considered groundbreaking in natural language processing (NLP). The dataset was collected from Amazon Reviews. In order to preprocess, the dataset was separated into several batch groups. Each batch consists of tokenized vocabularies 32,000 in total.

As interest in microblog services gradually increasing, a number of topics are produced over time. The microblog attracts much attention in the analysis of emotional expression. Ren [5] proposed emotion extraction from Chinese microblogs. This work is rule-based, which contains three tasks. They are opinion findings, emotion analysis, and opinion target extraction.

III. Proposed Work

In this section, the proposed work is separated into several steps, such as Dataset Gathering and Annotation, Text Preprocessing, Feature Selection, Text Representation, Emotion Classification, and Model Evaluation.

A. Dataset

- Gathering and Annotation
  We gathered dataset from Twitter API between January and March 2019 with 2518 tweets in total. The tweets filtered by using some hashtags which are represented Sundanese emotion, for instance, #persib, #corona, #saredih, #nyakakak, #garoblog, #sangsara, #gumujeng, #bungah, #sararieun, #ceurik, and #hariwang. This dataset contains four distinctive emotions: anger, joy, fear, and sadness. Each tweet is annotated using related emotion. For data validation, we consulted a Sundanese language teacher for expert validation.
- Data Identity
  Our dataset consists of four distinctive emotions that have a balanced amount of data for each class. This can be seen in Fig 1. Our dataset can be accessed at here[1].

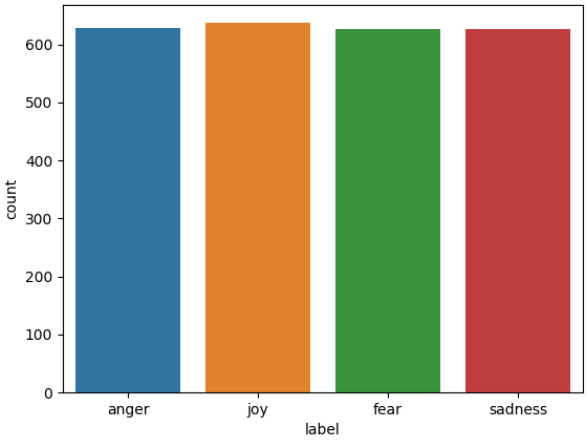

Fig. 1. Dataset identity.

B. Text Preprocessing

This step consists of four phase: case folding, filtering, tokenizing, and stemming.

- Case Folding
  It is broadly known that case-folding is often used in data preprocessing. This task is simple. All letters are reduced into the lowered-case form. In some cases, a lower case may be useful for data normalization. On the contrary, such a lowered word might be translated into another word. Therefore, it is better to let alone the letter case.
  In this step, the data separated into two groups: label and column. Tweets may contain inconsistent case. Therefore, we were uniformly using lowercase in every single word. For instance, this tweet,"Gokar! Punteun, buat yg masih pd nongkrong yg masih jalan2 gajelas" transformed into "gokar punteun buat yg masih pd nongkrong yg masih jalan gajelas"
- Stopword Filtering
  Stopword is defined as meaningless word. Such word does not affect much in a sentence. It is safe to ignore this kind of words. There are many words like this in English, for instance: is, are, were, that, this, which, etc. Subsequently, removing this word is an important task.
  Stopword filtering removes unnecessary characters from words because they are not representing any emotion. Here, we gathered many stopwords not only from Sundanese but also from Bahasa Indonesia. Twitter is vast and full of composite languages. We cannot expect a tweet contains a language from a specific country or tribe. Accordingly, our stopword is a mixture between Indonesia and Sundanese language. Here is some examples of out stopword: "tapi", "sanajan", "salain", "ti", "ku", "kituna", "sabalikna", "malah", "adalah", "nyah", "euy".
- Tokenizing
  Tokenizing is a process of splitting a sentence into several words. Each tweet is split into a word vector. This process may uses a different separator. However, in our case, we utilize space delimiter. For example:"gokar punteun buat yg masih pd nongkrong" transformed into 'gokar', 'punteun', 'buat', 'yg', 'masih', 'pd', 'nongkrong'.
- Stemming
  Stemming is a process of extracting or reducing words into its root form. It is similar to normalization but for text-based data. Stemming is useful for reducing the

[1]https://github.com/virgantara/sundanese-twitter-dataset.git

number of words in the corpus. Since both Sundanese and Bahasa have similar words, we adopted stemming from Bahasa.

