Twitter Sentiment Analysis in Under-Resourced Languages using Byte-Level Recurrent Neural Model

Ridi Ferdiana[1], Wiliam Fajar[2], Desi Dwi Purwanti[3], Artmita Sekar Tri Ayu[4], Fahim Jatmiko[5]
Department of Electrical Engineering and Information Engineering
Universitas Gadjah Mada, Yogyakarta, Indonesia[1, 2, 3, 4]
Microsoft Innovation Center, Universitas Gadjah Mada, Yogyakarta, Indonesia[5]

***Abstract*—Sentiment analysis in non-English language can be more challenging than the English language because of the scarcity of publicly available resources to build the prediction model with high accuracy. To alleviate this under-resourced problem, this article introduces the leverage of byte-level recurrent neural model to generate text representation for twitter sentiment analysis in the Indonesian language. As the main part of the proposed model training is unsupervised and does not require much-labeled data, this approach can be scalable by using a huge amount of unlabeled data that is easily gathered on the Internet, without much dependencies on human-generated resources. This paper also introduces an Indonesian dataset for general sentiment analysis. It consists of 10,806 twitter data (tweets) selected from a total of 454,559 gathered tweets which taken directly from twitter using twitter API. The 10,806 tweets are then classified into 3 categories, positive, negative, and neutral. This Indonesian dataset could help the development of Indonesian sentiment analysis especially general sentiment analysis and encouraged others to start publishing similar dataset in the future.**

***Keywords*—Sentiment analysis; under-resourced problem; Indonesian dataset; twitter**

I. Introduction

Sentiment analysis is a problem of systematically identifying and studying personal information. This is commonly translated into the task of classifying polarity detection (thus this term is used interchangeably): Given a piece of written text, the problem is to categorize text into positive or negative classes or can be expanded to the ordinal classification problem. It assigns text to a value (e.g., Numbers from -2 to +2) instead of only positive or negative. There are some who think that polarity detection is not only related to the term sentiment analysis, polarity detection is only one subtask of the sentiment analysis process [1], [2]. However, this article uses the term sentiment analysis and polarity detection interchangeably as a focus on this task in this work.

Plenty of methods have been introduced to deal with sentiment analysis problem in previous studies. In general, the method can be either supervised or unsupervised. A lexicon-based approach is often used in unsupervised cases, where a list of words with their sentiment score is required to assign overall sentiment of a document. On the other hand, supervised machine learning techniques can also be considered to build sentiment analysis system because there is no such exact mapping between patterns of character in the text and the polarity of the sentiments (positive or negative). To produce a model from a series of data and let the computer to learn the patterns. There are several machine learning methods for classifying polarity detection: neural networks [3], [4], decision trees [5], support vector machines (SVM) [6], and naive Bayesian [7]. Feature pre-processing and extraction are carried out before classification, which requires large computing power.

Both machine-learning and lexical-based methods need extensive resources that are manually prepared. Lexical-based methods need sentiment lexicons, while machine-learning-based needs a lot of labeled data. This may be scarcely available to many languages, especially non-English languages such as Indonesian. Human-generated resources are expensive, which require much time and manual labor. This problem motivates us to ease the problem by adding a resource that may help other researchers to conduct research in this area and proposing a sentiment analysis system that leverages unsupervised approach, which minimizes the need of human-generated resources.

In this paper, it is proposed an unsupervised method for addressing the under-resourced problem in sentiment analysis for the Indonesian language. This article presents a methodology to use a byte-level self-supervised neural network to generate sentence representation in sentiment analysis in Indonesian, under the hypothesis that leveraging this method with an existing popular technique such as TF-IDF method will make improvements in this sentiment analysis classification performance.

Our main contributions are as follows:

- The use of unsupervised approach to minimize the under-resourced problem in the Indonesian language, particularly the byte-level recurrent neural model to generate a representation of sentences.
- To gather twitter dataset that contains 10,806 labeled samples and 454,559 unlabeled samples, hoping this would be one resource of doing evaluation benchmark when building a sentiment analysis system in the Indonesian language.

