Text Preprocessing Approaches in CNN for Disaster Reports Dataset

Andriansyah Oktafiandi Arisha
Dept. of Computer System
Handayani University
Makassar, Indonesia
ecpand@gmail.com

Hazriani
Dept. of Computer System
Handayani University
Makassar, Indonesia
hazriani@handayani.ac.id

Yuyun Wabula
Dept. of Computer System
Handayani University
Makassar, Indonesia
yuyunwabula@handayani.ac.id

***Abstract*—This study aims to compare the performance of the text-preprocessing methods namely automatic and semi-automatic preprocessing techniques in the CNN algorithm to carry out learning on disaster report dataset. The experimental results on the disaster dataset with a total of 200 records with the automatic text preprocessing technique produce an average accuracy of 0.81 and 1 with training data of 80:20 and 90:10. While in the optimize model that is semi-automatic text preprocessing approach (which is the author's proposed approach), the average accuracy obtained are 0.95 and 1 for dataset 80:20 and 90:10. The experimental results conclude that cleaning the dataset with the semi-automatic text preprocessing model can improve accuracy compared to the previous model. The proposed model will get convergence with 80:20 training data at epoch 20, batch size 5 and random state 34, while for dataset 90:10 the best convergence value at epoch 20-30.**

Keywords—Text Preprocessing, CNN, Disaster, automatic, semi-automatic

I. INTRODUCTION

Disaster reporting system is an important phase in emergency response prevention efforts. Currently, almost all forms of disaster reports are in text format. Data collection techniques by volunteers from interviews, documentation and field observations written in the form of report documents [1] are time consuming and more expensive. As an alternative, a real-time data sourced from a web-based reports or social media site capture individual activity [2] can be used by disaster service organization to obtain information. However, data structure sourced from these sites has a weakness since most of them are unstructured information [3]. Working with unstructured data will burden the stakeholders in making a quick decision. To process and analyze this type of data requires a certain algorithm. One promising way to solve this issue is by applying text mining techniques, namely how to extract information from a set of text documents [4].

Studies at the big data scale allow changes to the way documents are analyzed. Generally by applying machine learning techniques and natural language processing (NLP). Studies have proven that this technique produces rich information, and is believed to be used to handle large amounts of disaster-related data [5]. In its application, these data require engineering techniques in handling the volume, variety and veracity of the data [6], [7]. To deal with these problems, it requires text preprocessing techniques, namely cleaning, imputation, scaling, as well as imbalance handling in order to produce a good analytical results [7]. Thus the more comprehensive the dataset features, the more accurate the machine in absorbing knowledge [8], [9].

. In extracting knowledge on text features, various preprocessing techniques are used [10] such as: *tokenization* a process of dividing text into certain parts [11]; *stemming* by returning words to basic word forms [12]; *case folding* that is changing text to lowercase; *removing* punctuation marks, numbers and blank characters [13]; and *filtering* or process of removing unnecessary words. The tendency in the filtering process only explores conjunctions, propositions, and adverbs [14], [15], [16], [17], [18]. The filtering technique does not remove the preprocessing substance words. Another drawback of the filtering technique is that it doesn't have a significant effect on the accuracy of the text classification results [19].

The purpose of this study is to examine the effect of the text *semi-preprocessing* technique (which is the author's proposed approach) on the accuracy of the analysis. Our first work is to design a model for data cleaning in a disaster reporting system by comparing the performance of the automatic text preprocessing algorithms and the *semi-automatic* text preprocessing. Second work is we create a grammatical corpus by exploring word relations to identify the features of the words related to disaster text based on the *semi-automatic* preprocessing model. In this study, disaster datasets of online reports titles are classified using the Convolutional Neural Network (CNN) algorithm and then evaluated using the Confusion matrix..

