JOURNAL OF SOFTWARE ENGINEERING & INTELLIGENT SYSTEMS
ISSN 2518-8739
31 December 2019, Volume 4, Issue 3, JSEIS, CAOMEI Copyright © 2016-2019
www.jseis.org

CONVOLUTIONAL NEURAL NETWORK FOR TEXT CLASSIFICATION ON TWITTER

AGUNG TRIAYUDI

Faculty of Information and Communication Technology
Universitas Nasional, Indonesia
Email: agungtriayudi@civitas.unas.ac.id

ABSTRACT

Natural Language Processing with a combination of Neural Network methods such as Convolutional Neural Network (CNN) that is included in the Deep Learning method and carries out a repetitive learning process to get the best representation of each word in the text. CNN Works by finding the pattern of a word among other words in the input matrix. The learning process in several convolution layers is carried out parallel and in sequence. Thus, each word is independent of other words around it. Twitter is a source of data that interests researchers to make research objects. However, the text in tweets contains many non-formal languages, abbreviations and everyday languages. Thus, it is more difficult to identify the information in it, when compared with the formal text. In this research, the Natural Language Processing method is implemented using the CNN algorithm to classify information related to the emergency-respond phase. This classification model was trained using two types of datasets, namely the crawling dataset of 1967 texts, and the dataset in the form of tweet texts from Twitter totalling 853 sentences and tested using 89 different text tweets. From the results of 3 iterations with 10 epoch training per iteration, an accuracy of 98% was obtained and a loss of 4% was obtained. Thus, it can be concluded that the algorithm functions optimally in identifying information.

OPEN ACCESS

Academic Editor: Hikmat Khan, COMSATS, Pakistan.
Published: 31 December, 2019
Funding: Authors have no funds.
Article Type: Research Article

Keywords: convolution neural network; twitter; deep learning; text classification; natural language processing;

1. INTRODUCTION

The development of data analysis and research using data mining algorithms, expert systems, neural networks are so popular, seen by the increasingly widespread intelligent systems in various systems today. Social media has received more attention now. Public and private opinions on various subjects expressed and disseminated continuously through various social media. Twitter is one of the social media that is gaining in popularity. Twitter offers organizations a quick and effective way to analyse customer perspectives on what is critical to success in a market place. Developing a program for sentiment analysis is an approach that will be used to compute measuring customer perceptions. in this research sentiment analysis design, extracting a large number of Prototyping tweets used in this development. The results classify customer perspectives through tweets into positive and negative, which are represented in pie charts and html pages. [1][2][3][4].

One use of social media in research is the analysis of tweets using Natural Language Processing techniques. For example, analysis of tweets for predictions of winners in football matches, sentiment analysis of legislative candidates in general elections, analysis of user trends toward certain products, analysis of tweets to identify natural disaster events, and others [5][6].

The text of the tweet is different from the formal text. Text tweets often use non-formal languages, synonymous words, abbreviations and colloquial languages. The information contained in it also has the potential to contain inaccurate information. This condition is a challenge in itself to identify valuable information on tweets and classify them. Some NLP algorithms that are commonly used for text classification are SVM (Support Vector Machine), Naïve Bayes, Entropy, Random Forest, and Novel Algorithm [7][8].

The results of the accuracy of these algorithms are very dependent on the results of the feature extraction, the features used, and the order of words in the sentence. Here, the Deep Learning algorithm works better because of its ability to relearn the learning results until the maximum accuracy and loss values are as low as possible. CNN (Convolutional Neural Network) algorithm is the development of ANN (Artificial Neural Network) algorithm. This algorithm performs feature learning by convoluting using several filter sizes per channel simultaneously. In image processing, each pixel unit inputted will be connected to the output layer [9][10]. Thus, this algorithm works very well without being affected by the position of objects in the image. In the study conducted, CNN gave

Corresponding Author: Agung Triayudi
Email Address: agungtriayudi@civitas.unas.ac.id

1.5% higher accuracy than the SVM and ANN algorithms [11]. In an emergency, increasing accuracy means the more valuable information obtained to determine the emergency actions that can be taken [12][13].

In this study, the author aims to examine the implementation of Natural Language Processing techniques using CNN algorithms and their performance in classifying tweet texts related to the emergency response phase, so that they can be used as an alternative source of information for disaster organizations. The amount of data used is 951 text tweets consisting of 4 labels, namely: relief, report, sentiment, and non-earthquake

2. RESEARCH METHOD

This study uses a quantitative paradigm by measuring the accuracy of the CNN (Convolutional Neural Network) algorithm to classify text in tweets. The training process and text classification on Twitter can be seen in Figure 1.

