

A multi domains short message sentiment classification using hybrid neural network architecture

Devi Munandar, Andri Fachrur Rozie, Andria Arisal

Research Center for Informatics, Indonesian Institute of Sciences, Bandung, Indonesia

Article Info

Article history:

Received Jan 10, 2021

Revised May 20, 2021

Accepted Jun 3, 2021

Keywords:

CNN

Domain-specific

LSTM

MLP

Sentiment analysis

ABSTRACT

Sentiment analysis of short texts is challenging because of its limited context of information. It becomes more challenging to be done on limited resource language like Bahasa Indonesia. However, with various deep learning techniques, it can give pretty good accuracy. This paper explores several deep learning methods, such as multilayer perceptron (MLP), convolutional neural network (CNN), long short-term memory (LSTM), and builds combinations of those three architectures. The combinations of those three architectures are intended to get the best of those architecture models. The MLP accommodates the use of the previous model to obtain classification output. The CNN layer extracts the word feature vector from text sequences. Subsequently, the LSTM repetitively selects or discards feature sequences based on their context. Those advantages are useful for different domain datasets. The experiments on sentiment analysis of short text in Bahasa Indonesia show that hybrid models can obtain better performance, and the same architecture can be directly used in another domain-specific dataset.

This is an open access article under the [CC BY-SA](#) license.



Corresponding Author:

Devi Munandar

Research Center for Informatics

Indonesian Institute of Sciences

Jl. Sangkuriang, Bandung 40135, Indonesia

Email: devi.munandar@lipi.go.id

1. INTRODUCTION

The public is discussing various issues through different media platforms. Social media is one of the media channels where public users can share their opinions. However, it inherits some limitations, such as the media types, including video, audio, text, or emoticons, with maximum text length, file format, and other limitations. The various limitations of the social media platform could become another problem for social media analytic. Thus, our research focused on opinions in short text messages shared by a user through social media channels (especially Twitter). It is because the decision-makers are only interested in the sentiment of opinions exchanged during the conversations in social media, not the detailed messages in the discussion. Nowadays, public opinions, in the form of trending topics or virality, could drive and change public policy.

Sentiment analysis is a study to understand people's opinions, sentiments, emotions on something from text messages [1]. The object of emotions might includes products, services, persons, or other topics of conversation. Sentiment analysis of short texts is challenging because its context might present in other short messages or other conversations. Nevertheless, researchers have done various research in sentiment analysis using a statistical model with lexicon methods [2]-[4], machine learning [5]-[9], to semi-supervised methods with deep learning [10]-[12]. In order to get good sentiment accuracy, researchers often come with additional techniques such as data tuning and model tuning. Data tuning uses domain-specific data, which is trained, and tuned to build models on that specific domain. This technique expects overfitting and a good result

within that specific domain. On the other hand, model tuning uses various algorithms and combinations to get a new model, which gives results with better accuracy. On deep learning methods, reordering architectures could build a model with better results. For example, combining convolution neural network (CNN) for its coarse-grained features extractions and recurrent neural network (RNN) for its sequential features relationship could give better results [13]. This research explores various architectures of deep neural networks for sentiment analysis. We use the architectures to build a satisfactory model for a specific domain using its domain-specific data and evaluate the model performance against another domain with its domain-specific dataset. For that purpose, we employ two different domains employment and telecommunication.

A great extent of employment issues is frequently discussed in social media, from compensation and employee tenure [14], wages and working locations [15], to the labor market and gender issues [16]-[17]. On the other hand, social media can be used to inform the worker about government policy, labor issues, and bridging communication between the public, laborers, and policymakers. The sentiment of those interactions is essential for policymakers. Another domain with frequent discussion is related to telecommunication operators are also frequently discussed in social media. Those can be related to operator performance on user service, bonuses, bandwidth availability, downtime, and connection quality. Mobile network operators can use the discussion to direct their business policy and services [18]-[20].

In this research, we explore various deep learning methods and architectures and a combination of methods to build models for sentiment analysis on short texts in Bahasa Indonesia. We use domain-specific datasets (on employment and telecommunication issues) for model evaluation and compare the models' performance on the dataset. This research has three-fold contributions in evaluating hybrid neural network methodologies for sentiment analysis on short text messages in Bahasa Indonesia and its application in different domain datasets. They are (1) explaining techniques in building models with hybrid neural network architecture with a domain-specific dataset to induce overfitting model; (2) evaluating the architecture on different domain with direct model implementation and model tuning with another domain-specific dataset; and (3) providing an experience for model migration from one domain to another.

The structure of this paper is presented by defining the purpose of the research in the first section, which is followed by short reviews of various sentiment analysis methods with additional emphasis on deep neural network techniques and hybridization of neural network architectures in the second section. The third section explains the research methods and experiments from data collection, pre-processing, and model building using various techniques for sentiment analysis to compare their analysis results. The fourth section gives experiment results and analysis. The conclusion is given in the last section.

2. HYBRID NEURAL NETWORK ARCHITECTURE FOR SENTIMENT ANALYSIS

Even though there is a limited natural language resource for Bahasa Indonesia, researchers have conducted various studies on sentiment analysis for text in Bahasa Indonesia, such as using lexicon-based techniques [21]-[22], machine learning [23]-[24], and deep learning [25]-[27]. Le *et al.* [25] use the LSTM and CNN for analyzing the sentiment of 900 thousand Indonesian tweets. They obtained 73.22% accuracy with a standard deviation of 1.39 using LSTM without normalization. Franky and Manurung [24] evaluate several classification techniques; Naive Bayes, Maximum Entropy, and Support Vector Machines using unigram and word frequency feature on Bahasa Indonesia translation of movie reviews. They achieved 78.82% accuracy, which they considered satisfactory due to simple translation compared to 80.09% accuracy when those techniques were applied directly to the original English movie reviews. Problems that often occur in the use of social media with Indonesian text are unstructured text data and non-standard languages. Putra, *et al.* [28] showed the use of hybrid models for sentiment analysis by distributing lexicon based on and maximum entropy gives a good evaluation score with 84.31% accuracy.