II. RELATED WORK

Based on literature studies, almost all disaster research using text document datasets sourced from social media sites. For example, research conducted by [8] identifying types of information during a disaster. This research compared the performance of two classification algorithms, namely CNN and Naïve Bayes. From all the experiments it was concluded that CNN works more effectively. In another study, [20] developed several automated machine learning models to detect disaster-related information in user posting text. Two algorithms are compared, Naive Bayes and Support Vector Machine (SVM). The model generated from SVM has much better results compared to Naive Bayes. Similarly [21] uses regression analysis to predict the type of damage that will come due to a disaster. Their analysis concluded that data on micro blogging services has the potential as an additional tool for emergency services. Then [22] proposed the BERT algorithm and [23] combined BERT and LSTM in classifying user-generated content. The results show that combining the two algorithms give better performance compared to using only one algorithm.

Furthermore, [24] combines the twitter and Instagram data sets related to earthquakes. Three scenarios were used to evaluate the CRF model, namely: CRF combined with LSTM CRF, Optimization of CRF, and combination of LSTM with CRF. The results show that CRF Optimization is superior to other models. To evaluate this model [24] developed a natural language processing (NLP) model to analyze a collection of disaster recovery texts. The presented method uses statistical syntax-based semantic matching. The results of this study

show that this model is effective when applied to disasters that contain a news corpus. Research by [20] using eyewitness information sources on social media by perfecting linguistic grammar rules to extract features of disaster text. In the first trial, adopting manual classification produced an F-Score of 0.81. Then the model was improvised with the LR-TED approach, and produced an F-score of 0.93. The advantage of this method is that it can process millions of tweets in real-time and can predict various types of disasters in the future. The same approach was done by [25] extracting information during a disaster. Three machine learning models were used to analyze the data, namely: classification, clustering and extraction. At the classification stage, the proposed models are sLDA, SVM, and logistic regression. Then the clustering stage uses filtering techniques to identify word relationships, and feature extraction focuses on infrastructure damage, types of damage, and casualties. This research concluded that the proposed model resulted an imbalance label/class. Then [23] compares the performance of the SVM and Naïve Bayes algorithms in classifying disaster text. Two class targets as output analysts are informative and non-informative tweets. Based on the method proposed in this study, the Support Vector Machine Algorithm produces an accuracy of 81.03%. superior to the Naïve Bayes algorithm which produces an accuracy of 80.30%. Then [26] proposed Convolutional Neural Networks (CNN) and BERT for the classification of disaster tweets. The proposed method is to train text to classify objects independently. This study concludes that combining models can improve the performance of the CNN and BERT algorithms for self-training data.

Having a deeper analysis, it can be concluded that those previous studies focused on measuring the performance of algorithms with disaster text datasets sourced from social networking platforms. Most of them uses an automatic text preprocessing techniques, namely how the data is cleaned. Another challenge issue, it is found in the previous study that there is a high gap in accuracy even though using the same algorithm as presented in Table 1. This issue lead to a doubt whether the dataset used are free from noise or not. Another consideration is that processing text is difficult because the language structure of social media sites has uncertainty [22]. This indicates that the great success of an algorithm in processing data is influenced by the quality of the dataset [27].

TABLE I. COMPARISON OF TEXT MINING APPROACHES ON DISASTER DATASET

Researches	Applications	Preprocessing Techniques	Accuracy
Venkata Kishore Neppalli et al. [8]	Deep Neural Networks versus Naïve Bayes Classifiers	Bag-of-words, content features, user-based features, and polarity-based features.	0.84; 0.87
Beverly Estephany Parilla-Ferrer et al. [28]	Naive Bayes and Support Vector Machine (SVM)	Tokenization, stemming, and stop words removal	0.56; 0.80
Guoqin Ma [29]	Combination BERT and LSTM	Tokenization	0.67
A K Ningsih et al. [22]	BERT	Preprocess_kgptalkie	0.8
Hathairat Ketmaneechairat et al. [30]	CRF and LSTM	Tokenization	0.98
Sajjad Haider et al. [20]	LR-TED, ANN, RNN, CNN	Tokens, pos, lemmatization, deprel	0,93.
Zahra Ashktorab et al. [25]	sLDA, SVM, and Logistic Regression	Tokenization	Avg. 0.81
Windu Gata et al. [23]	SVM and Naïve Bayes	Anotation Removal, Transformation Remove URL, Tokenization, Transformation Not Negative, Indonesian Stemming, Indonesian Stopword Removal	81.03; 80.30
Hongmin Li et al. [26]	CNN and BERT	GloVe's Ruby preprocessing script	89.01; 89.01; 93.82; 91.28