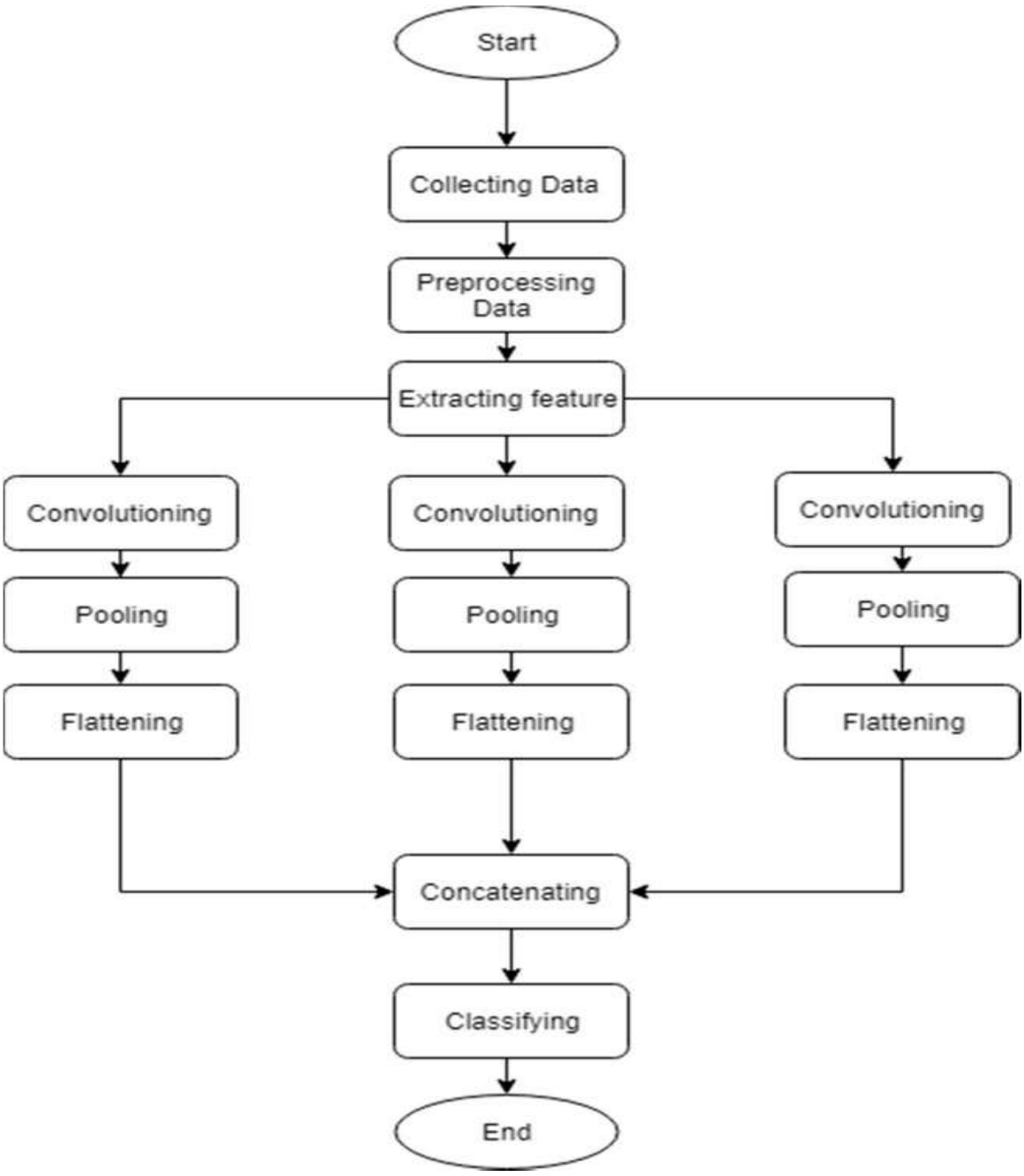

Figure. 1 Flowchart of research methods

2.1 Collecting data

Data collection is done by web scraping. Tweets obtained amounted to 951 with a span of years 2016-2018. These tweets are divided into training data and testing data with a ratio of 80%: 20%.

