Contents lists available at Science-Gate

International Journal of Advanced and Applied Sciences

Journal homepage: http://www.science-gate.com/IJAAS.html

Multi-label text classification on unbalanced Twitter with monolingual model and hyperparameter optimization for hate speech and abusive language detection

Ahmad A. Alzahrani [1], Arif Bramantoro [2,*], Asep Permana [3]

[1]*Faculty of Computing and Information Technology, King Abdulaziz University, Jeddah, Saudi Arabia*
[2]*School of Computing and Informatics, Universiti Teknologi Brunei, Bandar Seri Begawan, Brunei*
[3]*Faculty of Information Technology, Universitas Budi Luhur, Jakarta, Indonesia*

ARTICLE INFO

Article history:
Received 19 December 2023
Received in revised form
2 May 2024
Accepted 5 May 2024

Keywords:
Hate speech
Abusive language
Imbalanced dataset
Multi-label text classification
Hyperparameter optimization

ABSTRACT

The increase in hate speech and abusive language on social media leads to uncomfortable interactions among users. Many datasets available publicly that address hate speech and abusive language are not balanced, particularly those from Indonesian Twitter. To develop a more effective classification model that also considers minority classes, we needed to optimize the hyperparameters of a monolingual model, use four different data preprocessing scenarios, and improve the treatment of slang words. We assessed the model's effectiveness by its accuracy, achieving 81.38%. This result came from optimizing hyperparameters, processing data without stemming and removing stop words, and enhancing the slang word data. The optimal hyperparameters were a learning rate of 4e-5, a batch size of 16, and a dropout rate of 0.1. However, using too much dropout can decrease the model's performance and its ability to predict less common categories, such as physical- and gender-related hate speech.

1. Introduction

Hate speech and abusive language have spread more quickly due to the Internet and social media. Most of this speech is considered criminal. Detecting it requires automatic classification. Multi-label text classification is a challenging area because of the complexity of the data (Ibrohim and Budi, 2019; Kovács et al., 2021). This complexity makes it hard to identify information about labels that are not mutually exclusive, such as hate speech and abusive language in tweets.

Hate speech is considered any form of communication that demeans a person or group based on their attributes, such as race, religion, gender, and sexual orientation (Warner and Hirschberg, 2012). Abusive language includes several aspects, such as cyberbullying, verbal abuse, and offensive curses intended to humiliate others (Niemann et al., 2020). One of the main issues related to hate speech research is data imbalance (Ibrohim and Budi, 2019; Prabowo et al., 2019; Hana et al., 2020; Hendrawan et al., 2020). This problem has been under investigation for many years because it provides poor classification and unpredicted labels. A classifier tends to be burdened by the majority labels and ignores the minority labels (Ramyachitra and Manikandan, 2014; Winata and Khodra, 2015). In other words, the classifier is unable to provide an expected accuracy for all labels. The best classification method is required for imbalanced data because of the data explosion that provides imbalanced data in daily life.

To date, there are three approaches to dealing with the problem of imbalanced data: Data-level, algorithm-level, and hybrid (Johnson and Khoshgoftaar, 2019). The data-level approach focuses on the balance of the label distribution by sampling the minority label (i.e., oversampling) or reducing the samples of the majority label (i.e., undersampling). Oversampling can increase the likelihood of overfitting because it creates multiple copies of the minority label samples (Fernández et al., 2018). The problem of overfitting and underfitting during the training session decreases the ability of the model to generalize the sampling (Li et al., 2019). Undersampling is basically data reduction by eliminating several samples with the majority label. The aim of undersampling is to

* Corresponding Author.
Email Address: arif.bramantoro@utb.edu.bn (A. Bramantoro)
https://doi.org/10.21833/ijaas.2024.05.019
Corresponding author's ORCID profile:
https://orcid.org/0000-0003-2772-9427
2313-626X/© 2024 The Authors. Published by IASE.

standardize the number of samples for each label. The main drawback of this method is that it can discard potentially useful data, which can be important for induction. The algorithm-level approach (Fernández et al., 2018) aims to modify the classifier learning procedure. This approach does not cause any modifications in the distribution of the data. Hence, it is more adaptable to any type of imbalanced dataset.

This study proposed to use an algorithm-level approach by evaluating and analyzing the ability of the IndoBERT monolingual model (Wilie et al., 2020). The proposal included the optimization of hyperparameters in handling imbalanced Twitter data in multi-label text required by classification tasks. Real datasets from Indonesian Twitter are used, and confidentiality is strictly maintained. Several preprocessing scenarios, such as full pre-processing, without stemming, without stop word removal, and without stemming and stop word removal, were proposed to determine the effect on the performance of the classification model. This study also investigated the errors in the list of slang words published in previous research (Ibrohim and Budi, 2019) that were used as word normalization in the pre-processing stage and analyzed its effect on the model's performance

There are two main contributions of this research. First, the proposed method was effective in dealing with imbalanced Indonesian Twitter data using an algorithm-level approach with hyperparameter optimization and monolingual models. Second, the slang word data were improved to reduce the ambiguity and improve the performance of the classification model.