III. RESEARCH METHODOLOGY

A. Data Collections

The dataset in this study is sourced from online news sites, especially the new title that contain disaster information (*in Indonesian language*). In this study, the data collection method uses web scraping techniques. A total of 200 data were collected and divided into 3 classes. The dataset testing model is divided into 2 categories, namely *automatic preprocessing* and *semi- automatic preprocessing*.

B. Preprocessing

The classification technique is heavily influenced by the learning dataset. To ensure the quality of the data, the dataset requires cleaning before it before analysis. The preprocessing technique used in the research are:

1) Tokenization: is the technique of dividing text in sentences into the smallest units as in the following example.

Banjir Lumpur Terjadi di Objek Wisata Baby Volcano, Tanggul Pembatas Rusak	[banjir, lumpur, terjadi, di, objek wisata, baby, volcano, tanggul, pembatas, rusak]
Banjir Lumpur Menutup Jalan Nasional Bandung-Purwokerto di Cilacap	[banjir, lumpur, menutup, jalan, nasional, bandung-purwokerto di, cilacap]

2) Case folding: namely changing text to lowercase in the following example.

Banjir Lumpur Terjadi di Objek Wisata Baby Volcano, Tanggul Pembatas Rusak	banjir lumpur terjadi di objek wisata baby volcano, tanggul pembatas rusak
Banjir Lumpur Menutup Jalan Nasional Bandung-Purwokerto di Cilacap	banjir lumpur menutup jalan nasional bandung-purwokerto di cilacap

3) Filtering/ Stop Word removal: is the process of selecting data by removing unnecessary words such as connecting words between sentences, punctuation marks, prepositions and symbols in the following example

[banjir, lumpur, terjadi, di, objek wisata, baby, volcano, tanggul, pembatas, rusak]	banjir lumpur terjadi di objek wisata baby volcano, tanggul pembatas rusak
[banjir, lumpur, menutup, jalan, nasional, bandung-purwokerto di, cilacap]	banjir lumpur menutup jalan nasional bandung purwokerto cilacap

C. Annotation Labels

To train and evaluate the proposed model, the annotation label of the disaster dataset is done manually. The

This Research is Funded by Indonesian National Research and Innovation Agency "BRIN".

classification of labels is divided into 3 classes, namely the National Search and Rescue Agency (*Basarnas*), Regional Disaster Management Agency (*BPBD*), and Fire Department (*Damkar*). The purpose of classifying this label is to facilitate coordination in reporting disaster information. This annotation process considers the suitability of the text on 3 research labels. Class suitability in each field refers to the main tasks and functions [31], [32].

D. Proposed Annotation Approach

In this study we propose a Semi-automatic preprocessing (SAP) technique as presented in Figure 1. The work process is to remove words that are irrelevant to the aftermath of the disaster through an automatic preprocessing technique.

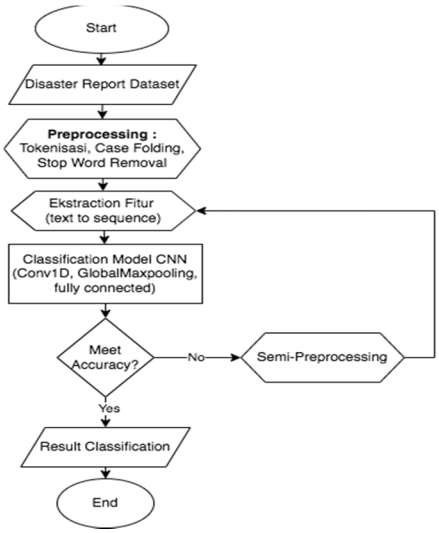


Fig. 1. Flowchart of the semi-automatic preprocessing (SAP)

E. Convolutional Neural Network (CNN)

The CNN algorithm in this study is intended to classify the disaster datasets. Although this algorithm is good for classifying images, we found that many researchers have used the CNN algorithm to classify datasets that originate from text documents. Such as research [26] in classifying disaster or crisis texts, identifying eyewitness reports on the emergency response system [20] and classifying disaster information [8]. They concluded that this algorithm can process millions of text data in real-time and has good performance.