2.2 Pre-processing data

At this stage the pre-processing process involves removing symbols, numbers, ASCII strings, and punctuation, tokenization, case folding, stemming, stopword removal, normalization. Finally, new labelling is done in multiclass labelling as shown in Figure 2.

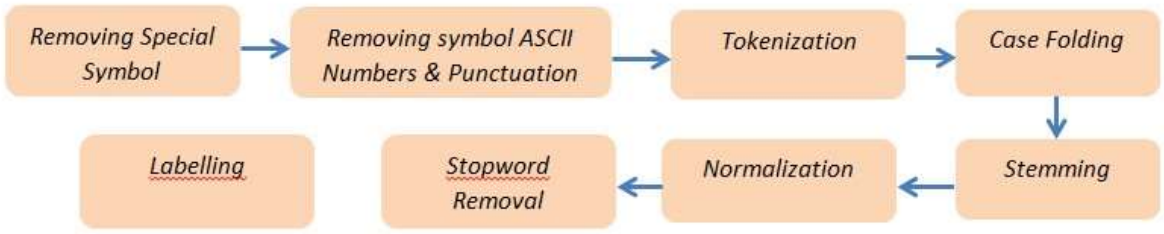

Figure. 2 Pre-processing

2.3 Remove special symbols

At this stage the cleansing process is done by removing the symbols and special characters in the tweet, namely "@", "RT", ASCII symbols, numbers and punctuation.

2.4 Tokenization case folding stemming

At this stage the text is converted into lowercase letters, separated per word and then stemmed or returned to the basic form per word.

2.5 Normalization

At this stage, each word is matched with Indonesian words in the literary literature. Word that is not found, then saved into a special file. The words in the file are then matched with the slang-word dictionary that contains the slang-word pair and the Indonesian language it should be. If found, the slang words or words that are not normal will be replaced with basic words in the Indonesian language that becomes his partner

2.6 Stopword removal

This process is used to eliminate words that are common but have no meaning (stopword). Stopword removal is done by using literary libraries.

2.7 Labelling

The labelling process is the process of grouping datasets into categories of reporting, relief or assistance, sentiment, non-earthquake.

2.8 Report

Reporting is an activity to report the current situation at the emergency response stage. Whether it's in the form of reports from the media, assessment and damage information.

2.9 Relief or assistance

Information in this category is information about assistance. Not limited to any institution or organization.

2.10Sentiment

This category contains tweets related to personal sentimental feelings or feelings about earthquake events

2.11Non-earthquake

Any information that is not related to an earthquake disaster, or is related to but does not support the emergency response process will be included in this category.

Text before and after pre-processing in shown in Figure 3 and Figure 4.

```
#PrayForPalu pic.twitter.com/irG2ZSn9gS
Donggala Kuat #PrayForDonggala
Turut berduka cita atas bencana gempa yang menimpa Donggala, Palu, dan sekitarnya. Tetap berkabar dan jaga diri,
.
#PrayForDonggala #PrayForPalu
Sejumlah gempa dengan kekuatan 5 sampai 7,7 SR secara beruntun mengguncang kawasan Donggala dan Palu.

#PrayForDonggala #PrayForPaluhttps://tirto.id/c3rl
Doa kita untuk Palu dan Donggala.

Tuhan bersama kalian. https://twitter.com/MilanFamiglia/status/1045670644866670593 …
Sejenak mari kita berdoa untuk saudara kita yang tengah dilanda gempa dan tsunami.

Semoga diberi keselamatan, kesabaran dan tetap tegar, Palu dan Donggala
#PrayforPalu #PrayforDonggala
```

Figure. 3 Text before pre-processing

```
siap \tnon-earthquake
mari ikan bantu pada saudara-saudara kita  paludonggala yang na tsunami lalui dompetkemanusiaanpaludonggala yayasan media gr
tak hadir  reuni bagi relaw hilmi fpi  palu dan donggala masih sibuk bantu saudara yang timpa musibah \tnon-earthquake
mari kita tonton bareng ramerame nanti malam  net tv pukul wawancara ekslusif dengan pak jokowi putar bencana alam  palu sig
baru terima rompiteam relaw kpu komunitas peduli umat mohon doa palu sigi donggala \trelief
```

Figure. 4 Text after pre processing

2.12Extracting Feature

The extracting feature process is the process of converting tweet text into a vector that can be consumed by the CNN algorithm. These vectors then form a matrix. For example, as shown in Figure 5.