2. Related works

The proliferation of labeled datasets has sparked considerable interest in classification across diverse domains, including telecommunications (Makruf et al., 2021), biology (Pratondo and Bramantoro, 2022), and law (Bramantoro and Virdyna, 2022). Several studies related to hate speech and abusive language have been proposed, and their datasets have been shared to support other researchers. One study (Ibrohim and Budi, 2019), which proposed using a random forest decision tree (RFDT) combined with a label powerset and word unigram, claimed to be the best method for classifying multi-label data. Although the proposed method achieved good results, an accuracy of 77.36%, it involved only hate speech labels and abusive language. There was no identification of the target, category, and level of hate speech. To identify a label that includes the target, category, and level of hate speech, the approach was unable to achieve good results, a 66.12% accuracy. It was stated that the high number of errors in terms of false negatives was caused by an imbalanced dataset that required a more advanced technique to handle.

Prabowo et al. (2019) combined the RFDT, naïve Bayes, and support vector machine (SVM) methods. The feature extraction method was word n-gram and character n-gram. They conducted five scenarios with different label hierarchies to achieve the highest possible accuracy by hierarchical classification. The experimental results show that the hierarchical approach with the SVM algorithm and the word unigram feature had an accuracy of 68.43%. This proves that the hierarchical approach can improve the performance of multi-label classification. The classification performance increased by up to 2.27% when using a hierarchical approach, although it was performed on imbalanced data. The power-set labels approach was found to encounter difficulties in classifying group categories because it tended to group categories into one label, and it has drawbacks in classifying tweets that have more than one group category.

Putra and Purwarianti (2020) focused more on binary text classification by analyzing the effect of combining English data and Indonesian-language data using the XLM-RoBERTa (Cross-Lingual Language Model RoBERTa) multilingual model. The findings show that adding English data can improve the classification performance of Indonesian data, with the best accuracy of 89.9%. Apart from the limitation in binary classification, other findings are also presented. There were cases where the addition of excessive English data could reduce the classification performance.

Hana et al. (2020) exploited the use of support vector machines (SVMs), convolutional neural networks (CNN), and DistilBERT classification methods. The label powerset and classifier chain methods were proposed as a combination method for data transformation. They provided an analysis of the relationship between hate speech labels, such as targets, categories, and levels of hate speech, and abusive words. It considered the effect of data preprocessing, such as stop words, stemming, and translation, on classification performance. It was shown that classification using SVM and a classifier chain without stemming, without removing stop words, and without translation gives the best accuracy of 74.88%. They also showed that most scenarios using the CNN model failed to predict the dataset at low labels.

Hendrawan et al. (2020) proposed RFDT, bidirectional long short-term memory (BiLSTM), bidirectional encoder representations from transformers (BERT), and BiLSTM+BERT as classification methods. The classifier chain, label powerset, and binary relevance methods were used for data transformation, and the TF-IDF method was used for feature extraction. They analyzed the effect of hate speech labels that included targets, categories, and levels of hate speech, as well as abusive words, on the classification performance. They also studied the effect of data preprocessing, such as stop word, stemming, and machine translation. The best result was achieved using the RFDT method combined with classifier chains and TF-IDF without translation, without stemming, and without removing stop words, with a 76.12% accuracy. It is interesting to note that the deep learning methods of BERT and BiLSTM were considered less effective due to the frequently occurring overfit. This overfitting was because the dataset normally has several labels that are imbalanced.

Finally, to the best of our knowledge, to conclude the literature review, imbalanced data, especially on Indonesian-language Twitter, are nontrivial when being handled for the improvement of multi-label text classification tasks. The use of a reliable classifier of a pre-trained IndoBERT model is challenging because of the need to optimize the hyperparameters. The detection of hate speech and abusive language can be used as a scenario for this task of classification.

3. Methodology

The IndoBERT monolingual model has previously been trained for various NLP tasks using Indonesian-language datasets developed by IndoNLU (Wilie et al., 2020). This model shows good achievement, especially for multi-label text classification tasks. Hence, we propose to use and improve the same IndoBERT model.

To obtain an effective classification model for dealing with imbalanced data, we propose to optimize the hyperparameters, including the adaptive moment estimation (Adam) (Kingma and Ba, 2015), learning rate, batch size, dropout, and epoch. Each hyperparameter was analyzed gradually to determine the best combination. Special attention was given to the epoch configuration, which was implemented at each of the fine-tuning steps.

Adam is basically a gradient-based optimization based on low-level adaptive moment estimation (Kingma and Ba, 2015). Adam uses the first and second gradient moment estimation to adjust the learning speed for each neural network weight. The nth moment of a random variable is defined as the expected value of that variable to the power of n, as described in the following equation:

$$m_n = E[X^n] \tag{1}$$

where, m is the moment, and X is the random variable. The gradient of the cost function in a neural network is considered a random variable because it is usually evaluated on several small random datasets. The first moment is the mean, and the second moment is the uncentered variance, i.e., the mean is not reduced during the calculation of the variance. To estimate the moments, Adam uses exponential moving averages, calculated on gradients with mini-batches, as described in the following equation:

$$a_t = \beta_1 a_{t-1} + (1 - \beta_1) g_t$$
$$b_t = \beta_2 b_{t-1} + (1 - \beta_2) g_t^2 \tag{2}$$

where, a and b are the moving average vectors, g is the gradient of the current mini-batch, and β is the new hyperparameter of the algorithm. The moving average vector is initialized to zero in the first iteration.