The hybrid neural network architecture is a combination of several neural network architecture not limited to multi-layer perceptron (MLP), CNN, or long short-term memory (LSTM). Figure 1 shows a hybrid neural network architecture that consists of MLP, CNN, and LSTM. The hybrid architecture is making use of the best of other architectures. The MLP is a simple model of neural networks that accommodates the use of the previous model to obtain classification output. In contrast, the CNN layer uses its local feature advantages to extract the word feature vector from text sequences. LSTM repetitively selects or discards feature sequences based on their context.

The feedforward method on MLP multiplies each input neuron with a weight to produce feature maps as a feature vector and passes it to the next layer through networks of neurons. It will carry the learning process before obtaining the β_1 vector as the output. On the other hand, CNN uses word vectors as the input dataset and processes them through convolutional layers, pooling layers, and fully connected layers. At the end of the CNN process, the fully connected layer produces β_2 vector map. The similar words vector is fed into the LSTM to be processed by extracting the input features using various filters and evaluating the

sequence of features within their context in a timely order. Thus, it produces another feature sequence β_3 . The output vectors β_1 from MLP, β_2 from CNN, and β_3 from LSTM are combined. The simplest combination function is by concatenating those outputs into the input of the classification process using the sigmoid activation function.

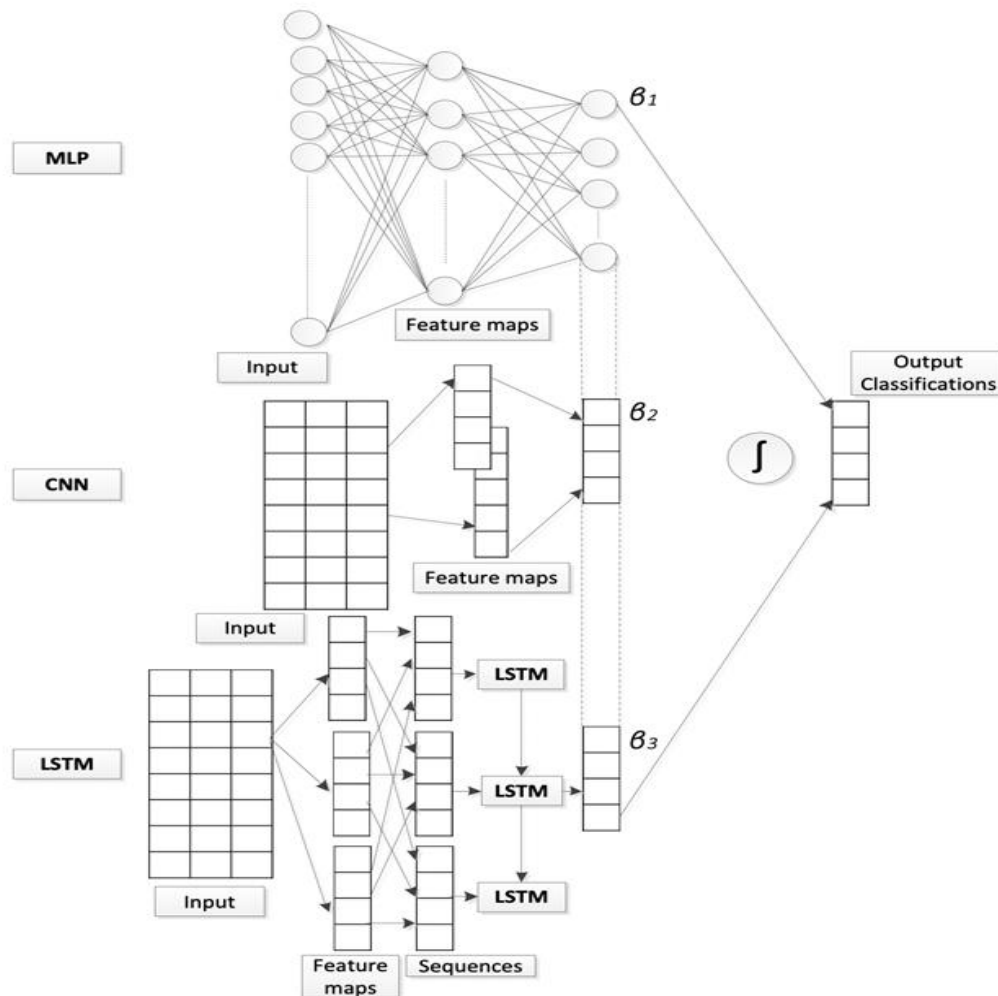


Figure 1. Hybrid neural network architecture

The basic concept of this hybridization is to concatenate the prior process's output as the input to the next one. In this study, we explore those three concept models; MLP, CNN, and LSTM. We use each architecture as the baseline of the neural network models and create a hybrid with the combination of MLP+CNN+LSTM. We store the result models of the last layer process on every single architecture as a parameter vector during the training phase. Those parameter vectors are merged to provide the final process for model classification output. We use sigmoid as the activation function for determining the classification output after the merging of the parameter vectors. It is because the sigmoid function can give a rational value output between 0 and 1. The sentiment value is determined by splitting a lower and higher fraction of the output value. The output value of 0 to 0.500 is considered a negative sentiment, while it is 0.5001 to 1 as a positive one.

3. RESEARCH METHOD

We evaluate our approach in the hybrid neural network to evaluate its effectiveness for sentiment analysis of short messages in Bahasa Indonesia, especially in employment and mobile telecommunication topics. The analytics phases are described in Figure 2.