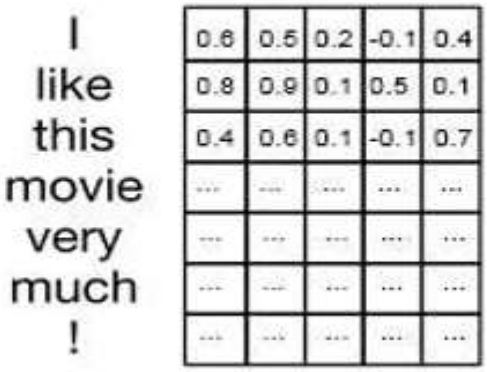

Figure. 5 Extracting features

2.13Convolutional Layer

Feature learning at this stage is done using several filter sizes, both on one channel or more. The value in the filter is multiplied by the input matrix, added with bias. The result, is included in feature maps. This study uses 3 filter sizes, namely 3, 4 and 5 with 1 channel. The size of the filter represents the number of words (n-grams). Each filter will convolute i.e. shifting from one side of the matrix to the other by a certain stride calculation. The results are calculated, then accommodated in the pooling layer to be convoluted to the next step. The size of the output layer can be calculated using the following formula:

$$(S - h + 1) * 1 \quad (1)$$

Where,
S = means sentence length,
h = region size

2.14Pooling Layer

The convolution value is stored in feature maps to be convoluted again, then stored in the pooling layer. In the pooling layer there is a feature reduction process that affects the speed of the training process. The pooling method used in this study is Max-pooling which takes the largest value on the pooling layer as shown in Figure 6.

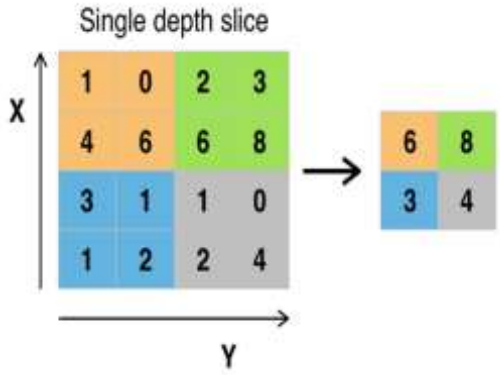

Figure. 6 Max-pooling

2.15Flattening Layer

Flattening changes, the output format of each pooling layer to a 1-dimensional matrix. The results of each flattening layer will be concatenated into a single feature vector.

2.16Classifying

The results of the concatenating or merging process will be processed at the Fully Connected Layer using the softmax function calculation as shown in Figure 7. The result is a probability value for each class

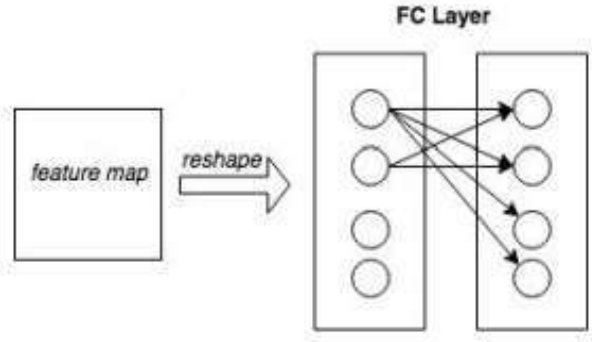

Figure 7. Fully connected layer

2.17The output

Output is the introduction of tweets into a particular class resulting from the feature learning process.

2.18Python programming language

Python is a programming language that is widely used in the field of data science. This language is equipped with many libraries that make it easy for its users. Applications in the results of this study use the Python programming language and the Django framework

3. RESULTS AND DISCUSSION

This research discusses the process of text classification on tweets into the Report, Relief, Sentiment, Victim and Non-Earthquake class.