Dropout is a recently introduced algorithm for training neural networks by dropping units randomly during training (Hinton et al., 2012). Overfitting can be reduced by using dropout to prevent complex co-adaptations of training data. Fig. 1 illustrates a dropout of the neural network model, as explained in Srivastava et al. (2014). Fig. 1 shows a standard neural network with two hidden layers, and an example of a thinned network generated by applying a dropout to the network. In this phase, the crossed units have been dropped. The unit to be dropped is selected randomly. In the simplest case, each unit is maintained with a fixed probability that is independent of other units. The probability value is assigned using the validation set.

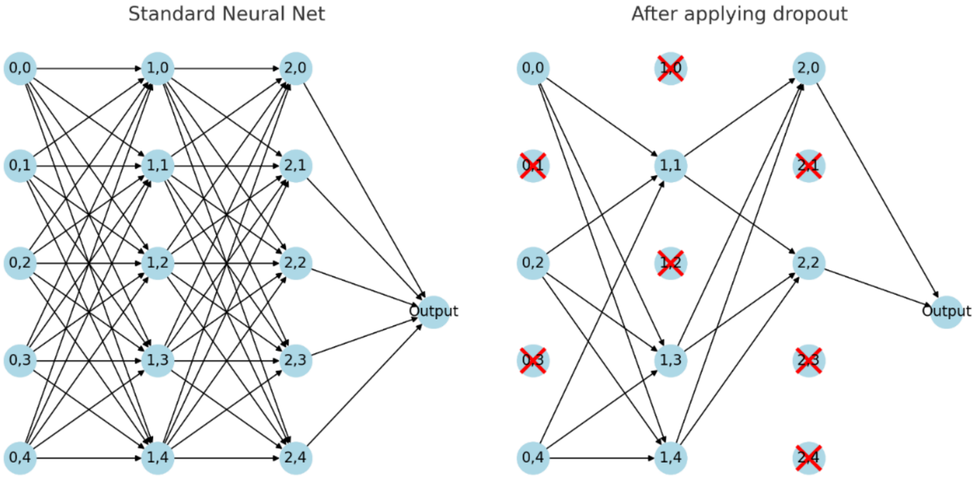

Fig. 1: Dropout in a neural network model (Srivastava et al., 2014)

Epoch is a training algorithm for neural networks that includes all training data in one cycle. The forward and backward processes are counted as a single pass. One epoch means that each sample in the training data has an opportunity to update the internal model, and one epoch consists of one or more batches as iterations. The hyperparameter optimization is divided into two searching phases:

Searching for the learning rate and batch size values and searching for the dropout values. The search for hyperparameter combinations can be seen in Fig. 2. Both phases produce performance reports of the same IndoBERT algorithm. In the first phase, we set the learning rate value to be tested: 1e-5, 2e-5, 3e-5, 4e-5, and 5e-5.

If the batch size refers to the baseline from IndoBERT, it is set to 8.16. If the batch size refers to the BERT baseline, it is set to 32. We recommend using 24 as the batch size. At this phase, the dropout value is 0.1 based on the default BERT value. The data preprocessing used in this phase refers to previous studies (Hana et al., 2020; Hendrawan et al., 2020) to provide the best accuracy. The fine-tuning result is evaluated for its performance for each learning rate and batch size value until it obtains the best combination of learning rate and batch size.

At the end of the first phase, the learning rate and batch size values are set for the second phase. The determination of the learning rate and batch size values is based on the model's accuracy in evaluating the results of the testing data. The second phase is required to determine the dropout value, and the dropout values evaluated in this study were 0.1, 0.2, 0.3, 0.4, and 0.5. Hence, this phase requires five attempts to evaluate its performance. The selection of the dropout value is based on the accuracy value of the classification model. After the two phases are executed, the best combination of hyperparameters is obtained based on the results of the performance evaluation in each phase of the experiment.

