TWEET CLASSIFICATION USING DEEP LEARNING ARCHITECTURE FOR CONCERT EVENT DETECTION

Adenuar Purnomo[1], Ahmad Afiif Naufal[2], Ery Permana Yudha[3], Agus Zainal Arifin[4]

[1][2][3][4]Departemen Informatika, Institut Teknologi Sepuluh November, Surabaya, 60111 Indonesia

E-mail: adenuar14@mhs.if.its.ac.id[1], afiifnaufal30@gmail.com[2], erypermana94@gmail.com[3], agusza@cs.its.ac.id[4]

Abstract

Twitter social media is used by millions of users to share stories about their lives. There are millions of tweets sent by Twitter users in a short amount of time. These tweets can contain information about an incident, complaints from Twitter users, and others. Finding information about events from existing tweets requires great effort. Therefore, this study proposed a system that can detect events based on tweets using the CNN-LSTM architecture. Based on the classification testing obtained precision results of 70.97%, and recall amounted to 63.76%. The results obtained are good enough as a first step to detect events on Twitter.

Keywords: *Deep Learning, Word Embedding, Information Extraction, Event detection, Social Media*

Abstrak

Media sosial Twitter digunakan jutaan pengguna untuk berbagi cerita tentang kehidupan mereka. Ada jutaan tweet yang dikirimkan oleh pengguna Twitter dalam waktu yang singkat. Tweet-tweet tersebut dapat berisikan sebuah informasi tentang sebuah acara konser, keluh-kesah dari pengguna Twitter, dan lain-lain. Mencari acara konser dari tweet-tweet yang ada membutuhkan usaha yang besar. Oleh sebab itu, penelitian ini membuat sebuah sistem yang dapat mendeteksi acara konser berdasarkan data tweet menggunakan arsitektur CNN-LSTM. Berdasarkan pengujian klasifikasi didapatkan hasil presisi sebesar 70,97% , dan recall sebesar 63,76%. Hasil yang didapat sudah cukup bagus sebagai langkah awal untuk mendeteksi acara konser di Twitter

Kata Kunci: *Deep Learning, Word Embedding, Ekstraksi Informasi, Deteksi Acara Konser, Media Sosial*

1. Introduction

Nowadays, social media like Twitter becomes one of the most important means of communication due to the impact of smartphone development. Twitter in a day produces a new status (or tweets) of more than 500 million. Twitter is a social media platform that millions of users use to share stories about their lives. Because almost everyone has a smartphone, anyone can post a message when they watch or are involved in an event. Most often, this tweet is about events that occur around users. Media companies need time to report on these events. The time needed is still slow to study, investigate, and report on the event. Instead, users' interpretations of the concert event might be uploaded to Twitter in real-time. Their Twitter followers can learn about a concert before the news company can broadcast the information.

By overcoming this problem, all hidden information can be extracted. The concert program can be identified and presented directly by the first witness. In fact, only 1% of Twitter's freely available public data includes around 95% of all events reported in traditional event services [1]. In addition, Twitter data also contains several other events that are too small, too local, or too specific to be new events, as well as events that have a short period of time. As a result, it is very important to have a system that can identify, and separate concert events from spam floods and daily events reported on Twitter in a timely manner. The system can be useful for journalists who increasingly adopt social media as a professional tool [2],[3]. The government agencies and other organizations, such as human rights organizations, will be interested in early detection of concert events that occur around the world [4] and on average, Most Twitter users will be happy to have a system that allows them to keep up with the latest concert shows.

The increasing use of social media in the past decade has necessitated the development of new

methods beyond traditional text analysis methods. Despite attention to the detection of events on Twitter that are targeted, this is still an unsolved task because of the tweet characteristics mentioned above. The approach is based on entities, hashtags, and paraphrases. Different from this approach, the method used in this research looks at all tweets to achieve good event detection, which means that it does not depend on existing entities, thus making the approach more general and flexible.