3.1 Pre-processing and processing data Input

Pre-processing is done by using python libraries. Pre-processing is done by eliminating non-alphabet characters, eliminating stopword, stemming, labelling and tokenizing. Figure 8 and Table 1 show the pre-processing.

```
#preprocess per sentence
def clean_doc(doc):
    #split into tokens by white space
    tokens = doc.split()
    table  = str.maketrans('','',punctuation)
    tokens = [w.translate(table) for w in tokens]
    stop_words = set(stopwords.words('english'))
    tokens = [w for w in tokens if not w in stop_words]

    #filter out short tokens
    tokens = [word for word in tokens if len(word) > 1]
    tokens = ' '.join(tokens)
    return tokens
```

Figure. 8 Python scripts for pre-processing

Table. 1 Pre-processing techniques

Technique	Initial Data	Final Data
Non-alphabetic removing	atas nama #relawanprosan di kami turut berduka cita	atas nama relawanprosandi kami turut berduka cita
Stopword	atas nama relawan	nama relawan
Tokenizing	atas nama relawan	[‘atas’,’nama’,’relawan’]

3.2 Labelling

Figure 9 shows the labelling of the text.

```
suksesnya ini adlah sebagai pelipur lara yg sdg kena \tsentiment
moment silence untuk lombok palu dan donggala Semoga cepat pulih kembali \tsentiment
```

Figure. 9 Text already labelled

The following are examples of datasets that are ready to be processed as shown in Figure 10.

```
IkanIkan Mujahir di Danau Talaga Kab Donggala Sulawesi Tengah ditemukan mati
```

Figure. 10 Text ready to play

This research is doing feature extraction with Count Vectorizing technique. Where the unique corpus (D) and the number of data rows (N) will form the matrix D X N. First of all, each row of data (N) will be tokenized so that it forms a corpus list. Corpus on each data line must have the same amount. For this reason, padding is done, by adding as many dummy corpus as the number of corpus that is lacking in the sentence. For example, by tucking the word "<post>" in the sentence. Then, the text is tokenized and directed so that it forms as shown in Figure 11.

```
trainX = encode_text(tokenizer, trainLines, length)
type(trainX)
print(trainX)
```

```
[[ 896  897    2 ...    0    0    0]
 [ 902  903  904 ...    0    0    0]
 [ 184  234  154 ...    0    0    0]
 ...
 [  43   48   14 ...    0    0    0]
 [ 311 2172  311 ...    0    0    0]
 [ 105  656   53 ...    0    0    0]]
```

Figure. 11 Count vectorizing result

Count Vectorizing is one of the vectorization techniques, which converts text into numbers so that it can be recognized by machines and processed by Deep Learning algorithms such as CNN.

3.3 Training Process

Training on CNN is a learning feature process that includes Embedding Layer, Convolutional and Pooling Layer, while in the Fully Connected Layer, classification process is carried out. Embedding Layer functions to map the position of a vocabulary in a low dimensional vector representation. Furthermore, at the embedding layer, the convolution process is carried out using 3 filter sizes and 1 channel (convolution layer). The dropout functions to randomly delete values in the matrix, while the max-pooling technique in the pooling layer selects the most prominent value in the training data. The results of this training will be flattened and then combined for later classification on the Fully Connected Layer.

Thus, the learning process in this algorithm does not depend on the position of words in the sentence. Figure 12-14 show the accuracy.

```
Epoch 00004: val_loss improved from 0.35178 to 0.18023, saving model to model-3.h5
Epoch 5/10
760/760 [==============================] - 182s 239ms/step - loss: 0.1490 - acc: 0.9461 -

Epoch 00005: val_loss improved from 0.18023 to 0.15092, saving model to model-3.h5
Epoch 6/10
760/760 [==============================] - 181s 238ms/step - loss: 0.0994 - acc: 0.9750 -

Epoch 00006: val_loss improved from 0.15092 to 0.14828, saving model to model-3.h5
Epoch 7/10
760/760 [==============================] - 181s 238ms/step - loss: 0.0703 - acc: 0.9789 -

Epoch 00007: val_loss did not improve from 0.14828
Epoch 8/10
760/760 [==============================] - 180s 236ms/step - loss: 0.0645 - acc: 0.9829 -
```