II. Related Work

This section overviews existing research on sentiment analysis, focusing on sentiment analysis in general, with emphasis on the Indonesian language.

The work conducted in this paper is sponsored by Microsoft Rinna

A. Feature Extraction

Feature extraction techniques are used to compress the data in a more compact way than the raw data such that the redundancy is removed while retaining relevant information [8]. Data patterns can be more easily discovered so this will ease the classification task. A good feature is required to have discriminatory properties, i.e., maximizing inter-class variability while minimizing intra-class variability. Machine learning systems can increase with the appropriate representation of features.

One of favorite feature in sentiment analysis is the Lexicon feature [9]. It uses a list of negative and positive words that are used to express the positive and negative sentiment. In the English dataset, the researcher can use SentiWordnet [10], or other similar databases. Based on Lexicon features, to determine the sentiments given by text, it is not necessary to train machine learning-based classifiers. it calculates the positive and negative polarity of text based on the occurrences of the positive and negative word using Pointwise Mutual Information (PMI) [11]. In this task, it may check whether the total value of the threshold is certain to determine its polarity. The Lexicon feature can be useful if have a complete list of words or can make a dictionary. But to make it takes a long time for manual work and may not be available in non-English languages. In addition, the N-gram feature is also popularly used by many works [1], [6] (a set of words / n-pair words). it counts the occurrence of words (1-gram or unigram) or n-pairs of words in the dictionary that have been determined to form the feature n-gram sentence.

Feature extraction techniques produce hand engineering features, i.e. the process of generating hand-crafted features is explicitly driven by predetermined algorithms. Designing such an algorithm takes time and requires a human expert. Therefore, there are several attempts to delegate the task of design extraction of this feature automatically to a computer. For classification, computers can determine for themselves which should be the best feature, considering raw data. This approach is called learning representation. Because computer data processing is increasing and more data is available, this is becoming popular in modern machine learning systems.

For text called word2vec, Mikolov et al. [12] proposed a method of learning representation that transforms words into multi-dimensional vectors. This transformation is carried out by a neural network encoder that is trained to predict the following words in the text. First, the neural network is initialized randomly and converts the word into a random vector. But once trained, encoders change the word vectors in a structure so that the words with similar meaning have a close distance and a pair of words that have a certain relationship will likely have the same distance as the other pairs of words that have the same relationship. This word embedding feature can be used in several specific NLP tasks such as sentiment analysis, text summarization and generating sentences that are given images.

To predict the preceding and succeeding sentence given a sentence, Kiros et al. [13] extends the success of the word insertion method by building sentence encoding by training neural networks. Inspired by these works, Radford et al. [14] proposed the learning of byte-level text representation. They propose an encoder that can produce multi-dimensional feature vectors from a sentence. These neural network encoders are also repeatedly trained to predict text that is not labeled and widely available on the internet. Instead of using word sequences as inputs, encoders are fed character by character. To predict the semantic polarity of text, encoders are sorted by a particular classifier machine learning.

B. Recent Workshops on Sentiment Analysis

SemEval is one of the challenges that has been held every year since 2013 [15]. In 2017, there were 48 teams that were successfully drawn by SemEval to be involved in the task of tweet sentiment analysis. In this task, participants will determine the sentiment value given by the tweet data. There are several techniques used, namely Logistic Regression, Random Forest, Maximum Entropy, Conditional Random Field, and Naif Bayes classifiers. SVM is more popular, and the best performing teams use such deep neural network as Convolutional Neural Network (CNN), Long Short-Term Memory (LSTM), and Gated Recurrent Unit (GRU). The top 5 teams for the English dataset use lexical, semantic features, dense word embedding, and the use of ensemble features. Available metadata for each tweet, such as the number of followers, user id, location, time zone, name, and a number of friends was not used by participants because they cannot increase their model performances. Analyzing and using effective metadata is very possible for future work.