Research related to detection systems on Twitter has been done. Inuwa-Dutse et al. [5] proposed detection of spam-posting accounts using user data features. Balakrishnana et al. [6] suggested use of Big Five and Dark Triad features for cyberbullying detection on Twitter. Other approaches have focused on detecting events in realtime. Kunneman et al. [7] for instance, the goal was to detect events through a machine-learning approach based on term pivoting. Repp et al. [8] extract news events using a deep-learning model. Ajao et al. [9] identify fake news with Hybrid CNN and RNN Models. Dabiri et al. [10] present traffic event detection model using deep learning architectures. Finally, Hasan et al. [11] investigated real-time event detection using the TwitterNews+ Framework. Earlier approaches are still concerned in general terms of the event detection. Thus, we proposed a specific event to detect concert events.

One of those past event detection researches[10] proposed a comparison between CNN, LSTM, and CNN-LSTM in event detection and got a good result. From that paper, the accuracy of the 3 methods is more than 90%. Masci et al [12] use CNN as a sentence encoder in a variety of natural language applications and achieved excellent results. In that research, CNN was used as an encoder on the encoder-decoder framework. CNN convoluted the input sentence and then produced a vector that would be inserted into the decoder. Many decoders are used in previous research, one of them is LSTM. In Greff et al research [13], LSTM is very good to use because it can reduce the number of parameters and computational costs without significantly reducing performance.

This study proposes a tweet classification system using deep learning to detect concert events on Twitter data. Classification of tweets is based on information related to the concert. Solving this problem will be a fast way to notify users about concerts that are happening in their area. To achieve this goal, the system learns how to select features from tweets and how the system must represent them when using deep learning to filter concerts tweets from non-concerts tweets. The research began by retrieving Indonesian tweets data using the Twitter API. Then each tweet data is done pre-process to get the tweet that contains important and general information. The clean data is then extracted using word embedding features and then used as input to be classified using the Deep Learning method with the CNN-LSTM architecture.

2. Preliminary Studies

There is no official concept in general for the definition of "event". So it is difficult to have the same understanding of event detection in research. Comparison of results for various event detection approaches is very difficult. Especially the level of detail of the program differs from one study to another. Some studies consider certain events to be a single event (eg earthquake), while others can break up the same event into several (eg earthquake and tsunami). So the researchers put some limits on the proposed system and make comparisons between systems difficult. To overcome this problem a corpus [14] detected a large event on Twitter data and proposed a more general event definition as shown on Definition 2.1.

Definition 2.1 (Events [5])

1. An event is a significant thing that happens at a certain time and place.
2. Something significant if it can be discussed in the media. For example, you can read an event article or watch an event report about it.

Those definition is very appropriate and will be used for event detection which can be considered a concert event. However, please note, even though a concert event must occur at a certain time and place, that does not mean that tweets that refer to an event will explicitly mention the time and place. For example, a tweet might say: "Jeno NCT Dream shows her abs". This Tweet doesn't mention the time or place, but if it refers to the actual episode, the episode is undoubtedly a concert program. In this study, the aim is to detect concert events for Twitter data. Therefore, the general assumption is that every event mentioned has concert relevance at the time the tweet that refers to it is posted. In other words, an event that has happened, or that will happen in the future, can still produce a concert event on Twitter today. Next, we distinguish between concert tweets and concert as shown on Definition 2.2.

Definition 2.2 (Tweet Concerts and Concert Events)

Concert tweets are tweets that refer to a particular concert, or directly related to the event as

described in Definition 2.1. A concert event is a group of concert tweets that are naturally connected to each other due to temporal and semantic similarities.

A further explanation of Definition 2.2 is a group of concert tweets that discuss the same topic at the same time. So any tweets that are not related to the concert are not considered relevant. In other words, research focuses on concerts. Note that although a concert program can refer to any type of concert, the type of concert cannot be determined. Furthermore, the time of occurrence for real world events that produce events on Twitter is not important. This is because a sudden increase in interest can attract Twitter to show that the event is rated as a concert at the time and it can be concluded concerts. However, we can assume that Twitter users are more vulnerable to tweets about the latest shows than old shows.