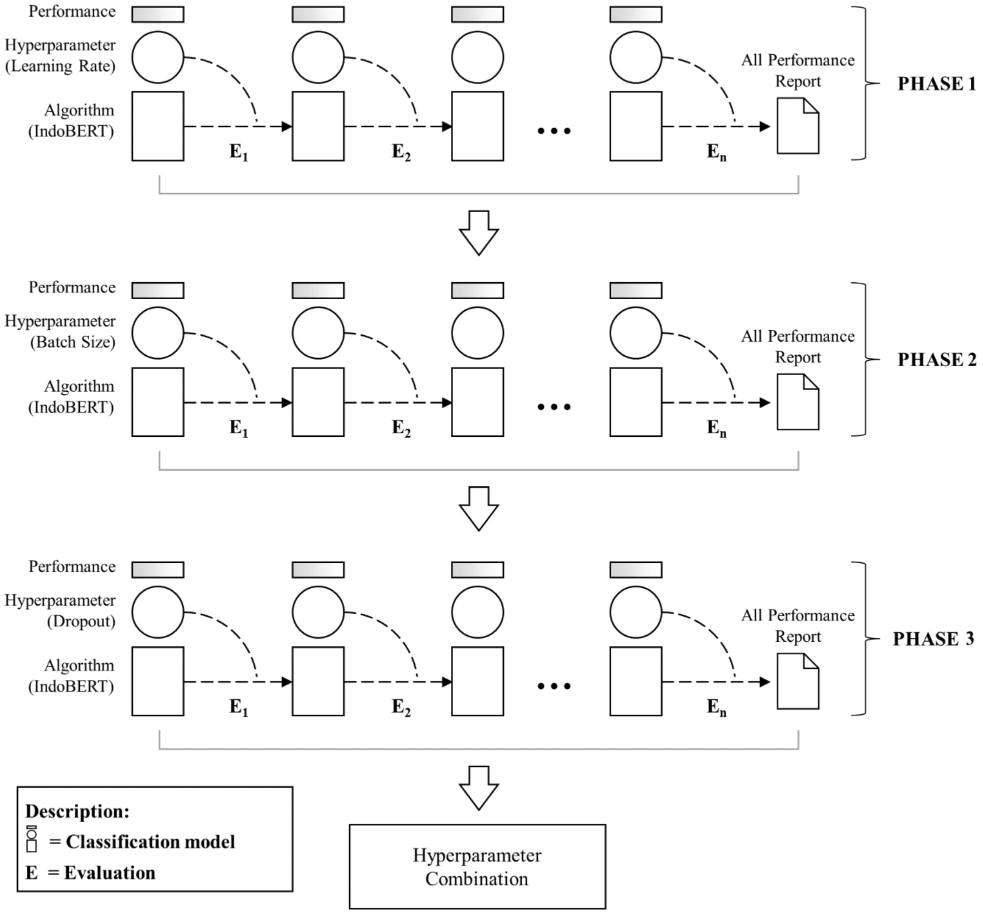

Fig. 2: Hyperparameter optimization

4. Analysis

The analysis is performed in several stages utilizing several analytical techniques, as shown in Fig. 3, which can generally be divided into two processes: Data classification and hyperparameter configuration. Data classification consists of data preprocessing, data preparation, data training, data validation, data testing, fine-tuning, IndoBERT classification, evaluation, and performance reports. Hyperparameter configuration consists of searching for the batch size, learning rate, number of epochs, and dropout. The result of the hyperparameter configuration is used in the data preparation, fine-tuning, and IndoBERT classification. The data preprocessing stage uses four scenarios: full preprocessing; without stemming; without stop word removal; and without both stemming and stop word removal. Data transformation is performed to meet the requirements of the model evaluation. To save time, the data transformation is done on the initial data before it is cleaned and divided into the four preprocessing scenarios. This way, the transformation does not need to be repeated for each scenario. Additionally, emoji byte codes are deleted before character deletion and dataset splitting.

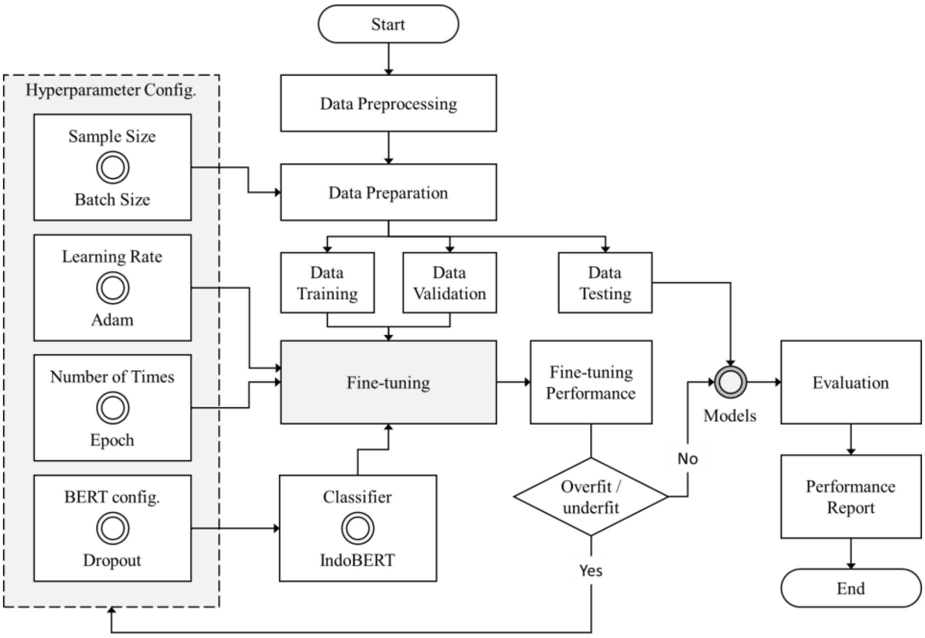

Fig. 3: Analysis process flow

There are two main processes in the early stages of data preprocessing: data transformation and emoji byte code removal. Data transformation is necessary for the experiment to meet the analysis requirements and ensure the proposed method works smoothly. Before data cleaning, three adjustments are made: adding a column labeled "Normal" to the dataset, changing the column header "Tweet" to "Sentence," and changing all column headers to lowercase.

This research is basically text classification; therefore, emoji byte code removal is required because emoji byte codes are considered meaningless for classification. An emoji byte code in the dataset is in a string format; therefore, it cannot be deleted using standard regular expressions. For example, an emoji byte code in the dataset was “\xf0\x9f\x98\x82”. We used the string replacement function to remove one or more repeated “\x” substring literals followed by two hex characters in Python code. The hex character is a string constant that can express the character code in hexadecimal with a “\x” prefix. The emoji byte code removal step is required before the character or symbol is deleted; otherwise, it may affect the result in that the emoji byte code is not deleted.