In forming a classification model, CNN requires *hyper-parameters* to produce a model with the best performance. Four hyper-parameters are used to test the proposed model, including: *learning rate* to measure how fast the algorithm learns to calculate the corrected weight values during the training process; *batch size* functions to calculate the number of samples to be trained in one process; *epoch* to set the number of iterations during the training process for the entire data; and random state to define how the data training and data testing are pickup prom the dataset. To implement this model, we use the python programming language with the *Keras* libraries. In this simulation the process of converting text into numeric numbers uses the *text to sequence* approach, in which the weight of each word is determined based on its order in the dataset.

IV. EXPERIMENTAL SETUP

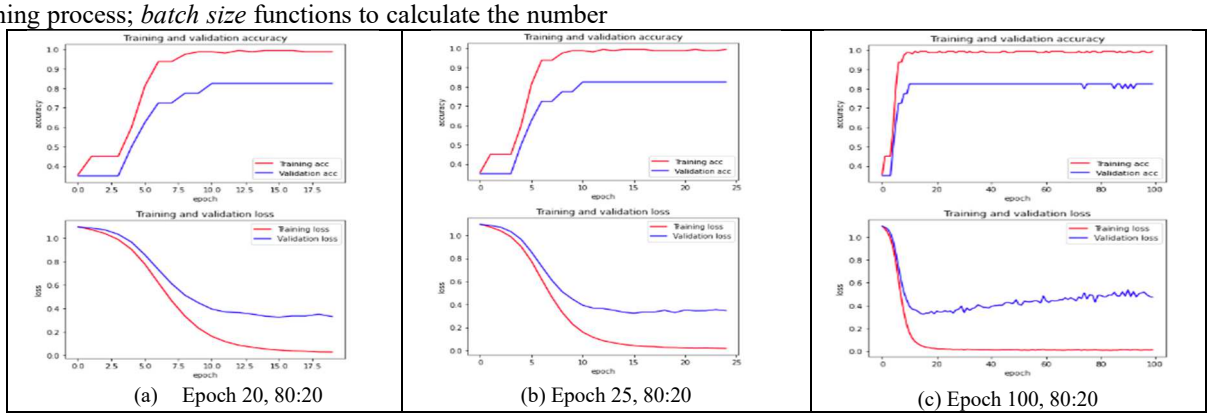
The experiments were conducted using series combination of four *hyper-parameters* on CNN, namely learning rate, bath size, epochs and Random State. These four parameters are used to test the performance comparison of *automatic text processing* and *semi- automatic text preprocessing* techniques to disaster text input data. Combination of the experimental parameters including their performance of each preprocessing technique are recapitulate in Table II and Table III, as well as its comparison in Fig 2 & Fig. 3. Where *DS (*total record of dataset), TRD (data training, TSD (data testing), LR (learning rate), EP (epoch), BS (batch size), and RS (random state).