3. Methods

In this study, the tweet classification process is based on four main stages, namely 1) Pre-processing, 2) Word Embedding, 3) Create Dictionary and 4) Classification. Broadly speaking, the stages in the tweet classification process can be seen in Figure 1.
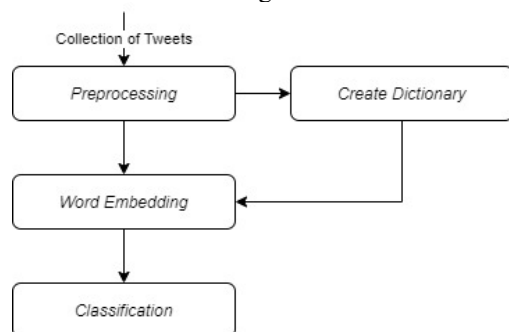

Figure 1. Stages in Tweet Classification

Pre-Process

The pre-process or preprocessing stage is the initial stage carried out in the tweet classification process. The preprocessing process consists of four main stages: the process of equating cases, tokenization, removing unnecessary components, and stemming. Case equalization is done by converting the whole text into a standard form (lowercase). Tokenization is the process of changing sentences into tokens or words. Removing unnecessary components is done by removing all web URLs, special characters, punctuation, and stopwords. The last stage of

---

1 https://kbbi.web.id/

preprocessing is stemming. Stemming is a process change every word into a basic word.Stemming process is done by matching the tokenization results with a dictionary. The dictionary contains the basic words of all words resulting from tokenization. The basic word used is referring to *Kamus Besar Bahasa Indonesia* (KBBI) online version[1]. The process of making a dictionary will be explained in the next section. Tweets that have been preprocessed are ready to be used in the next tweet classification process.

Create Dictionary

Collection of words from tokenization results contain many words that are still not standard. In addition, they also contain slang, abbreviations and foreign languages. So it needs to manually make a dictionary that contains both non-basic words and basic words, so it can replace non-basic words to basic words. The first process to make the dictionary is to filter the words so that there are no words with the same meaning. Then search foreign words, slang, and abbreviations in those filtered words. Of all the words that have been found, the words then changed into its basic words. This collection of non-basic words and basic words is compiled into a dictionary used in the stemming process.

Word Embedding

After the preprocessing stage, the tweet data that is clean from the noise is obtained. The data is used as input in the feature extraction process. The extraction feature at this stage uses word embedding by using a library called Gensim. This study does not use TF-IDF because it does not match the method used because the TF-IDF output is sparse vector instead of vector of real numbers. The results of the feature extraction process using word embedding produce a vector with a length of as much as the number of vocabulary words followed by their respective weights. Each vocabulary can display whatever vocabulary is closest to it and its weight. The vector will be used as input in the classification process.

Classification

In this study using Deep Learning as its classifier. The architecture used is the CNN-LSTM architecture. The CNN-LSTM architecture used can be seen in Figure 2. While the output produced by each layer in this architecture can be seen in Table 1.

TABLE 1
CNN-LSTM ARCHITECHTURE OUTPUT

Layer	Output
Embedding	500x500
Convolutional Layer	499x200
Max Pooling	249x200
Convolutional Layer	248x100
Max Pooling	124x100
Convolutional Layer	124x50
LSTM Layer	256
Fully Connected Layer	128
Dropout Layer	128
Fully Connected Layer	64
Dropout Layer	64
Fully Connected Layer	32
Dropout Layer	32
Fully Connected Layer	16
Dropout Layer	16
Classification Layer	2

4. IMPLEMENTATION

To evaluate the proposed method, an automatic evaluation is carried out based on the existing dataset and the appropriate ground truth. In addition, to prove the feasibility of our approach and the validity of automated evaluations, we conduct user-based evaluations. First, we describe the dataset that we use. Then, we explain the steps we applied for evaluation.

Datasets

To evaluate this research, we collected Indonesian tweets on November 9, 2019 using the Twitter API. The reason why tweets are taken on that date is because that day is the weekend, the day on which concerts generally occur. Daily Twitter data is enough to represent this research because every day many people make tweets.

The tweet collection process is carried out through several stages. First, request an access token and API key. At this stage Twitter API users are asked to fill in several requirements that must be filled.

Second, crawling data using a program script. Because our research is related to concert detection, it uses keywords that represent Indonesian tweets with concert topics such as 'concerts', 'tickets', and 'performances'. At this stage the program script is set so that it can connect to the Twitter API and can crawl tweets with those keywords.