Figure. 12 Accuracy and loss value per epoch

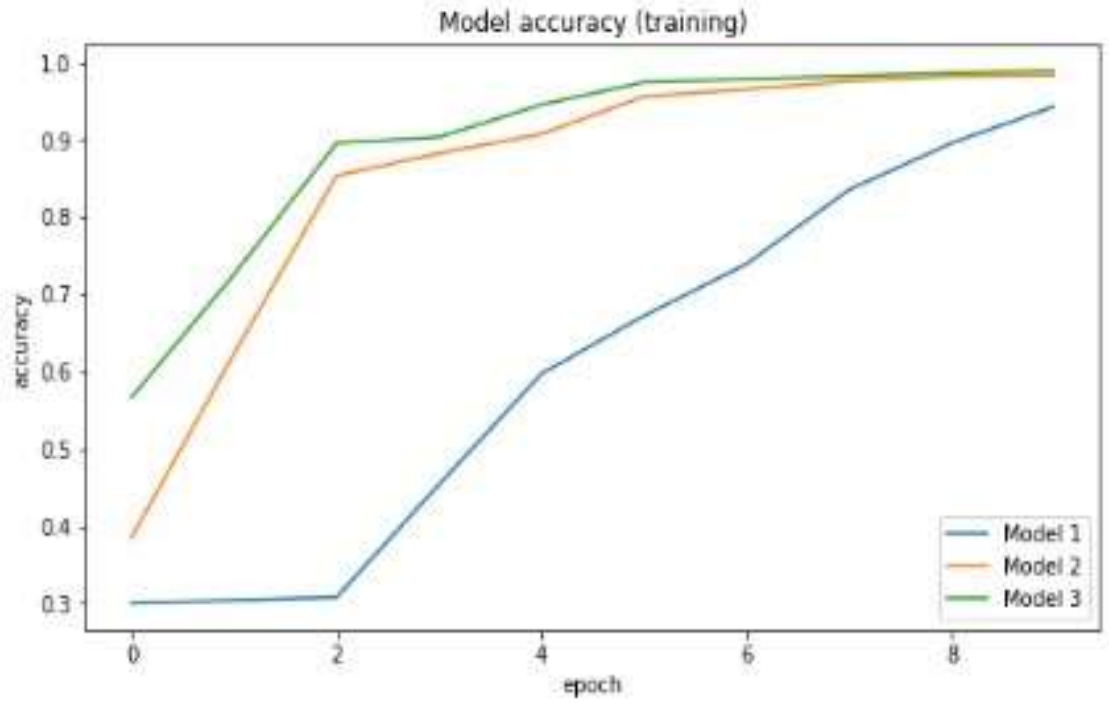

Figure. 13 Accuracy graph obtained per iteration (model)

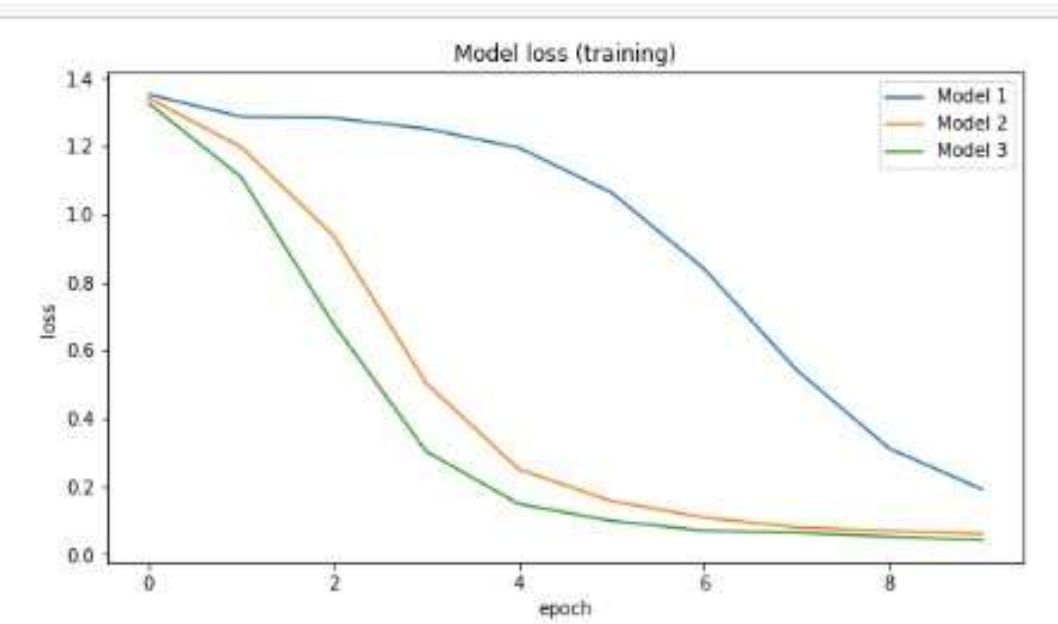

Figure. 14 Gambar nilai loss diperoleh per iterasi (model)

In the training process, the algorithm performs the learning process repeatedly. Like Deep Learning, CNN updates the weight of each epoch. Thus, at each epoch, accuracy tends to increase while loss decreases. The training process is carried out by 10 epochs with 3 iterations getting an accuracy of 98% with a loss of 4%.