TABLE II. EXPERIMENTAL SETUP AND RESULT OF THE *AUTOMATIC* TEXT PREPROCESSING

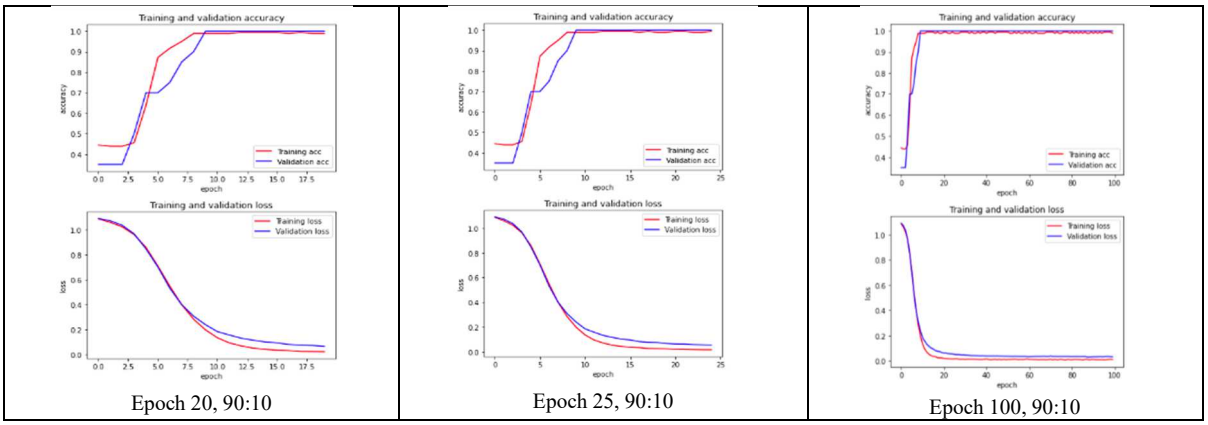
DS	TRD (%)	TSD (%)	LR	EP	BS	RS	Acc	Loss
200	80	20	0.001	10	5	34	0.77	0.44
				15	5	34	0.82	0.33
				20	5	34	0.82	0.32
				25	5	34	0.82	0.34
				100	5	34	0.82	0.47
	90	10	0.001	10	5	34	1	0.24
				15	5	34	1	0.10
				20	5	34	1	0.06
				25	5	34	1	0.05
				100	5	34	1	0.03

TABLE III. EXPERIMENTAL SETUP AND RESULT OF THE *SEMI-AUTOMATIC* TEXT PREPROCESSING

DS	TRD (%)	TSD (%)	LR	EP	BS	RS	Acc	Loss
200	80	20	0.001	10	5	34	0.94	0.32
				15	5	34	0.97	0.16
				20	5	34	0.97	0.12
				25	5	34	0.97	0.13
				100	5	34	0.94	0.19
	90	10	0.001	10	5	34	1	0.14
				15	5	34	1	0.05
				20	5	34	1	0.03
				25	5	34	1	0.03
				100	5	34	1	0.02