Third, after getting tweets with specific keywords then tweets are saved into an excel file. Some examples of tweets obtained can be seen in Table 2.

From the data collection process, we get 2000 tweets which 971 tweets labeled as class 0 (non-concert tweet), and 1029 tweets labeled as class 1 (concert tweet). Those tweets are manually labelled by 3 annotators. 3 of the writers are the annotators, where each annotator gives the label to each tweet, then the annotators select the class by using majority vote from the label.

If on that date there is a big event that is an event with a large number of tweets or as well as a rapidly developing event then it is suspected that there is an important event. We hoped that our system could provide concert information on that date.

Evaluation Steps

Evaluation for event detection on microblog data, such as Twitter, is generally challenging, because finding good steps for quantitative and qualitative performance that enables comparative studies is not an easy task. To overcome the lack of general evaluation methods, event detection techniques can only be evaluated "on their own", e.g. with respect to setting different parameters [15].

The fact that almost no one published the dataset or source code makes deep comparison with other approaches difficult. Because our focus is evaluating the ability to detect concerts, recall, and precision are an appropriate evaluation measure.

In this paper the evaluation steps can be described in Figure 2. We have some scenarios to evaluate concert detection.

First scenario is the splitting of training data and testing data using CNN-LSTM architecture as described in Figure 3, then we try to compare the 3 methods used in this paper by using the best training-testing split. Finally, the fully connected layer size of the best method is tuned to find the best architecture.

5. Result and Discussion

Hardware and Software Specifications

In this detection study using hardware and software devices. Some hardware devices used in this study are shown in Table 3, and the software used is shown in Table 4.

Result

Table 5 shows a comparison of the amount of training data and testing data used for the training dataset in the CNN-LSTM model. Table 5 also shows the accuracy, precision, and recall of the test data. We can see that the best accuracy is when the train data is 80% and the test data is 20%. After that, we test the best method with default parameters and best training-test split.

Table 6 shows that CNN-LSTM gives the best accuracy when compared to other 2 methods. Since CNN-LSTM is giving the best result from the previous scenario, hyperparameter tuning is done with 3 scenarios that can be seen in table 7. We can see that the fully connected layer size of 128,64, and 32 gives the best accuracy.

The confusion Matrix of the best train test data split is shown in Table 8. There are still many misclassifications in the confusion matrix especially concert classified as non-concert.

In table 9, we can see some tweet classification results. some misclassification like tweet number 4 happens because of some error in the stemming process.

The error can happen when the stemming process could not replace non-basic words to basic words. Also there are many tweets that explicitly say that there is a concert but in reality that's not what the tweet meant.

6. Conclusion

The test results show that the tweets classification using the CNN-LSTM architecture can detect concert events. Based on the test that has been done, the classification of tweets using a dictionary with CNN + LSTM as a classifier has an accuracy value of 66%, a precision of 70.97%, and a recall of 63.76%. This research has not been able to handle words that use informal language but have the same meaning (synonym). Future studies are expected to develop tweets classifications by

paying attention to each unofficial word and word synonym.

TABLE 2
TWEET EXAMPLE

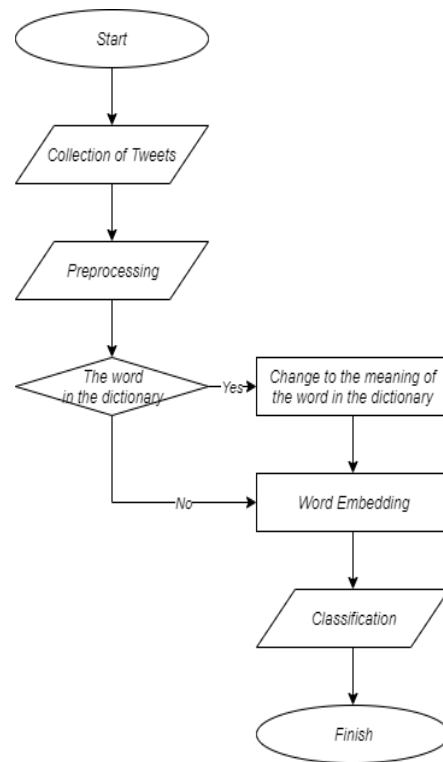| Number	Tweet	Class
1	[askmf] nonton konser kpop outfit nya begini yay/nay? https://t.co/9wy1oeh0dr	0
2	seru banget malmingku nonton konser day6, lightsticknya lucu banget lagi ðŸ—¤ #day6 #sweet_chaos https://t.co/vn0pkjaybp	1