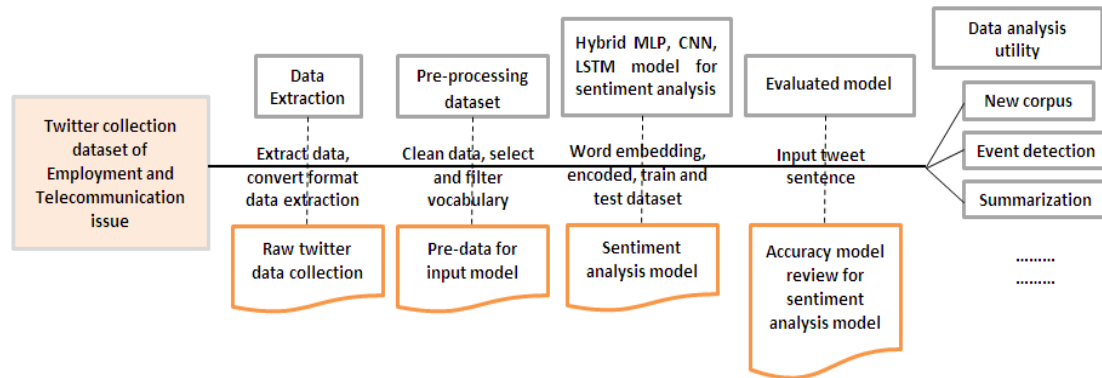


Figure 2. Analytics phases in sentiment evaluation

3.1. Data collection

The data is obtained from the Twitter collection provided and has been labeled by Ivosights. The dataset is classified into positive and negative sentiment. We use human annotators. They are native Bahasa Indonesia speakers because short text messages do not use standard Bahasa Indonesia. However, most are a mixture of Bahasa Indonesia, acronym, slang, English, Arabic, Javanese, or other local languages, emoticons, emojis, and other symbols. The annotator gives a sentiment label on each tweet message using their sense of how they would feel about the tweet's sentence. The most agreed classification by several annotators determines the final sentiment classification.

3.2. Data extraction

We take two different topics and characteristics from the Twitter dataset, which relate to employment and telecommunication operators' issues. There are 67,106 tweets in Bahasa Indonesia relate to Employment issues, which discuss issues ranging from welfare to social security to salary. There are 30,102 tweets with positive sentiments and 37,004 tweets with negative sentiments from January 2017 to November 2017. On the other hand, there are 81,159 tweets from January 2017 to May 2018 in Bahasa Indonesia about telecommunication operators' related issues, where 40,938 tweets of those are labeled as positive sentiments and 40,221 tweets others as negative sentiments. We select Bahasa Indonesia tweet messages only and delete tweet messages with local, foreign, or slang language and split the dataset into 80% of training data and 20% of test data.

3.3. Dataset pre-processing

Data pre-processing is carried out to prepare a clean dataset for further process. We clean the text messages by filtering unnecessary characters, such as punctuation characters, ASCII codes, tokens with non-alphabetic characters. The sentence is tokenized. Each token and word excluding Bahasa Indonesia's stop words [29] is called vocabulary, which is added to build a dictionary. We group tokens from the word dictionary by their sentiment into vocabulary files representing words from every record with similar sentiment polarity. There are 14,075 tokens with 27,736 dictionaries and 13,994 tokens with 36,264 dictionaries in the employment and telecommunication issue. The use of Indonesian stop words is beneficial for sorting out words that are compliant according to Indonesian language requirements, including five features of twitters (mention, hashtag, URL, discourse maker, and emoticons). The pre-process generates a data dictionary for each dataset. Figure 3 shows the top ten vocabularies in employment issues (a) and telecommunication issues (b).

3.4. Hybrid neural network model development for sentiment analysis

We build several models with CNN, MLP, LSTM and the hybrid of those three architectures. Firstly, we build a text representation where each word is represented as a matrix of real numbers using word embedding. A real number representation is required because the neural network model can only accept numerical input values. The word embedding technique is suitable for the neural network model because it keeps the order and interaction of the words within sentences and the probability functions of each word sequence. Thus, it can be more expressive than the classical model, such as bag-of-words, bigram, or trigram [30]. The next step is model building; we use CNN, MLP, LSTM, and the hybrid of those three models for document classification. The CNN uses a filter value for parallel work processing, kernel value, and activation function. The output of this stage is a two-dimensional vector that represents the extracted features.

The LSTM units consist of cells, input gates, output gates, and forgotten gates. The MLP uses a back-end model for feature interpretation. The output layer uses an activation function with values between 0 for negative sentiment and 1 for positive sentiment. After that, we look for the best training models which produce the maximum accuracy using gradient descents. A topological change of the architectural model is required to find the optimal configuration for minimizing the error. The last stage is storing the best-generated model for evaluation.