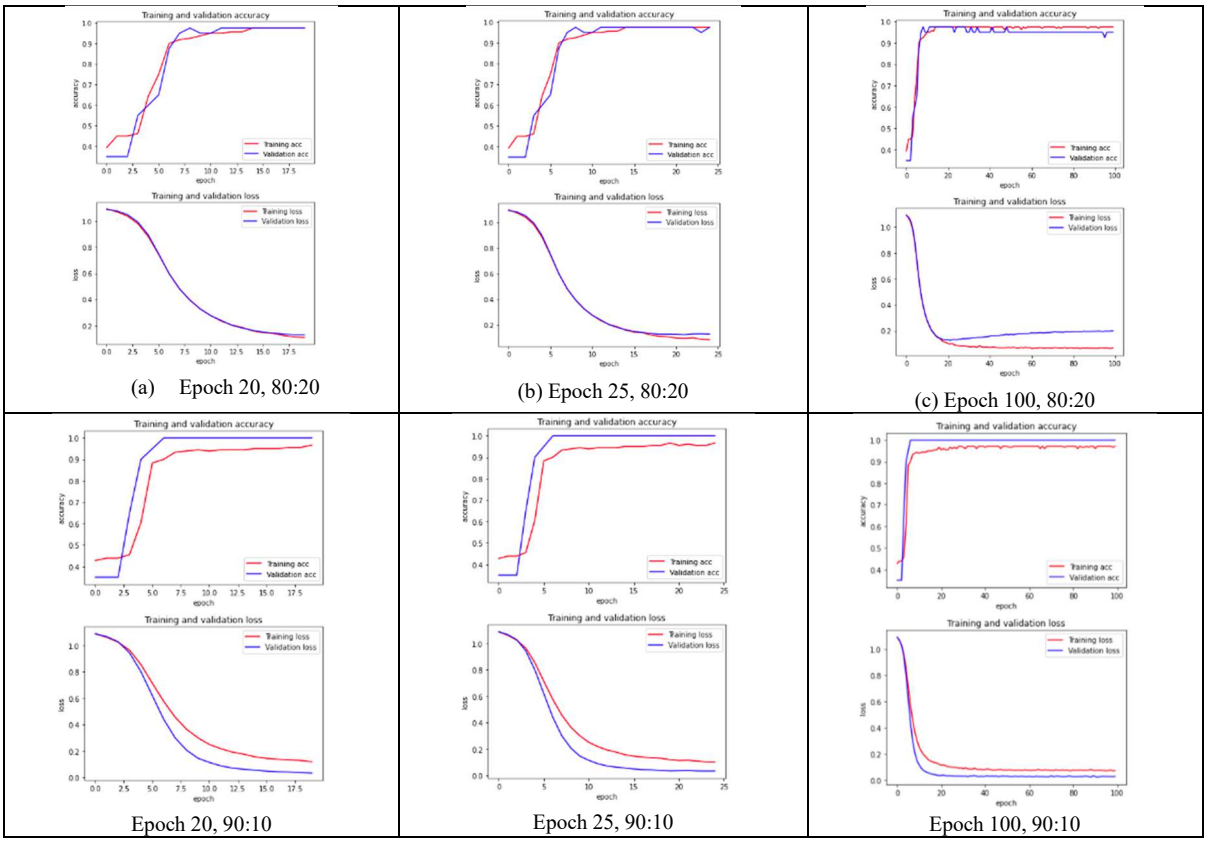


(a) Epoch 20, 80:20 (b) Epoch 25, 80:20 (c) Epoch 100, 80:20



Epoch 20, 90:10 | Epoch 25, 90:10 | Epoch 100, 90:10

Fig. 2. comparison of Automatic Text Preprocessing Techniques: data training to data testing 80%: 20% and 90%:10%.



(a) Epoch 20, 80:20 | (b) Epoch 25, 80:20 | (c) Epoch 100, 80:20

Epoch 20, 90:10 | Epoch 25, 90:10 | Epoch 100, 90:10

Fig. 3. comparison of Semi-Automatic Text Preprocessing Techniques: data training to data testing 80%: 20% and 90%:10%.

V. DISCUSSION

The study in [30] proposes 2 algorithms namely CRF and LSTM, the automatic preprocessing technique i.e. tokenization is used with accuracy reaches 0.98. Then [20] proposed four preprocessing techniques, namely tokenization, pos, lemmatization and deprel. Four classification algorithms were compared, namely LR-TED, ANN, RNN, CNN. The accuracy obtained reaches 0.93. Another approach in [26] proposes CNN and BERT algorithms with GloVe's Ruby preprocessing script for preprocessing technique. The comparison results produce an accuracy of 89.01; 89.01; 93.82; 91.28. In our approach we compare the performance of automatic preprocessing, namely the tokenization technique, case folding and filtering with our proposed algorithm the *semi-automatic preprocessing* (SAP). This algorithm works by eliminating irrelevant words in the disaster reports dataset after going through the automatic preprocessing stage. In testing this model, the CNN Algorithm is used to classify the research dataset. Four *hyper-parameters* are used to test the proposed model, namely learning rate, batch size, epoch, and random state. In this simulation, we set the learning rate by default to be 0.001, bath size 5, and random state 34. Epoch values are varied, namely 10, 15, 20, 25, and 100 see Table II and Table. Recapitulation of the evaluation metrics are presented in Table IV.