C. Feature Selection

In here, feature selection utilizes stopword removal. This process removed all conjunctions. A list of stopwords is created out of Bahasa and Sundanese language. This list contains 508 words. We combined Bahasa and Sundanese because of many tweets containing these two languages. So, it would be ineffective for the results if we abandon or only use one of them.

D. Text Representation

Text Representation (TR) is considered as one of the main factors in text mining. Some feature extractions that employ Bag-of-Words (BoW) may reduce semantic information [12], not to mention sparsity. Here, in order to ease the classification, every tweet is transformed into a vector. Then, we work on some basic features such as BoW, TFIDF, and N-Grams.

- BoW (Bag-of-Words)
  BoW is a way to extract features from the text. It is also often described as the presence of words. In BoW, the more frequent word comes out, the more likely it becomes the feature.
- TF-IDF
  Different from BoW, TFIDF measures how important a word is within a document. This feature is composed of two parts. The first one calculates the term frequency (TF). TF represents the frequency of words from a document compared with the total number of words in the same document. The second one is Inverse Document Frequency (IDF), which computes the number as the logarithmic function of a document then divides the document count in which a related term emerges. Simply saying, TF-IDF is feature extraction, which the more word appears in a specific document, the more likely it works as the main feature.
- N-Gram
  In a nutshell, N-Gram is considered as sequential words. It also has a similar meaning to a phrase. A phrase may consist of more than one word. Should a phrase split into many words, it may obscure the meaning of the phrase. For example: a sentence such as "I do not like fried rice." if tokenized into some standalone words, it could have the opposite meaning. The word "not" is likely removed by the stopword. Thus, by applying N-Gram, we can handle such a sentence to maintain its meaning.

E. Emotion Classification

In this section, the dataset is processed using some machine learning algorithms. Each algorithm has a similar output which produce models. These models are later used for classifications. Prior to model evaluation, we incorporated K-Fold Cross Validation with K equals to 10. Several classifiers exhibited in this research are KNN, RF, NB, LR, and SVM.

TABLE I
EMOTION CLASSIFICATION BASED ON TFIDF FEATURE

Model	Emotion	Prec.	Rec.	F1	Acc.
KNN	Anger	85 %	97 %	90 %	84 %
	Fear	81 %	84 %	82 %	
	Joy	83 %	81 %	82 %	
	Sadness	87 %	73 %	79 %	
RF	Anger	93 %	95 %	94 %	92 %
	Fear	87 %	94 %	90 %	
	Joy	98 %	87 %	92 %	
	Sadness	92 %	92 %	92 %	
NB	Anger	73 %	79 %	76 %	65 %
	Fear	65 %	64 %	65 %	
	Joy	65 %	57 %	61 %	
	Sadness	57 %	59 %	58 %	
LR	Anger	94 %	95 %	95 %	94 %
	Fear	93 %	97 %	95 %	
	Joy	98 %	89 %	93 %	
	Sadness	90 %	94 %	92 %	
SVM	Anger	94 %	98 %	96 %	**95** %
	Fear	97 %	98 %	98 %	
	Joy	99 %	90 %	95 %	
	Sadness	92 %	95 %	94 %	