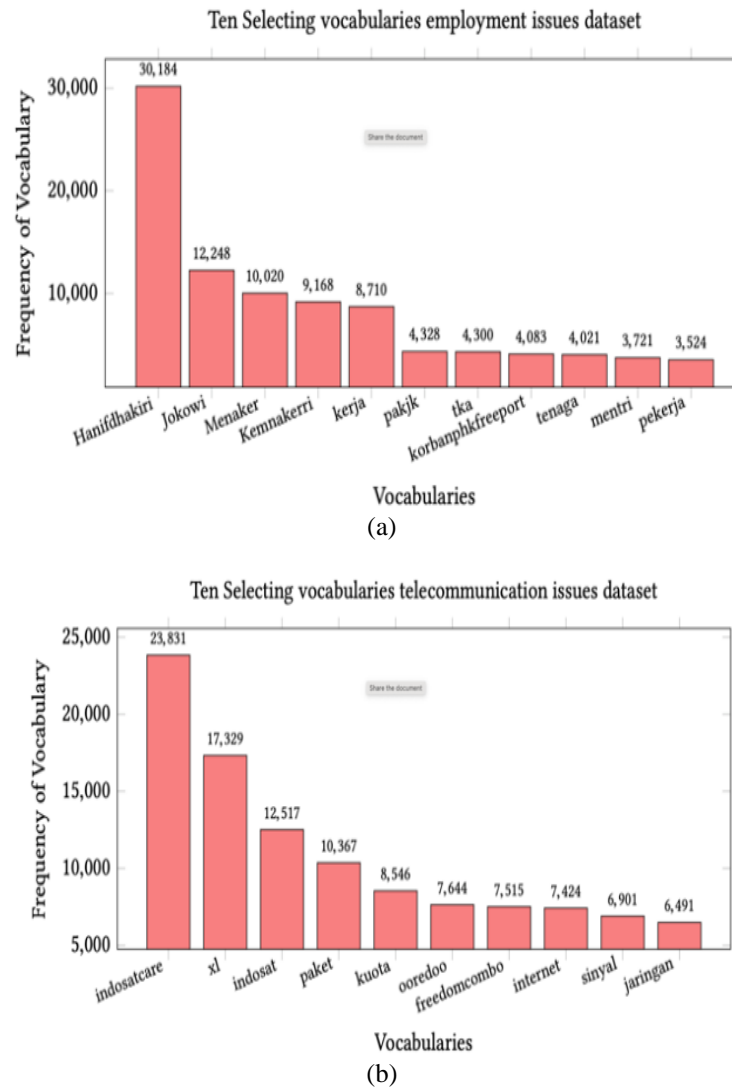


Figure 3. Top vocabularies in employment issues, (a) employment issues, (b) telecommunication issues

3.5. Model evaluation

We evaluate the generated model from the training process with the new testing dataset. Evaluation is carried out with a similar estimation function for evaluating training and testing datasets, where new data is encoded with a similar scheme of training data encoder.

4. RESULTS AND DISCUSSION

4.1. Experiment results

Twitter raw data collected offline on the employment and telecommunication issues are stored in a JSON file format. The data contains mentions, hashtags, URLs, discourse makers, and emoticons filtering, which could make an unexpected analysis result. Thus, the documents are extracted and converted into text files for the pre-processing stage. A clean dataset is modified to provide features according to the need for sentiment analysis to be analyzed. The pre-processing and cleaning of the document from retweet, double

post, and spam messages returns 89,373 messages on the telecommunication topic and 75,126 tweets on the employment issues topic. Table 1 shows the number of tweets that collected each sentiment feature and the number of cleaned tweets that are prepared after data extraction and pre-processing.

Table 1. Dataset detail

Dataset name	Properties	Number
Employment issues	Number of data	124,951
	Cleaned data	72,126
	Positive sentiment	30,102
	Negative sentiment	37,044
	Neutral sentiment	67,022
	Vocabularies	27,736
Telecommunication issues	Number of data	299,997
	Cleaned data	89,373
	Positive sentiment	40,938
	Negative sentiment	40,221
	Neutral sentiment	218,838
	Vocabularies	36,264

The training dataset is prepared by placing all the files that have been grouped into files with positive sentiments labeled with class 1 and files with negative sentiments labeled with class 0. The training is done by tokenization and word embedding to build vectors of real numbers as the word representation. The tokenization is representing documents as a sequence of consecutive integer numbers. The number amps a single token as a vector-specific representation of a real number. The placement of vector numbers is randomly assigned during the training process, and an embedding layer's API can be used to create class initiation for all document datasets. Since the inputs for the training process must have the same vector size, tokens in the documents used as the inputs should be sorted and padded with 0s if the number of tokens is less than the defined vector size.

Each word of each document is represented as a vector of real numbers using word embedding. The vectors are passed through various deep neural network architectures for document classification of sentiment analysis. The Convolutional Neural Network architecture consists of filtering, kernel size definition, and pooling layer for classification output simplification. The LSTM architecture will produce output after passing through forget, input, cell, and output gates. Furthermore, the MLP will classify after passing through input, several hidden, and output layers. The hybrid architectures are designed by appending one architecture to another, such as CNN and MLP, LSTM and MLP, and CNN and LSTM and MLP.

Figure 4 (a) shows the performance of implemented neural network architectures to get the minimum loss error value in the training stage. It shows that the hybrid (CNN+LSTM+MLP) outperforms the model algorithm with a more optimal level of error generated. The results of the hybrid model apply to both datasets that are provided as input models. Besides that, the LSTM only architecture is in the second-best. The result shows that LSTM could well enough to find context in sentiment analysis. The MLP shows the worst performance by generating a significant loss error because it used a feed-forward algorithm with a one-time learning process for its every layer. The training accuracy result is shown in Figure 4 (b) affirms the training loss error by showing again that the hybrid (CNN+LSTM+MLP) architecture has the best optimal performance for both trained datasets.