EXP	Technique	TRD:TSD (%)	Average (Acc)	Average (Lost)
I	Automatic Preprocessing	80:20	0.81	0.38
II		90:10	1	0.09
III	Semi- Automatic Preprocessing	80:20	0.95	0.18
IV		90:10	1	0.05

VI. CONCLUSION

Based on the experiment results on 200 records of the disaster dataset using the automatic preprocessing technique, the average accuracy was 0.81 and 1 with training data 80:20 and 90:10. Then we optimize the model with the semi-automatic preprocessing technique, which gives an average accuracy of 0.95 and 1 with training data of 80:20 and 90:10. It can be concluded that our proposed approach i.e. cleaning the dataset with the *semi-automatic* preprocessing model can improve accuracy compared to the previous model. The proposed model will get convergence with 80:20 training data on epoch 20, batch size 5 and random state 34. Then for 90:10 the convergence value find on epoch 20-30. Since, the accuracy is strongly influenced by the number of datasets, for future work it is necessary to test more datasets.

ACKNOWLEDGMENT

This work is supported by Indonesian government through National competitive research grants, organized by the National Research and Innovation Agency "BRIN".

REFERENCES

[1] B. U. S. Gina, "Modul 2 Manajemen Penanggulangan Bencana".

[2] Yuyun, F. Akhmad Nuzir, and B. Julien Dewancker, "Dynamic Land-Use Map Based on Twitter Data," Sustainability, vol. 9, no. 12, p. 2158, Nov. 2017, doi: 10.3390/su9122158.

[3] A. Dilo and S. Zlatanova, "Data modelling for emergency response".

[4] S. Dang and P. H. Ahmad, "Text Mining: Techniques and its Application," vol. 1, no. 4, 2014.

[5] J. Qadir, A. Ali, R. ur Rasool, A. Zwitter, A. Sathiaseelan, and J. Crowcroft, "Crisis analytics: big data-driven crisis response," Int J Humanitarian Action, vol. 1, no. 1, p. 12, Dec. 2016, doi: 10.1186/s41018-016-0013-9.

[6] R. Kitchin and G. McArdle, "What makes Big Data, Big Data? Exploring the ontological characteristics of 26 datasets," Big Data & Society, vol. 3, no. 1, p. 205395171663113, Jun. 2016, doi: 10.1177/2053951716631130.

[7] E. Cho, T.-W. Chang, and G. Hwang, "Data Preprocessing Combination to Improve the Performance of Quality Classification in the Manufacturing Process," Electronics, vol. 11, no. 3, p. 477, Feb. 2022, doi: 10.3390/electronics11030477.

[8] V. K. Neppalli, C. Caragea, and D. Caragea, "Deep Neural Networks versus Naïve Bayes Classifiers for Identifying Informative Tweets during Disasters," 2018.

[9] M. Srivastava, R. Garg, and P. K. Mishra, "Analysis of Data Extraction and Data Cleaning in Web Usage Mining," in Proceedings of the 2015 International Conference on Advanced Research in Computer Science Engineering & Technology (ICARCSET 2015) - ICARCSET '15, Unnao, India, 2015, pp. 1–6. doi: 10.1145/2743065.2743078.

[10] Z. Jianqiang and G. Xiaolin, "Comparison Research on Text Pre-processing Methods on Twitter Sentiment Analysis," IEEE Access, vol. 5, pp. 2870–2879, 2017, doi: 10.1109/ACCESS.2017.2672677.

[11] R. M. Kaplan, "A Method for Tokenizing Text".

[12] A. Arif siswandi, Y. Permana, and A. Emarilis, "Stemming Analysis Indonesian Language News Text with Porter Algorithm," J. Phys.: Conf. Ser., vol. 1845, no. 1, p. 012019, Mar. 2021, doi: 10.1088/1742-6596/1845/1/012019.

[13] Rianto, A. B. Mutiara, E. P. Wibowo, and P. I. Santosa, "Improving the accuracy of text classification using stemming method, a case of non-formal Indonesian conversation," J Big Data, vol. 8, no. 1, p. 26, Dec. 2021, doi: 10.1186/s40537-021-00413-1.

[14] B. Alhadidi and M. Alwedyan, "Hybrid Stop-Word Removal Technique for Arabic Language," vol. 30, no. 1, 2008.