TABLE II
EMOTION CLASSIFICATION BASED ON BOW FEATURE

Model	Emotion	Prec.	Rec.	F1	Acc.
KNN	Anger	67 %	87 %	76 %	69 %
	Fear	63 %	81 %	71 %	
	Joy	73 %	61 %	67 %	
	Sadness	80 %	44 %	57 %	
RF	Anger	91 %	97 %	94 %	91 %
	Fear	87 %	94 %	90 %	
	Joy	95 %	85 %	89 %	
	Sadness	93 %	89 %	91 %	
NB	Anger	80 %	85 %	83 %	75 %
	Fear	68 %	87 %	77 %	
	Joy	77 %	69 %	72 %	
	Sadness	77 %	58 %	66 %	
LR	Anger	91 %	96 %	94 %	**93** %
	Fear	90 %	96 %	93 %	
	Joy	98 %	89 %	93 %	
	Sadness	93 %	90 %	91 %	
SVM	Anger	90 %	94 %	92 %	**93** %
	Fear	92 %	96 %	94 %	
	Joy	97 %	91 %	94 %	
	Sadness	93 %	90 %	91 %	

F. Model Evaluation

Here, we evaluate our models from each algorithm by calculating their precision, recall, f1-score, not to mention accuracy.

IV. EXPERIMENT AND RESULTS

In this part, all models produced by the aforementioned algorithms were tested. They were tested using a laptop with RAM 16 GB, Processor Intel I7-8750H, VGA NVIDIA GeForce GTX 1050 Ti, and operating system Ubuntu 18.04 LTE.

Table I illustrated the results of emotion classification from generally used algorithms KNN, RF, NB, LR, and SVM. As observed, the majority of algorithms achieved high accuracy,

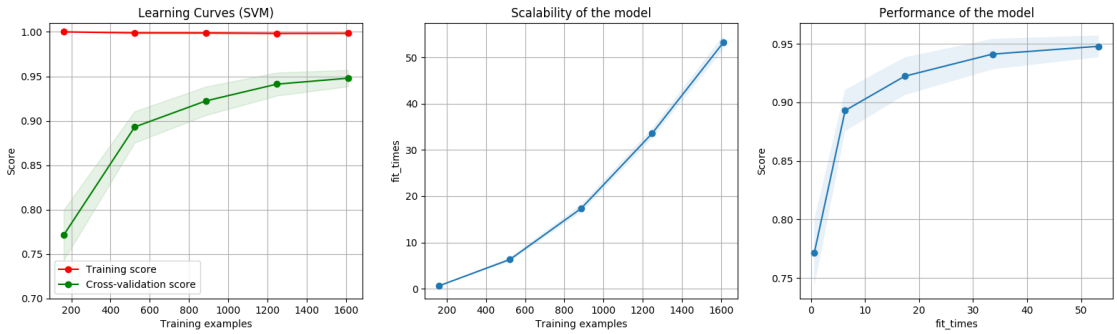

Fig. 2. Learning curve, scalability, and performance of the model.

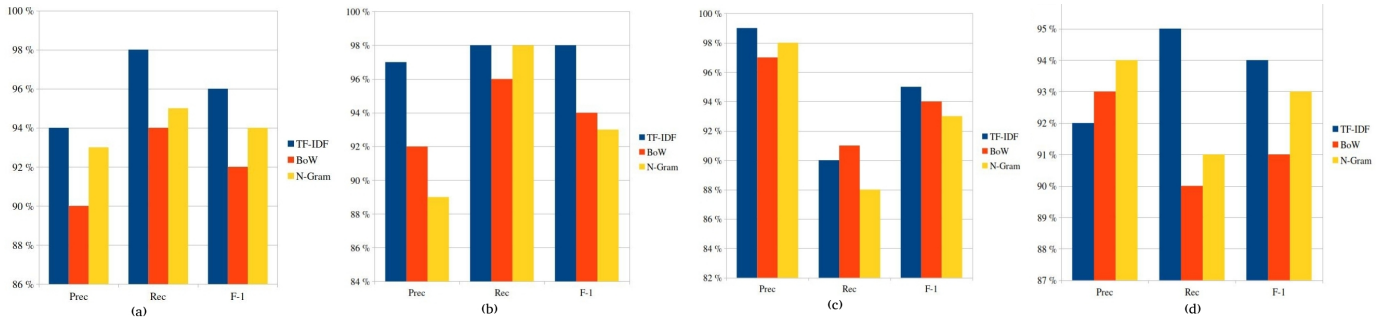

Fig. 3. Comparison of four emotions with three measurement (Precision, Recall, and F1-Score). (a) Anger, (b) Fear, (c) Joy, and (d) Sadness