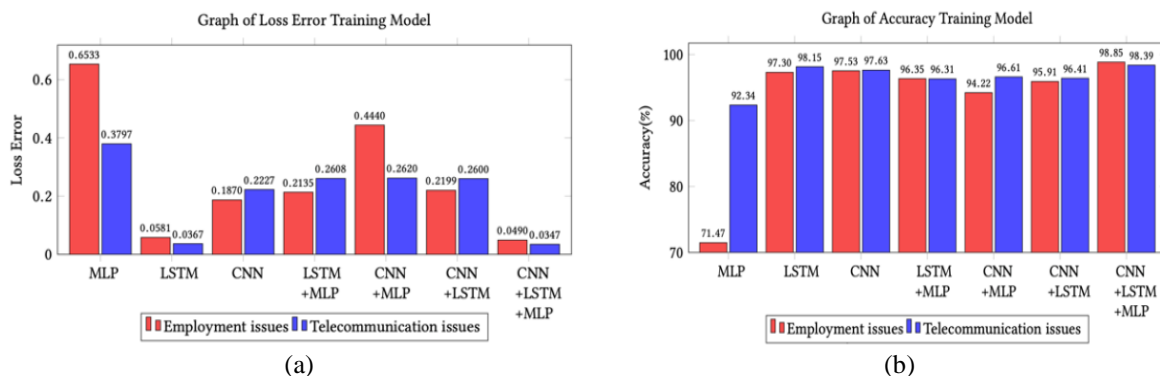


Figure 4. Training evaluation, (a) loss error, (b) accuracy

In the testing phase, we evaluate the loss error of the experimental results. Figure 5 (a) shows a significant difference between the loss error of evaluation on the employment issues and the telecommunication issues dataset. However, the main concern in this study is the performance difference of different architectures. It shows that the single CNN and hybrid (CNN+LSTM+MLP) architecture show optimal performance. It might be because CNN is pretty good for extracting features from the text in sentences. After all, at the testing phase, the trained model is tested against 20% of the dataset. The result shows that the hybrid architecture has an optimal minimum loss error compared to other single architectures' performance. Figure 5 (b) shows the accuracy performance of various architectures on test datasets. It shows that the hybrid (CNN+LSTM+MLP) architecture gives the best accuracy in both employment and telecommunication datasets.

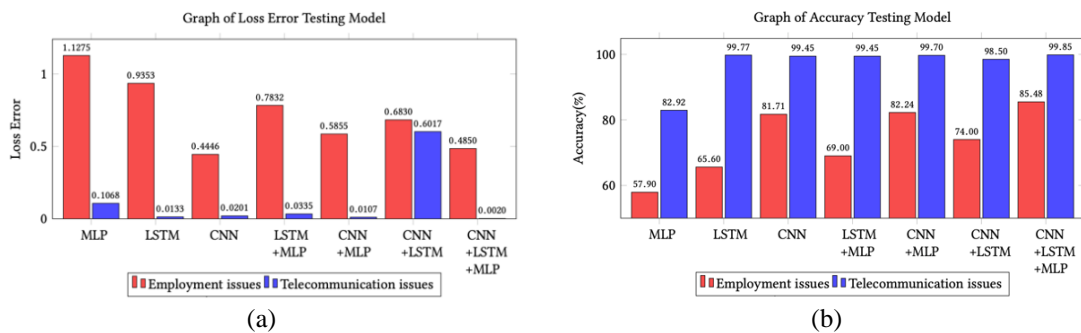


Figure 5. Testing evaluation, (a) loss error, (b) accuracy

4.2. Discussion

The previous section describes the results of our experiments in using seven different neural network architectures (MLP, CNN, LSTM, CNN+MLP, CNN+LSTM, LSTM+MLP, and CNN+LSTM+MLP) for sentiment analysis of short messages in Bahasa Indonesia on employment and telecommunication domains. The MLP architecture uses a simple one and two hidden layers. It takes input from the dataset according to the maximum length defined at the pre-processing stage. The best result of the learning process with MLP is obtained when using two hidden layers. It gives 0.6335 and 0.3797 on training loss error for employment and telecommunication issues datasets and 71.47% and 92.34% on training accuracy for both datasets, respectively. However, the loss error and the accuracy against both testing datasets are significantly worsened. They are to 1.1275 for testing loss error and 57.90% accuracy for employment issues and 0.1068 loss error and 85.92% accuracy on the telecommunication test dataset.

The CNN architecture is more reliable in extracting features from sentences that are defined in "negative" and "positive" sentiments directory. The loss error and accuracy in the training phase are better than those values on MLP with 0.1870 and 0.2227 on training loss error for employment and telecommunication issues datasets and 97.53% and 97.63% on training accuracy for each dataset. The values are not consistent on the testing phase where the loss error and accuracy for employment issues dataset is worsened to 0.4446 on testing loss error and 81.71% on testing accuracy. In contrast, those values for the telecommunication issues dataset becomes much better to 0.0201 on testing loss error and 99.45% on testing accuracy. The result might be because the telecommunication issues dataset has more records and richer vocabularies than the employment issues dataset.

The LSTM architecture is more context-oriented on the input sentence encoding. In this experiment, we add a useful dropout function to partially process the LSTM function before the output process to improve accuracy, minimize overfitting and set the value to a maximum of 0.5. The experiment shows that this architecture achieves better than MLP or CNN, especially on a larger dataset (such as the telecommunication issues dataset). The training phase gives optimal loss error (0.0581 and 0.0387) and good accuracy 97.30% and 98.15% for employment and telecommunication issues datasets. However, similar to the CNN, the testing loss error and accuracy worsen in the employment issues dataset to 0.9353 for testing loss error and 65.60% for testing accuracy, and better in telecommunication issues dataset to 0.0133 for testing loss error and 99.77% for testing accuracy.

The hybrid LSTM and MLP architecture is built by concatenation, where the result of LSTM and MLP processes are combined to obtain a single output. The final sentiment classification process is determined using the sigmoid activation function. The experiment shows that appending MLP does not help to improve the architecture performance, where the loss error and accuracy in the training and evaluation phase are somewhat worse than those in LSTM only architecture. The training loss error on hybrid LSTM and MLP architecture is 0.2135 and 0.2608 on the employment and telecommunication issues dataset, while

training accuracy is 96.35% and 96.31% for each dataset. Like other experiments, the analysis on the test dataset of employment issues is worse than the training one. Its loss error is 0.7832 and accuracy 69.00%. On the other hand, the analysis on the test dataset of telecommunication issues is better where its loss error and accuracy are 0.0335 and 99.45%, respectively.