Case folding is the process of converting documents into lowercase letters. It is required because the documents, especially Twitter posts, are not always consistent in using capital letters. Twitter posts also tend to be inherently poor in quality. There are several symbols that have no meaning, as well as words that do not comply with standard vocabulary, spelling, and syntax. Therefore, unnecessary characters and words need to be removed to improve the quality of the tweets to be analyzed. In detail, it removes any symbols, punctuation marks, and numbers that have no meaning. Twitter also has several special symbols or terms, such as the ones to denote uniform resource locator and retweet. However, sentences or words contained in the hashtag are not removed because they are meaningful in the tweet.

Word normalization is the process of changing nonstandard words into standard words without changing the meaning of the word. This study uses a normalization dictionary from previous work (Ibrohim and Budi, 2019). An example of a nonstandard word that was changed to a standard word was “bapake (his father),” becoming “ayahnya (his father).” A nonstandard word also includes a typo. An example of a typo is “biza (able)” becoming “bisa (able).”

Stop word removal is the process of removing words that are considered to contain no information or have no meaning (Srividhya and Anitha, 2010). However, this data preprocessing step must be executed meticulously to prevent the inadvertent removal of contextually significant information. The stop-word removal process was carried out using the stop-word list published in Putra and Purwarianti (2020). Stemming is the process of returning affixed words to their basic form (Srividhya and Anitha, 2010). The stemming process used the PySastrawi stemmer library. An example of an affixed word that was converted into a basic word through the stemming process was “mempercayai (believing)” becoming “percaya (believe).”

The data exploration stage is useful for understanding the condition of the dataset to be analyzed. After several data preprocessing steps are carried out, there are several changes in the data that lead to the possibility of missing values. These missing values stop the classification. Therefore, data exploration is required to deal with this problem. Data exploration is carried out in four scenarios: Full preprocessing, without stemming, without stop word removal, without stemming, and without stop word removal.

Data preparation includes data separation of the dataset. This research follows IndoBERT (Wilie et al., 2020), which divides the data into three parts:

Training data, validating data, and testing data. We propose to separate the data into 90% training data, 5% validating data, and 5% testing data.

Before fine-tuning, the hyperparameters were configured by looking for the best combination of batch size, learning rate, epoch, and dropout values. To determine the combination of hyperparameters, this study required several experimental fine-tunings. The learning process involved the hyperparameters and classification models that were trained and validated to produce a classification model that could detect hate speech and abusive language.

The final stage of the analysis was to measure the ability of the designed model with an evaluation. The evaluation was carried out on the testing data utilizing the previously trained model. This paper followed the evaluation method used by El Kafrawy et al. (2015) that proposed the accuracy performance of the multi-label classification equation as follows:

$$Accuracy = \left(\frac{1}{D}\sum_{i=1}^{D}\left|\frac{\hat{L}^{(i)} \wedge L^{(i)}}{\hat{L}^{(i)} \vee L^{(i)}}\right| x100\%\right) \tag{3}$$

where, D is the number of data samples; $\hat{L}^{(i)}$ is the predicted labels for the i^{th} data sample; $L^{(i)}$ is the actual label for the ith data sample; $\wedge$ is the logical AND operator; and $\vee$ is the logical OR operator. It is important to note that the equation could only be used in this research if we assigned a new label of "normal" for the dataset that had all bit values of zeros. This is because of the nature of the equation, which requires nonzero values for all bits in the predicted and actual datasets; otherwise, the logical operator is unable to provide the correct result.

5. Result and discussion

This paper used the same data as published by previous research (Ibrohim and Budi, 2019) in order to have a baseline comparison. The data collection was carried out by crawling Twitter posts for approximately seven months, from March 2018 to September 2018. Additional data were also collected from several previous studies (El Kafrawy et al., 2015; Alfina et al., 2017; Saputri et al., 2018). There are 13,169 tweets and 12 labels in total. This paper used the same labels as Ibrohim and Budi (2019). The labeling was obtained from a focus group discussion with Indonesian police officers who are authorized to handle cybercrime. However, this labeling approach may not universally apply to all types of textual data or languages, potentially leading to the inadvertent removal of contextually significant information. The distribution of labels on the dataset can be seen in Fig. 4. The labels are not distributed equally. Several labels, such as HS_Gender, HS_Physical, HS_Strong, HS_Race, and HS_Religion, are significantly low in proportion compared to other labels.