TABLE III
EMOTION CLASSIFICATION BASED ON N-GRAM FEATURE

Model	Emotion	Prec.	Rec.	F1	Acc.
KNN	Anger	55 %	72 %	62 %	59 %
	Fear	53 %	71 %	60 %	
	Joy	71 %	48 %	58 %	
	Sadness	70 %	46 %	55 %	
RF	Anger	90 %	**98 %**	94 %	91 %
	Fear	83 %	**98 %**	90 %	
	Joy	97 %	82 %	89 %	
	Sadness	96 %	83 %	89 %	
NB	Anger	91 %	96 %	94 %	86 %
	Fear	85 %	89 %	87 %	
	Joy	85 %	84 %	84 %	
	Sadness	84 %	77 %	80 %	
LR	Anger	93 %	96 %	**95 %**	92 %
	Fear	89 %	**98 %**	93 %	
	Joy	**98 %**	86 %	92 %	
	Sadness	91 %	90 %	91 %	
SVM	Anger	93 %	95 %	94 %	**93 %**
	Fear	89 %	**98 %**	93 %	
	Joy	**98 %**	88 %	93 %	
	Sadness	94 %	91 %	93 %	

precision, recall, and F1-Score. Meanwhile, only one algorithm at a low value of the measurement. First of all, SVM stood overall remaining algorithms with roughly 96%. This was comprised of four emotions i.e., Anger, Sadness, Joy, and Fear. It can be seen that LR has a slightly lower accuracy than SVM with 94% and followed by RF at 92%. On the other hand, NB has the worst performance, with 65%. By the term of precision, SVM still performed its best in nearly all emotions with more than 95% on average. Surprisingly, RF overpowered all algorithms at Sadness emotion with 95%. Then, SVM achieved at top-notch in Recall and F1-Score, with the same value at 95.5%.

As we found that SVM gained top condition, we measured its performance. Fig. 2 shows the trend of three different types of evaluators. Overall, it can be seen that the trend of the learning curve for training examples experienced remarkable change throughout time.

To begin with, it is clear that the learning curve climbs dramatically in the first stage of training data from 200 to 500. At the second stage, its performance increasingly steady but at low speed.

Second of all, model performance has slightly the same performance as the learning curve. In the beginning, it gradually climbs up overfitting times and achieves a score of 0.95.

Interestingly, the scalability model just started its speed climbing exponentially to peak with more than 50 fitting times.

We found different results, as illustrated in Table II. Both SVM and LR make to the top rank with 93%, followed by RF, NB, and KNN with 91%, 75%, and 69%, respectively. By using BoW feature extraction, NB has switched position with KNN, which is no longer at the bottom tier of classifiers. In Table III, we evaluated our model using unigram and bigram features with three different parameters. It can be seen that SVM still dominates other algorithms with 93% followed by LR, RF, LR, and KNN. On the contrary, KNN performance

decreased drastically for accuracy at 59%. This is the lowest score compared in Table I and Table II.

The graph in Fig. 3 compares the performance of all text representation features, which are calculated in precision, recall, and F1-score. Overall, TF-IDF was significantly higher in all emotions in which contributes to precision.

To begin with, the precision of TF-IDF for Anger class stood at 94 percent, while for the counterparts are greater than 95 percent. Surprisingly, the recall of TF-IDF peaked at 98 percent, followed by N-Gram and BoW, not to mention F1-Score in which gained the top position at 96 percent.