The hybrid (CNN and MLP) architecture is created by combining MLP at the end of the feature extraction on CNN, before executing the flatten function that bridges the CNN process with the output process and giving a pool size of two in the max polling process. The results are quite good and better than hybrid LSTM and MLP in the training stage. The testing stage is slightly better but provides a comparable trial average value. Loss error values in the training stage are 0.4440 and 0.2620 for the employment and telecommunication issues dataset, while its accuracy is 94.22% and 96.61% for each dataset. The testing evaluation on the employment issues dataset is slightly better than single CNN or other dual hybrid architecture with a loss error of 0.5855 and 0.0107 for the employment and telecommunication issues dataset. Its accuracy is 82.24% and 99.70% for both datasets.

The hybrid CNN and LSTM use a combination of feature extraction with context orientation. At the end of the fixed output, it uses the sigmoid activation function for returning the sentiment classification. The performance values are quite good as the training loss error is 0.2199 and 0.2600 for the employment and telecommunication issues dataset, while its accuracy is 95.91% and 96.41% for each dataset. The architecture still has difficulties in employment issues' testing dataset while loss error is 0.6830, and the accuracy is 74.60%. In contrast, the telecommunication issues' testing dataset is evaluated with better performance as 0.6017 for its loss error and 98.50% for its accuracy.

The hybrid (CNN, LSTM, and MLP) architecture is built by concatenating CNN processes with LSTM processes and MLP processes. This integration makes a more complicated learning process, which results in a longer training computation time. However, that gives significant benefits for the architecture performance in sentiment classification. The training loss errors are 0.0490 and 0.0347 for the employment and telecommunication issues dataset. Their accuracies are 98.85% and 98.39% for each dataset. The architecture also shows its best performance for the testing dataset of the employment and the telecommunication issues, with loss error is 0.4850 and 0.0020. at the same time, their accuracies are 85.48% and 99.85%.

The performance of implemented architectures for sentiment classification can be listed as Table 2 for their performance on training datasets of employment and telecommunication issues and Table 3 for their respective testing datasets. We expect the optimal architecture can give the lowest loss error and the highest accuracy. On training and testing datasets, the hybrid (CNN+LSTM+MLP) shows its superiority over other architectures by giving the lowest loss error and highest accuracy on both training and testing datasets. Even though on the employment issues testing dataset, the CNN architecture gains the lowest loss error, but the hybrid (CNN+LSTM+MLP) is the second-best.

Table 2. Training performance for each model on each input domain datasets

Parameters	Employment dataset	Telecommunication dataset
Max. Error	MLP	MLP
Min. Error	CNN+LSTM+MLP	CNN+LSTM+MLP
Min. Accuracy	MLP	MLP
Max. Accuracy	CNN+LSTM+MLP	CNN+LSTM+MLP

Table 3. Testing performance for each model on each input domain datasets

Parameters	Employment dataset	Telecommunication dataset
Max. Error	MLP	MLP
Min. Error	CNN	CNN+LSTM+MLP
Min. Accuracy	MLP	MLP
Max. Accuracy	CNN+LSTM+MLP	CNN+LSTM+MLP

Figure 6 shows the model plot of the hybrid (CNN+LSTM+MLP) architecture. The input for each model contains vocabulary tokens with a limit of maximum length of 962 words. The CNN converts the input with word embedding by mapping each word into 100-dimensional vectors. The convolutional process extracts features from the input data with 32 parameters with four kernels. In order to reduce overfitting and optimization, we use a 0.5 dropout to minimize the computation process. The CNN is stopped on the flatten function because its result will be merged with results from other models. Similar to CNN side, the LSTM is initiated by applying word embedding on the input data. We use ten nodes neural network layer with the ReLu activation function for context-based processing to generate a ten-dimensional output vector using a 0.2 dropout optimization. The MLP side uses three hidden layers that consist of 20, 10, and 10 nodes with varying input dimensions. The flattened output CNN, LSTM, and MLP are merged by concatenation. We process the merged output using a feedforward neural network with a single hidden layer with ten nodes and predict the analysis output using a sigmoid function to give a positive and negative sentiment classification.