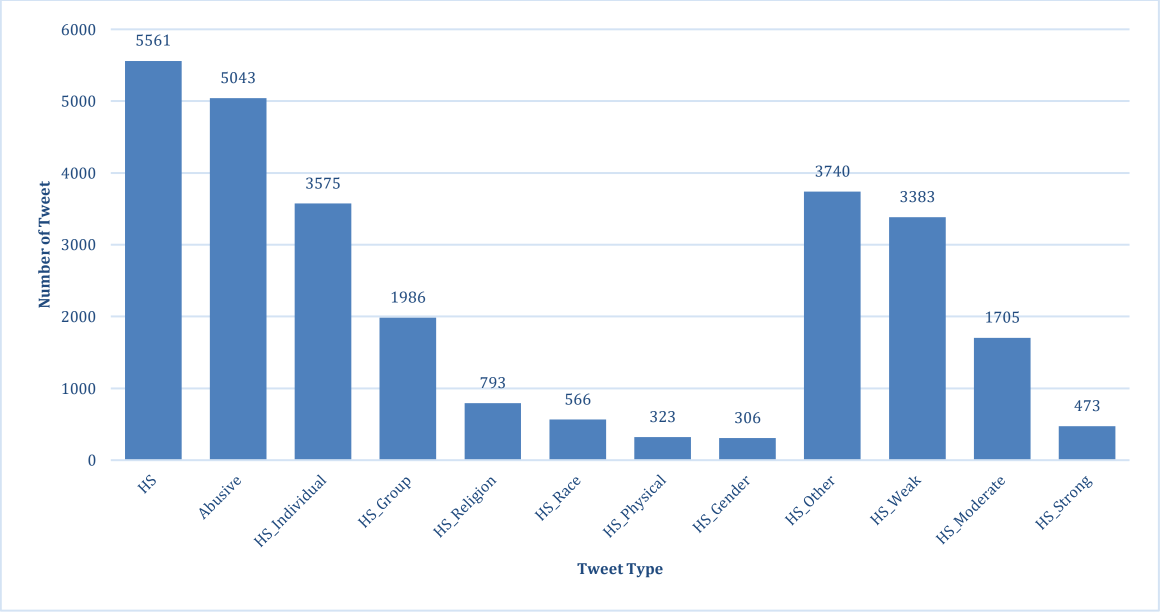

Fig. 4: Distribution of labels on the initial data

The imbalanced data based on the proportion of minority labels are divided into three levels: Mild for labels that have a proportion of 20-40% of the dataset, medium for labels that have a proportion of 1-20% of the dataset, and extreme for labels that have a proportion of less than 1% of the dataset. Hence, for the imbalanced data, a mild level was obtained for the HS_Abusive, HS_Other, HS_Individual, and HS_Weak labels; a medium level was obtained for the HS_Group, HS_Moderate, HS_Religion, HS_Race, HS_Strong, HS_Physical, and HS_Gender labels. Although HS_Physical and HS_Gender labels are not considered at an extreme level of data imbalance, they have the lowest distribution for the dataset, so these two labels are the most difficult to identify.

The model of the classification task was modified to obtain the required model by optimizing the hyperparameters. The first phase of the hyperparameter optimization was determining the learning rate and batch size values. As listed in Table 1, we ran 20 hyperparameter combination tests. The highest accuracy was achieved with a learning rate of 4e-5 and a batch size of 16. Once the first phase of the optimization was complete, the optimization of the dropout values continued.

The second phase was the optimization of the dropout values using the learning rate and batch size values obtained from the first phase. The results of the test scenarios in this phase are shown in Table 2. Five test scenarios were proposed to determine the best dropout value. With a dropout value of 0.1, the highest accuracy was achieved. An accuracy of 81.38% was obtained, and all labels were correctly predicted.

For comparison, a dropout value of 0.5 failed at predicting two labels, HS_Physical and HS_Gender, although the accuracy was not the lowest. Based on the results of this stage, we can conclude that the use of an excessive dropout can reduce the performance of the model. It can be seen in Fig. 5 that the accuracy value was quite low, with dropout values of 0.4 and 0.5. Although a dropout value of 0.5 provided a slightly higher result than 0.4, there is no guarantee that it would continue to rise. Hence, the best value for the dropout was 0.1 in the second phase of testing.

The results of the data preprocessing test scenarios to investigate the effect of stemming and stop word removal are shown in Table 3. The test results show that data preprocessing without stemming and without stopping word removal had the greatest impact on accuracy, at 81.38%. On the other hand, the use of stemming and stop word removal as full processing was the least accurate, at 75.89%. Hence, it can be inferred that data preprocessing without stemming and without removing stop-words was the best among the three other scenarios.

Table 1: The first hyperparameter optimization

Learning rate	Epoch	Batch size	Dropout	Accuracy (%)
1e-5	3	8	0.1	78.98
1e-5	3	16	0.1	78.36
1e-5	3	24	0.1	78.29
1e-5	4	32	0.1	80.30
2e-5	2	8	0.1	79.80
2e-5	2	16	0.1	80.23
2e-5	3	24	0.1	76.96
2e-5	3	32	0.1	77.38
3e-5	2	8	0.1	79.87
3e-5	3	16	0.1	78.24
3e-5	2	24	0.1	81.15
3e-5	3	32	0.1	76.47
4e-5	2	8	0.1	79.13
4e-5	4	16	0.1	81.38
4e-5	2	24	0.1	79.36
4e-5	3	32	0.1	74.56
5e-5	3	8	0.1	77.46
5e-5	3	16	0.1	76.85
5e-5	3	24	0.1	76.42
5e-5	3	32	0.1	75.74

Table 2: The second hyperparameter optimization

Dropout	Accuracy (%)	Failed label prediction
0.1	81.38	-
0.2	80.29	-
0.3	80.35	-
0.4	77.54	-
0.5	77.79	HS_Physical, HS_Gender

Table 3: The effect of stemming and stop word removal

Data preprocessing scenario	Accuracy (%)
Full preprocessing	75.89
Without stemming	77.70
Without stop word removal	80.49
Without stemming and without stop word removal	81.38

The improvement of slang word data can increase the performance compared to the baseline taken from previous research (Ibrohim and Budi, 2019). All experiments carried out on the four data preprocessing scenarios showed an increase in accuracy. The detailed test results are shown in Table 4. For the full data preprocessing, the proposed improvement of the slang word data outperformed the baseline by 0.84%. In the data preprocessing without stemming, the proposed improvement of the slang word data outperformed the baseline by 1.15%. For the data preprocessing without stop word removal, the proposed improvement of the slang word data outperformed the baseline by 2.43%. In the data preprocessing without stemming and without stop word removal, the proposed improvement of the slang word data outperformed the baseline by 1.47%.