In Fear class, the proportion of precision, recall, and F1-score for TF-IDF was around 97 percent. As in recall, TF-IDF stood at an equal position with N-Gram. However, the disparity of F1-Score between TF-IDF and the remainings was dramatically high. Now, we moved to Joy class. On average, all three features were almost at the same level. The precision was relatively high, at 98 percent.

In terms of Sadness class, the precision of TF-IDF was defeated by N-Gram, followed by BoW. On the contrary, both BoW and N-Gram were outclassed entirely by TF-IDF by 5 percent.

V. Conclusion

In this research, we have built a new public dataset for Sundanese emotion classification. Our dataset contains four distinguished annotated classes (fear, joy, anger, and sadness). We tested our dataset with five algorithms. As a result, the SVM model gained the highest score, with 95% accuracy followed by other algorithms. We found that different feature extraction exploits different results.

We need to employ stemming specifically for the Sundanese language and gather more massive datasets in future works.

References

[1] N. Kant, R. Puri, N. Yakovenko, and B. Catanzaro, "Practical Text Classification With Large Pre-Trained Language Models," *arXiv:1812.01207 [cs]*, Dec. 2018. Comment: 8 pages, submitted to AAAI 2019.

[2] M. S. Saputri, R. Mahendra, and M. Adriani, "Emotion Classification on Indonesian Twitter Dataset," in *2018 International Conference on Asian Language Processing (IALP)*, (Bandung, Indonesia), pp. 90–95, IEEE, Nov. 2018.

[3] M. O. Ibrohim and I. Budi, "Multi-label Hate Speech and Abusive Language Detection in Indonesian Twitter," in *Proceedings of the Third Workshop on Abusive Language Online*, (Florence, Italy), pp. 46–57, Association for Computational Linguistics, 2019.

[4] S. Sendari, I. A. E. Zaeni, D. C. Lestari, and H. P. Hariyadi, "Opinion Analysis for Emotional Classification on Emoji Tweets using the Naïve Bayes Algorithm," *Knowledge Engineering and Data Science*, vol. 3, pp. 50–59, Aug. 2020.

[5] F. Ren and Q. Zhang, "An Emotion Expression Extraction Method for Chinese Microblog Sentences," *IEEE Access*, vol. 8, pp. 69244–69255, 2020.

[6] N. Muna, U. D. Rosiani, E. M. Yuniamo, and M. H. Pumomo, "Subpixel subtle motion estimation of micro-expressions multiclass classification," in *2017 IEEE 2nd International Conference on Signal and Image Processing (ICSIP)*, (Singapore), pp. 325–330, IEEE, Aug. 2017.

[7] C. Li, J. Wang, H. Wang, M. Zhao, W. Li, and X. Deng, "Visual-Texual Emotion Analysis With Deep Coupled Video and Danmu Neural Networks," *IEEE Transactions on Multimedia*, vol. 22, pp. 1634–1646, June 2020.

[8] H. Fei, D. Ji, Y. Zhang, and Y. Ren, "Topic-Enhanced Capsule Network for Multi-Label Emotion Classification," *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, vol. 28, pp. 1839–1848, 2020.

[9] S. Ahmad, M. Z. Asghar, F. M. Alotaibi, and S. Khan, "Classification of Poetry Text Into the Emotional States Using Deep Learning Technique," *IEEE Access*, vol. 8, pp. 73865–73878, 2020.

[10] J. E. The, A. F. Wicaksono, and M. Adriani, "A two-stage emotion detection on Indonesian tweets," in *2015 International Conference on Advanced Computer Science and Information Systems (ICACSIS)*, (Depok, Indonesia), pp. 143–146, IEEE, Oct. 2015.

[11] S. Chaffar and D. Inkpen, "Using a heterogeneous dataset for emotion analysis in text," pp. 62–67, May 2011.

[12] Zhou, Wang, Sun, and Sun, "A Method of Short Text Representation Based on the Feature Probability Embedded Vector," *Sensors*, vol. 19, p. 3728, Aug. 2019.