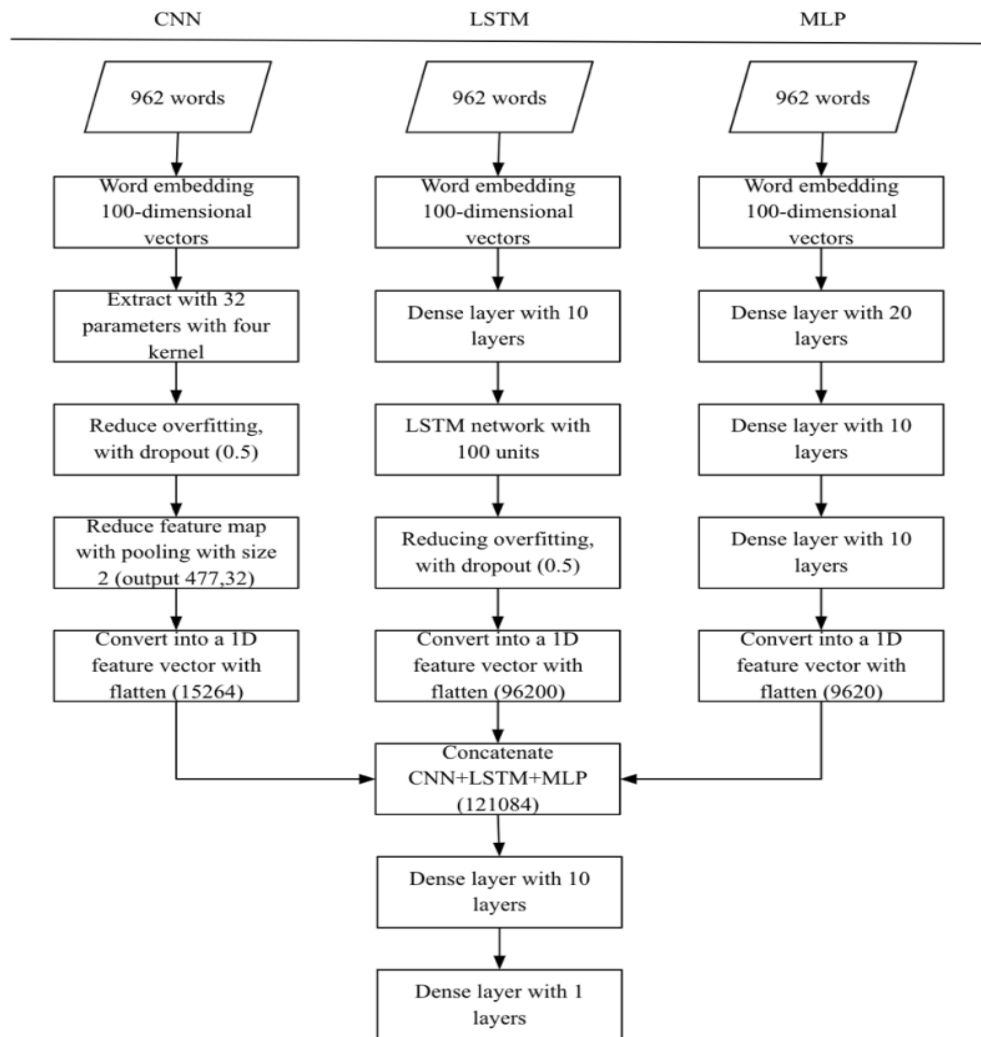


Figure 6. Hybrid model diagram

The result shows that combinations of CNN, LSTM, and MLP architectures give the best results and can be used in different domain datasets from its training domain. It is because the hybrid architecture gets the best of its building architectures models; the MLP that accommodates the use of the previous model to obtain classification output, the CNN layer that extracts the word feature vector from text sequences, and LSTM that repetitively selects or discards feature sequences based on their context. Those advantages are useful for different domain datasets, where one architecture might be better than others, but the combination might make it constantly similarly good. The experiments on sentiment analysis of short text in Bahasa Indonesia show that hybrid models can obtain better performance. The same architecture can be directly used in another domain-specific dataset.

5. CONCLUSION

Deep learning is a valuable method for developing sentiment analysis on short text messages, especially on limited resource language. It is due to its ability to infer and extract hidden information from a large number of data. CNN, LSTM, MLP architectures have shown their ability to solve the classification problems in natural language processing with a pretty good result. This research confirms that a hybrid architecture (CNN+LSTM+MLP) can give an even better result, and its architecture can directly be applied in different domain datasets. It is because the hybrid architecture can utilize the advantage of each building component.

ACKNOWLEDGEMENTS

The authors would like to thank PT. Ivonesia Solusi Data (Ivosight) which have provided us with a very valuable labeled dataset. All authors contributed equally as main contributors of this paper, read, and approved the final paper.

DECLARATION

All authors contributed equally as the main contributors of this paper. They read and approved the final paper.