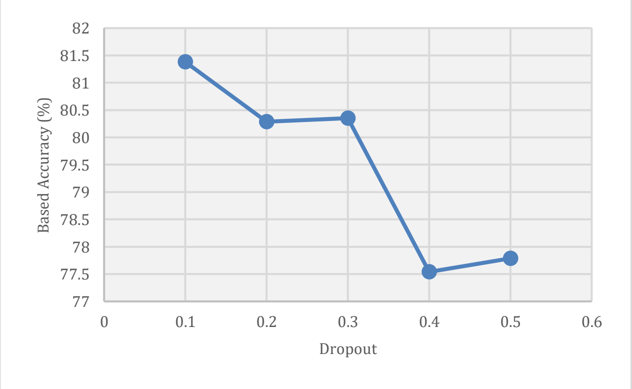

Fig. 5: Dropout effect on the model's accuracy

The proposed slang word data improvement was tested on four data preprocessing scenarios using the best combination of the hyperparameters.

Overall, a combination of the IndoBERT monolingual hyperparameter, the various data preprocessing scenarios, and the proposed improvements of slang word data are considered effective in building a multi-label text classification model on imbalanced Indonesian-language Twitter data. To provide a fair comparison with previous studies, this paper used the same dataset and labels. In addition to the baseline research, the highest accuracy of 81.38% in this study was also better than the accuracy of 68.43% in the previous research that used the hierarchical method (Prabowo et al., 2019). Moreover, it outperformed the previous research in Hendrawan et al. (2020), which had an accuracy of 76.12%. However, it is important to note that this improvement can be achieved by adding 5227 more data rows from Twitter because of the requirement to improve the slang word data. Moreover, we are aware that the limitation of the algorithm-level approach could impact the generalizability of the model to other datasets or languages.

Table 4: The effect of slang word data improvement

Technique	Data preprocessing scenario	Accuracy (%)
Baseline	Full preprocessing	75.05
	Without stemming	76.55
	Without stop word removal	78.06
	Without stemming and without stop word removal	79.91
Proposed improvement of slang word data	Full preprocessing	75.89
	Without stemming	77.70
	Without stop word removal	80.49
	Without stemming and without stop word removal	81.38

6. Conclusions

An algorithm-level approach is proposed to handle imbalanced data on Indonesian-language Twitter for a multi-label text classification task. It uses a classifier of a pre-trained IndoBERT model by optimizing hyperparameters. Several scenarios were carried out to determine the one that presented the best multi-label classification model for detecting hate speech and abusive language.

The dataset used in this research requires a new class of "normal" so that the equation for the accuracy based on logical operators can be calculated. The best-obtained combination of hyperparameters was 4e-5 for the learning rate, 16 for the batch size, 0.1 for the dropout, and 4 for the epoch. This was achieved with a data preprocessing scenario without stemming and without stop word removal, with an accuracy of 81.38%. However, the use of excessive dropout is not recommended because it can reduce the performance of the model. It also results in a failed prediction of low-distributed labels.

For future work, it is suggested that the grid search method be used to explore the combination of hyperparameters. It is also recommended that the slang word data be improved as a normalization dictionary so that the classification becomes more effective. Another improvement can be made using a 24-layer monolingual model as an alternative model architecture and integrating external knowledge bases or leveraging cross-lingual transfer learning techniques. Lastly, it is recommended to use a significance test to see the impact of the accuracy improvement.

Acknowledgment

This research work was funded by Institutional Fund Projects under grant no. (IFPIP: 1192-611-1443). The authors gratefully acknowledge the technical and financial support provided by the Ministry of Education and King Abdulaziz University, DSR, Jeddah, Saudi Arabia.

Compliance with ethical standards

Conflict of interest

The author(s) declared no potential conflicts of interest with respect to the research, authorship, and/or publication of this article.

References

Alfina I, Mulia R, Fanany MI, and Ekanata Y (2017). Hate speech detection in the Indonesian language: A dataset and preliminary study. In the International Conference on Advanced Computer Science and Information Systems, IEEE, Bali, Indonesia: 233-238. https://doi.org/10.1109/ICACSIS.2017.8355039 **PMid:35875654 PMCid:PMC9299239**

Bramantoro A and Virdyna I (2022). Classification of divorce causes during the COVID-19 pandemic using convolutional neural networks. PeerJ Computer Science, 8: e998. https://doi.org/10.7717/peerj-cs.998

El Kafrawy P, Mausad A, and Esmail H (2015). Experimental comparison of methods for multi-label classification in different application domains. International Journal of Computer Applications, 114: 19. https://doi.org/10.5120/20083-1666

Fernández A, García S, Galar M, Prati RC, Krawczyk B, and Herrera F (2018). Learning from imbalanced data sets. Springer, Berlin/Heidelberg, Germany. https://doi.org/10.1007/978-3-319-98074-4