[15] L. Dolamic and J. Savoy, "When stopword lists make the difference," J. Am. Soc. Inf. Sci., vol. 61, no. 1, pp. 200–203, Jan. 2010, doi: 10.1002/asi.21186.

[16] S. Ferilli, "Automatic Multilingual Stopwords Identification from Very Small Corpora," Electronics, vol. 10, no. 17, p. 2169, Sep. 2021, doi: 10.3390/electronics10172169.

[17] C. Bhadane, H. Dalal, and H. Doshi, "Sentiment Analysis: Measuring Opinions," Procedia Computer Science, vol. 45, pp. 808–814, 2015, doi: 10.1016/j.procs.2015.03.159.

[18] A. Amalia, W. Oktinas, I. Aulia, and R. F. Rahmat, "Determination of quality television programmes based on sentiment analysis on Twitter," J. Phys.: Conf. Ser., vol. 978, p. 012117, Mar. 2018, doi: 10.1088/1742-6596/978/1/012117.

[19] A. W. Pradana and M. Hayaty, "The Effect of Stemming and Removal of Stopwords on the Accuracy of Sentiment Analysis on Indonesian-language Texts," KINETIK, pp. 375–380, Oct. 2019, doi: 10.22219/kinetik.v4i4.912.

[20] S. Haider, A. Mahmood, S. Khatoon, M. Alshamari, and M. T. Afzal, "Automatic Classification of Eyewitness Messages for Disaster Events Using Linguistic Rules and ML/AI Approaches," Applied Sciences, vol. 12, no. 19, p. 9953, Oct. 2022, doi: 10.3390/app12199953.

[21] M. M. Bala, K. Navya, and P. Shruthilaya, "Text Mining in Real Time Twitter Data for Disaster Response".

[22] A. K. Ningsih and A. I. Hadiana, "Disaster Tweets Classification in Disaster Response using Bidirectional Encoder Representations from Transformer (BERT)," IOP Conf. Ser.: Mater. Sci. Eng., vol. 1115, no. 1, p. 012032, Mar. 2021, doi: 10.1088/1757-899X/1115/1/012032.

[23] W. Gata, F. Amsury, N. K. Wardhani, I. Sugiyarto, D. N. Sulistyowati, and I. Saputra, "Informative Tweet Classification of the Earthquake Disaster Situation In Indonesia," in 2019 5th International Conference on Computing Engineering and Design (ICCED), Singapore, Singapore, Apr. 2019, pp. 1–6. doi: 10.1109/ICCED46541.2019.9161135.

[24] L. H. Lin, S. B. Miles, and N. A. Smith, "Natural Language Processing for Analyzing Disaster Recovery Trends Expressed in Large Text Corpora," in 2018 IEEE Global Humanitarian Technology Conference (GHTC), San Jose, CA, Oct. 2018, pp. 1–8. doi: 10.1109/GHTC.2018.8601884.

[25] Z. Ashktorab, C. Brown, M. Nandi, and A. Culotta, "Tweedr: Mining Twitter to Inform," 2014.

[26] H. Li, D. Caragea, and C. Caragea, "Combining Self-training with Deep Learning for Disaster Tweet Classification," 2021.

[27] "15 OKC classifier an efficient approach for classification of imbalanced dataset using hybrid methodology.txt."

[28] "(15) (PDF) Automatic Classification of Disaster-Related Tweets." https://www.researchgate.net/publication/282154924_Automatic_Classification_of_Disaster-Related_Tweets (accessed Dec. 22, 2022).

[29] G. Ma, "Tweets Classification with BERT in the Field of Disaster Management".

[30] College of Industrial Technology Faculty and Information Technology Faculty, King Mongkut's University of Technology North Bangkok, Thailand, H. Ketmaneechairat, and M. Maliyaem, "Natural Language Processing for Disaster Management Using Conditional Random Fields," JAIT, pp. 97–102, 2020, doi: 10.12720/jait.11.2.97-102.

[31] "32 BPBD.txt."

[32] "33 Basarnas.txt."