3.4 Testing

Testing is done by confusion matrix technique, namely by comparing the predicted results of the classifier model with the actual label. The accuracy value obtained is 98% as shown in Table 2.

Table. 2 Testing accuracy/loss

Accuracy	Loss
98%	4%

4. CONCLUSION

The implementation of CNN for the classification of tweet texts that are few in number, can produce a pretty good accuracy of 98% with a loss of 4% only. CNN algorithm turns out to be different from the Deep Learning algorithm, where this algorithm does not depend on large amounts of data. For subsequent studies, accuracy may be compared to the results of training using certain techniques, such as drop outs, transfer learning, sub sampling, augmentation, and other techniques. In addition, the classification of information in the tweet text related to the emergency response phase can also be done by combining text and emoticons.

ACKNOWLEDGEMENT

This research is an outcome of the internal competitive grant scheme of the Universitas Nasional.

REFERENCES

1. Sarlan A, Nadam C, Basri S. *Twitter sentiment analysis*. InProceedings of the 6th International Conference on Information Technology and Multimedia 2014 Nov 18 (pp. 212-216). IEEE.
2. Prameswari EA, Triayudi A, Sholihati ID. *Web-based E-diagnostic for Digestive System Disorders in Humans using the Demster Shafer Method*. International Journal of Computer Applications.2019 ;975:8887.
3. Triayudi A, Fitri I. *A new agglomerative hierarchical clustering to model student activity in online learning*. Telkomnika. 2019 Jun 1;17(3):1226-35.
4. Triayudi A, Fitri I. *Comparison of parameter-free agglomerative hierarchical clustering methods*. ICIC Express Letters. 2018;12(10):973-80.
5. Anil B, Singh AK, Singh AK, Thota S. *Sentiment analysis for Product Reviews*. International Journal of Advanced Research in Computer Science. 2018 May 1;9(Special Issue 3):23.
6. Rathan M, Deepthi RN, Anupriya S, Vishnu V. *Football Match Outcome Prediction Using Sentiment Analysis Of Twitter Data*. International Journal of Advanced Research in Computer Science. 2018 May 1;9(Special Issue 3):78.

7. Carley KM, Malik MM, Kowalchuck M, Pfeffer J, Landwehr P. Twitter usage in Indonesia. Available at SSRN 2720332. 2015 Dec 21.
8. Yin J, Karimi S, Lampert A, Cameron M, Robinson B, Power R. *Using social media to enhance emergency situation awareness*. InTwenty-fourth international joint conference on artificial intelligence 2015 Jun 27.
9. De Priester L. *An approach to the profile of disaster risk of Indonesia*. Emergency and Disaster Reports. 2016;3(2):1-66.
10. Beigi G, Hu X, Maciejewski R, Liu H. *An overview of sentiment analysis in social media and its applications in disaster relief*. InSentiment analysis and ontology engineering 2016 (pp. 313-340). Springer, Cham.
11. Rudra K, Ghosh S, Ganguly N, Goyal P, Ghosh S. *Extracting situational information from microblogs during disaster events: a classification-summarization approach*. InProceedings of the 24th ACM International on Conference on Information and Knowledge Management 2015 Oct 17 (pp. 583-592). ACM.
12. Matuszka T, Vinceller Z, Laki S. *On a keyword-lifecycle model for real-time event detection in social network data*. In2013 IEEE 4th International Conference on Cognitive Infocommunications (CogInfoCom) 2013 Dec 2 (pp. 453-458). IEEE.
13. Stowe K, Paul MJ, Palmer M, Palen L, Anderson K. *Identifying and categorizing disaster-related tweets*. InProceedings of The Fourth International Workshop on Natural Language Processing for Social Media 2016 Nov (pp. 1-6).
14. Caragea C, Silvescu A, Tapia AH. *Identifying informative messages in disaster events using convolutional neural networks*. InInternational Conference on Information Systems for Crisis Response and Management 2016 May (pp. 137-147).
15. Costache M, Liénou M, Datcu M. *On bayesian inference, maximum entropy and support vector machines methods*. InAip conference proceedings 2006 Nov 29 (Vol. 872, No. 1, pp. 43-51). AIP.

AUTHORS PROFILE