Hana KM, Al Faraby S, and Bramantoro A (2020). Multi-label classification of Indonesian hate speech on Twitter using support vector machines. In the International Conference on Data Science and Its Applications, IEEE, Bandung, Indonesia: 1-7. https://doi.org/10.1109/ICoDSA50139.2020.9212992

Hendrawan R, Adiwijaya, and Al Faraby S (2020). Multilabel classification of hate speech and abusive words on Indonesian Twitter social media. In the International Conference on Data Science and Its Applications, IEEE, Bandung, Indonesia: 1-7. https://doi.org/10.1109/ICoDSA50139.2020.9212962

Hinton GE, Srivastava N, Krizhevsky A, Sutskever I, and Salakhutdinov RR (2012). Improving neural networks by preventing co-adaptation of feature detectors. ArXiv Preprint ArXiv:1207.0580. https://doi.org/10.48550/arXiv.1207.0580

Ibrohim MO and Budi I (2019). Multi-label hate speech and abusive language detection in Indonesian Twitter. In the 3rd Workshop on Abusive Language Online, Association for Computational Linguistics, Florence, Italy: 46-57. https://doi.org/10.18653/v1/W19-3506

Johnson JM and Khoshgoftaar TM (2019). Survey on deep learning with class imbalance. Journal of Big Data, 6: 27. https://doi.org/10.1186/s40537-019-0192-5

Kingma DP and Ba JL (2015). Adam: A method for stochastic optimization. ArXiv Preprint ArXiv:1412.6980. https://doi.org/10.48550/arXiv.1412.6980

Kovács G, Alonso P, and Saini R (2021). Challenges of hate speech detection in social media: Data scarcity, and leveraging external resources. SN Computer Science, 2: 95. https://doi.org/10.1007/s42979-021-00457-3

Li H, Li J, Guan X, Liang B, Lai Y, and Luo X (2019). Research on overfitting of deep learning. In the 15th International Conference on Computational Intelligence and Security, IEEE, Macao, China: 78-81. https://doi.org/10.1109/CIS.2019.00025

Makruf M, Bramantoro A, Alyamani HJ, Alesawi S, and Alturki R (2021). Classification methods comparison for customer churn prediction in the telecommunication industry. International Journal of Advanced and Applied Sciences, 8(12): 1-8. https://doi.org/10.21833/ijaas.2021.12.001

Niemann M, Riehle DM, Brunk J, and Becker J (2020). What is abusive language? Integrating different views on abusive language for machine learning. In: Grimme C, Preuss M, Takes F, and Waldherr A (Eds.), Multidisciplinary international symposium on disinformation in open online media: 59-73. Springer International Publishing, Cham, Switzerland. https://doi.org/10.1007/978-3-030-39627-5_6

Prabowo FA, Ibrohim MO, and Budi I (2019). Hierarchical multi-label classification to identify hate speech and abusive language on Indonesian Twitter. In the 6th International Conference on Information Technology, Computer and Electrical Engineering, IEEE, Semarang, Indonesia: 1-5. https://doi.org/10.1109/ICITACEE.2019.8904425

Pratondo A and Bramantoro A (2022). Classification of *Zophobas morio* and *Tenebrio molitor* using transfer learning. PeerJ Computer Science, 8: e884. https://doi.org/10.7717/peerj-cs.884 **PMid:35494845 PMCid:PMC9044276**

Putra IF and Purwarianti A (2020). Improving Indonesian text classification using multilingual language model. In the 7th International Conference on Advance Informatics: Concepts, Theory and Applications, IEEE, Tokoname, Japan: 1-5. https://doi.org/10.1109/ICAICTA49861.2020.9429038 **PMid:32268239**

Ramyachitra D and Manikandan P (2014). Imbalanced dataset classification and solutions: A review. International Journal of Computing and Business Research, 5(4): 1-29.

Saputri MS, Mahendra R, and Adriani M (2018). Emotion classification on Indonesian Twitter dataset. In the International Conference on Asian Language Processing, IEEE, Bandung, Indonesia: 90-95. https://doi.org/10.1109/IALP.2018.8629262

Srivastava N, Hinton G, Krizhevsky A, Sutskever I, and Salakhutdinov R (2014). Dropout: A simple way to prevent neural networks from overfitting. The Journal of Machine Learning Research, 15(1): 1929-1958.

Srividhya V and Anitha R (2010). Evaluating preprocessing techniques in text categorization. International Journal of Computer Science and Application, 47(11): 49-51.

Warner W and Hirschberg J (2012). Detecting hate speech on the world wide web. In the 2nd Workshop on Language in Social Media, Association for Computational Linguistics, Montreal, Canada: 19-26.

Wilie B, Vincentio K, Winata GI, Cahyawijaya S, Li X, Lim ZY, Soleman S, Mahendra R, Fung P, Bahar S, and Purwarianti A (2020). IndoNLU: Benchmark and resources for evaluating Indonesian natural language understanding. In the 1st Conference of the Asia-Pacific Chapter of the Association for Computational Linguistics and the 10th International Joint Conference on Natural Language Processing, Suzhou, China: 843–857.

Winata GI and Khodra ML (2015). Handling imbalanced dataset in multi-label text categorization using bagging and adaptive boosting. In the International Conference on Electrical Engineering and Informatics, IEEE, Denpasar, Indonesia: 500-505. https://doi.org/10.1109/ICEEI.2015.7352552