REFERENCES

- [1] B. Liu, "Sentiment Analysis and Opinion Mining," *Morgan Claypool Publ.*, 2012.
- [2] A. Bandhakavi, N. Wiratunga, D. Padmanabhan, and S. Massie, "Lexicon based feature extraction for emotion text classification," *Pattern Recognit. Lett.*, vol. 93, no. December, pp. 133–142, 2017, doi: 10.1016/j.patrec.2016.12.009.
- [3] M. A. Ayu, S. S. Wijaya, and T. Mantoro, "An automatic lexicon generation for Indonesian news sentiment analysis: A case on governor elections in Indonesia," *Indones. J. Electr. Eng. Comput. Sci.*, vol. 16, no. 3, pp. 1555–1561, 2019, doi: 10.11591/ijeecs.v16.i3.pp1555-1561.
- [4] D. H. Abd, A. R. Abbas, and A. T. Sadiq, "Analyzing sentiment system to specify polarity by lexicon-based," *Bull. Electr. Eng. Informatics*, vol. 10, no. 1, pp. 283–289, 2021, doi: 10.11591/eei.v10i1.2471.
- [5] S. Sangam and S. Shinde, "Sentiment classification of social media reviews using an ensemble classifier," *Indones. J. Electr. Eng. Comput. Sci.*, vol. 16, no. 1, pp. 355–363, 2019, doi: 10.11591/ijeecs.v16.i1.pp355-363.
- [6] M. Fikri and R. Sarno, "A comparative study of sentiment analysis using SVM and Senti Word Net," *Indones. J. Electr. Eng. Comput. Sci.*, vol. 13, no. 3, pp. 902–909, 2019, doi: 10.11591/ijeecs.v13.i3.pp902-909.
- [7] K. Arun and A. Srinagesh, "Multi-lingual Twitter sentiment analysis using machine learning," *Int. J. Electr. Comput. Eng.*, vol. 10, no. 6, pp. 5992–6000, 2020, doi: 10.11591/ijece.v10i6.pp5992-6000.
- [8] C. V. D., "Hybrid approach: naive bayes and sentiment VADER for analyzing sentiment of mobile unboxing video comments," *Int. J. Electr. Comput. Eng.*, vol. 9, no. 5, p. 4452–4459, 2019, doi: 10.11591/ijece.v9i5.pp4452-4459.
- [9] A. Hasan, S. Moin, A. Karim, and S. Shamsirband, "Machine Learning-Based Sentiment Analysis for Twitter Accounts," *Math. Comput. Appl.*, vol. 23, no. 1, pp. 1–15, 2018, doi: 10.3390/mca23010011.
- [10] C. Wu, F. Wu, S. Wu, Z. Yuan, J. Liu, and Y. Huang, "Semi-supervised dimensional sentiment analysis with variational autoencoder," *Knowledge-Based Syst.*, vol. 165, pp. 30–39, 2019, doi: 10.1016/j.knosys.2018.11.018.
- [11] S. Rani and P. Kumar, "Deep Learning Based Sentiment Analysis Using Convolution Neural Network," *Arab. J. Sci. Eng.*, vol. 44, no. 4, pp. 3305–3314, 2019, doi: 10.1007/s13369-018-3500-z.
- [12] D. Stojanovski, G. Strezoski, G. Madjarov, I. Dimitrovski, and I. Chorbev, "Deep neural network architecture for sentiment analysis and emotion identification of Twitter messages," *Multimed. Tools Appl.*, vol. 77, no. 24, pp. 32213–32242, 2018, doi: 10.1007/s11042-018-6168-1.
- [13] F. Abid, M. Alam, M. Yasir, and C. Li, "Sentiment analysis through recurrent variants latterly on convolutional neural network of Twitter," *Futur. Gener. Comput. Syst.*, vol. 95, pp. 292–308, 2019, doi: 10.1016/j.future.2018.12.018.
- [14] Q. Pan and J. L. Gastwirth, "Issues in the Use of Survival Analysis to Estimate Damages in Equal Employment Cases," *Law, Probab. Risk*, vol. 8, no. 1, pp. 1–24, 2009, doi: 10.1093/lpr/mgp001.
- [15] Z. Jing-xiao, L. Xiao-hua and J. Wei-xing, "Solving the Employment Problems in China Construction Industry: Localization, IT Service, Minimum Wage Limit," *2010 International Conference on E-Business and E-Government*, 2010, pp. 642–645, doi: 10.1109/ICEE.2010.168.
- [16] M. Pivec and A. Maček, "Employment background influence on social media usage in the field of European project management and communication," *J. Bus. Res.*, vol. 94, no. August 2017, pp. 280–289, 2019, doi: 10.1016/j.jbusres.2018.03.021.
- [17] D. M. Boajă and F. C. Ciurlău, "Actual Issues Concerning Employment in Romania," *Procedia-Soc. Behav. Sci.*, vol. 62, pp. 301–307, 2012, doi: 10.1016/j.sbspro.2012.09.049.
- [18] B. Fan, H. Tian, L. Jiang, and A. V. Vasilakos, "A Social-Aware Virtual MAC Protocol for Energy-Efficient D2D Communications Underlying Heterogeneous Cellular Networks," *IEEE Trans. Veh. Technol.*, vol. 67, no. 9, pp. 8372–8385, 2018, doi: 10.1109/TVT.2018.2846811.
- [19] H. J. Choi and C. H. Park, "Emerging topic detection in twitter stream based on high utility pattern mining," *Expert Syst. Appl.*, vol. 115, pp. 27–36, 2019, doi: 10.1016/j.eswa.2018.07.051.
- [20] M. Congosto, P. Basanta-Val, and L. Sanchez-Fernandez, "T-Hoarder: A framework to process Twitter data streams," *J. Netw. Comput. Appl.*, vol. 83, pp. 28–39, 2017, doi: 10.1016/j.jnca.2017.01.029.
- [21] R. Wijayanti and A. Arisal, "Automatic Indonesian Sentiment Lexicon Curation with Sentiment Valence Tuning for Social Media Sentiment Analysis," *ACM Trans. Asian Low-Resource Lang. Inf. Process.*, vol. 20, no. 1, 2021, doi: 10.1145/3425632.
- [22] F. Koto and G. Y. Rahmaningtyas, "Inset lexicon: Evaluation of a word list for Indonesian sentiment analysis in microblogs," *Proc. 2017 Int. Conf. Asian Lang. Process. IALP 2017*, vol. 2018-Janua, pp. 391–394, 2018, doi: 10.1109/IALP.2017.8300625.

- [23] A. F. Wicaksono, E. Nio, and S.-H. Myaeng, "Unsupervised Approach for Sentiment Analysis on Indonesian Movie Reviews," in *CISAK 2013*, 2013.
- [24] R. Manurung, "Machine learning-based sentiment analysis of automatic Indonesian translations of English movie reviews," *Proc. Int. Conf. Adv. Comput. Intell. Its Appl. 2008 (ICACIA 2008)*, vol. 1, pp. 1–2, 2008.
- [25] T. A. Le, D. Moeljadi, Y. Miura, and T. Ohkuma, "Sentiment Analysis for Low Resource Languages : A Study on Informal Indonesian Tweets," *12th Work. Asian Lang. Resour.*, no. December, pp. 123-131, 2016.
- [26] W. C. F. Mariel, S. Mariyah, and S. Pramana, "Sentiment analysis: A comparison of deep learning neural network algorithm with SVM and naïve Bayes for Indonesian text," *J. Phys. Conf. Ser.*, vol. 971, no. 1, pp. 1–8, 2018, doi: 10.1088/1742-6596/971/1/012049.
- [27] Nurkholis, M. Zuwari Negara, G. Fajar Shidik, A. Zainul Fanani, Muljono, and E. Noersasongko, "Sentiment Analysis of Indonesian News Using Deep Learning (Case Study: TVKU Broadcast)," *Proc.-2018 Int. Semin. Appl. Technol. Inf. Commun. Creat. Technol. Hum. Life, iSemantic 2018*, pp. 261–265, 2018, doi: 10.1109/ISEMANTIC.2018.8549723.
- [28] S. J. Putra, I. Khalil, M. N. Gunawan, R. I. Amin, and T. Sutabri, "A hybrid model for social media sentiment analysis for Indonesian text," *ACM Int. Conf. Proceeding Ser.*, pp. 297–301, 2018, doi: 10.1145/3282373.3282850.
- [29] F. Z. Tala, "A Study of Stemming Effects on Information Retrieval in Bahasa Indonesia," M.S. thesis, Institute for Logic, Language and Computation, Universiteit van Amsterdam, 2003.
- [30] Y. Bengio, R. Ducharme, P. Vincent, and C. Jauvin, "A neural probabilistic language model," *J. Mach. Learn. Res.*, vol. 3, pp. 1137-1155, 2003.