Figure 2. Tweet Classification Testing Phase

TABLE 3
SPECIFICATION HARDWARE

Name	Descriptions
CPU	Intel Core i9-9900K CPU @ 3.60GHz
GPU	NVIDIA GeForce RTX 2070 8GB
RAM	64 GB
HDD	1 TB
SSD	250 GB

TABLE 4
SPECIFICATION SOFTWARE

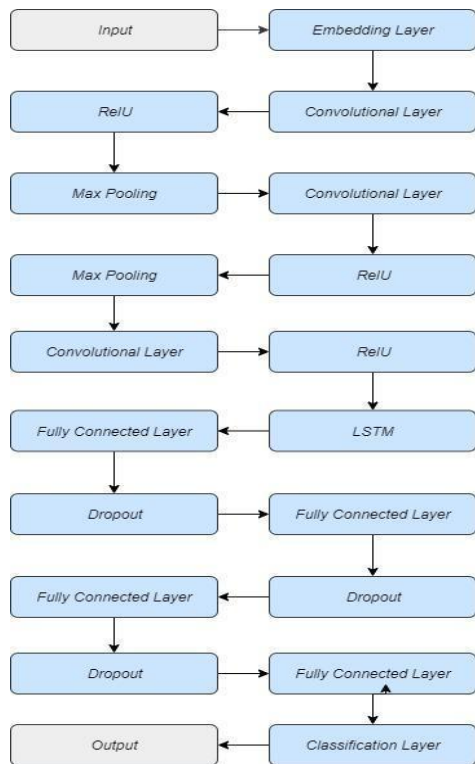| Name	Descriptions
Operating System	Windows 10 Pro 64-Bit
Anaconda Navigator	Version 1.9.7
Spyder	Version 3.3.6
Python	Version 3.6.0

Figure 3. CNN-LSTM architecture

TABLE 5
SPLITTING CLASSIFICATION RESULT USING CNN-LSTM

Train Data (%)	Test Data (%)	Accuracy (%)	Precision (%)	Recall (%)
20	80	46.87	0	0
33	66	60.15	73.63	41.34
40	60	62.25	67.43	58.39
50	50	60.4	65.46	57.01
60	40	61.75	68.69	52.68
66	33	64.84	68.94	62.71
80	20	66	70.97	63.76

TABLE 6
CLASSIFICATION RESULT WITH DIFFERENT METHOD USING
80% TRAINING AND 20% TESTING DATA

Method	Accuracy (%)	Precision (%)	Recall (%)
CNN	56	60.52	56
LSTM	65.25	42.57	65.25
CNN-LSTM	**66**	**70.97**	**63.76**

TABLE 7
FULLY CONNECTED LAYER SIZE HYPERPARAMETER
TUNING CLASSIFICATION RESULT

Fully Connected Layer Size	Accuracy (%)	Precision (%)	Recall (%)
128,64,32	**66**	**70.97**	**63.76**
64, 32, 16	60.5	62.28	60.5
32, 16, 8	63.25	62.24	63.25

TABLE 8
CONFUSION MATRIX

	Concert	Non-Concert
Concert	139	79
Non-Concert	57	125

TABLE 9
TWEET CLASSIFICATION RESULT

No	ground truth	predicted class	tweet
1	0	0	@duwarrrilopyu siapo klo itu mah, doain yaa moga taun dpn ikin konser d indo dan q bisa nnton lgi wkwkðŸ˜„ðŸ™
2	0	1	sebong konser in jkt aku konser di kondangan sepupu hiya hiya
3	1	1	@shininginjun apaan sih, berhentiin konser tuh maksudnya nge jeda dulu buduhhhh, diobatin dulu baru dilanjut lagi, ini tuh makanya kalo apa apa tuh positive thinking dulu gitu loh budayakan gitu napa sihhh cara berpikirnyaðŸ˜©ðŸ˜ᵃ
4	1	0	@mydayfess seru bgt nntn konser day6 sama mamanya :")

References

[1] S. Petrović, M. Osborne, R. McCreadie, C. MacDonald, I. Ounis, and L. Shrimpton, "Can Twitter Replace Newswire for Breaking News?," in *International Conference on Weblogs Social Media*, 2013, pp. 713–716.

[2] M. Jordaan, "Poke Me, I'm a Journalist: The Impact of Facebook and Twitter on Newsroom Routines and Cultures at Two South African weeklies," *Ecquid Novi African Journal Studies*, vol. 34(1), pp. 21–35, 2013.

[3] S. Schifferes, N. Newman, N. Thurman, D. Corney, A. Göker, and C. Martin, "Identifying and Verifying News through Social Media: Developing a User-Centred Tool for Professional Journalists," *Digital Journal*, vol. 2(3), pp. 406–418, 2014.

[4] F. Chen and D. B. Neill, "Human Rights Event Detection from Heterogeneous Social Media Graphs," *Big Data*, vol. 3(1), pp. 34–40, 2015.

[5] I. Inuwa-Dutse, M. Liptrott, and I. Korkontzelos, "Detection of Spam-Posting Accounts on Twitter," *Neurocomputing*, vol. 315, pp. 496–511, 2018, doi: 10.1016/j.neucom.2018.07.044.

[6] V. Balakrishnan, S. Khan, T. Fernandez, and H. R. Arabnia, "Cyberbullying Detection on Twitter Using Big Five and Dark Triad Features," *Personality and Individual Differences*, vol. 141, pp. 252–257, 2019, doi: 10.1016/j.paid.2019.01.024.

[7] F. Kunneman and A. Van den Bosch, "Event Detection in Twitter: A Machine-Learning Approach Based on Term Pivoting," in *Proceedings of the 26th Benelux Conference on Artificial Intelligence*, 2014, pp. 65–72.

[8] Ø. Repp and H. Ramampiaro, "Extracting News Events from Microblogs," *Journal of Statistics and Management Systems*, vol. 21, pp. 695–723, 2018, doi: 10.1080/09720510.2018.1486273.

[9] O. Ajao, D. Bhowmik, and S. Zargari, "Fake News Identification on Twitter with Hybrid CNN and RNN models," in *Proceedings of the 9th International Conference on Social Media and Society*, 2018, pp. 226–230.

[10] S. Dabiri and K. Heaslip, "Developing a Twitter-Based Traffic Event Detection Model Using Deep Learning Architectures," *Expert Systems with Applications*, vol. 118, 2018, doi: 10.1016/j.eswa.2018.10.017.

[11] M. Hasan, M. A. Orgun, and R. Schwitter, "Real-time Event Detection from The Twitter Data Stream Using the TwitterNews+ Framework," *Information Processing & Management*, vol. 56(3), pp. 1146–1165, 2019, doi: 10.1016/j.ipm.2018.03.001.

[12] J. Masci, U. Meier, D. Cireşan, and J. Schmidhuber, "Stacked Convolutional Auto-Encoders for Hierarchical Feature Extraction," in *International Conference on Artificial Neural Networks*, Berlin, Heidelberg, 2011, pp. 52–59, doi: 10.1007/978-3-642-21735-7_7.

[13] K. Greff, R. K. Srivastava, J. Koutník, B. R. Steunebrink, and J. Schmidhuber, "LSTM: A Search Space Odyssey," *IEEE Transactions on Neural Networks and Learning Systems*, vol. 28(10), pp. 2222–2232, 2017, doi: 10.1109/TNNLS.2016.2582924.

[14] A. J. McMinn, Y. Moshfeghi, and J. M. Jose, "Building a Large-Scale Corpus for Evaluating Event Detection on Twitter," in *Proceedings of the 22nd ACM International conference on Information & Knowledge Management*, 2013, pp. 409–418.

[15] S. Maneth, "Data science: 30th British International Conference on Databases, BICOD 2015 Edinburgh, UK, 2015 proceedings," Lect. Notes Comput. Sci. (including Subser. Lecture Notes in Artificial Intelligence, vol. 9147, no. January 2016, 2015.