SemEval continued to be held in 2018 [16]. A similar task is found in Task 1: Affect in Tweets, valence ordinal classification subtask. In the subtask, a participant is required to classify tweets into one of seven classes (-3, 2, -1, 0, +1, +2, +3) that represents the correct sentiment value. The best performing teams still used deep learning techniques such as CNN, LSTM, Gated Recurrent Unit (GRU), Bi-LSTM, and word embedding feature extraction methods combined with manually engineered features, i.e., sentiment and emoticon lexicons [17]. In the Arabic assignments, many teams use pre-processing techniques before doing the classification, such as stemming, lemmatization (MADAMIRA tool). By evaluating the result, it can be seen that although deep learning is interesting, performance can be improved by working together with hand engineering methods, which include feature pre-processing and extraction methods.

The use of non-English languages is still limited to several languages such as Spanish and Arabic. It may also be interesting to see other small languages such as Indonesian, Javanese, or Malay can be objects for the coming SemEval.

C. Challenges on Non-English Sentiment Analysis

Sentiment analysis of non-English texts has limited resources, such as Indonesian. Many unlabeled data available on the internet and labeled data are rarely available, so building an effective supervised machine learning system in non-English data can be challenging, especially if deep learning is used. In developing the Lexicon feature for sentiment analysis, a dictionary is needed in the form of a collection of negatives and positive sentiment words, which are not publicly available in Indonesian to the best of our knowledge.

The approach taken to build a lexicon dictionary can be done using automatic understanding in the available English lexicon database. Franky et al. [18] built an Indonesian lexicon dictionary by translating available English lexicons (Opinion lexicons, Harvard General Inquirer, SentiWordnet, and Bing Liu Opinions) using Google, Moses translations (statistical machine translation systems), and Kamus.net online dictionaries. Other works generally also use the availability of the English Lexicon database and conduct a mapping between English words and appropriate words in the language in which they work for the manufacture of non-English lexicon dictionaries [19], [20].

There are several works using the remote-controlled corpus to reduce the effort to build labeled datasets. Assigning labels to datasets is carried out remote monitoring using weak labels. For example, [21] forms a large number of labeled twitter datasets by setting tweets as positive if the text contains positive emoticons (such as ":)") and assigning it as negative if it contains any negative emoticon (such as ":("). Author in [21] proposed a multiple language CNN for sentiment analysis, which is CNN that is trained with various corpus with different language to increase the number of data samples further, so it is able to handle multiple languages in a text. However, it does not perform well compared with CNN trained with single language dataset. Author in [26] proposes distant learning as an additional training set for Convolutional Neural Network for sentiment analysis. It is shown that the usage of distant learning increases performance by 5 percent.

By manually giving labels, some researchers decided to create their own non-English datasets. Franky et al. [18] built a labeled dataset by manually annotating 446 sentences originating from user reviews in several domains on the KitaReview website. Others [22] use a semi-supervised approach to sentiment analysis to overcome the scarcity of labeled data. They first made a small dataset of around 400 words of data labeled and collected a lot of non-labeled data (3 million). Furthermore, slowly labeled data is labeled using a classifier that is trained in a small labeled dataset.

III. Methodology

The article uses Multiplicative Long Short-Term Memory (MLSTM) Cell to build a recurrent neural network model. MLSTM cells are modified version of Long Short-Term Memory (LSTM) cells that are hybridized with Multiplicative Recurrent Neural Network [23]. MLSTM cells are observed to converge faster than LSTM during training [23]. The model considers an input sentence as a sequence of characters. Each character is encoded by a byte of Unicode encoding UTF-8. During training, the hidden state of the model is updated for each byte and the model predicts a probability distribution over the next possible character. The hidden state of the model encodes all information the model has learned from the first sequence and it provides relevant information to predict the future bytes of the sequence.

It is explored the optimal hyperparameter of the neural network model. it chose the embedding size of 128 and the RNN size of 4096. All of the states are assigned to 0 at the beginning, Adam method is used to train the neural network model with a learning rate of 5×10^{-4} that was decreased to zero over the training iterations. The model is trained with a dataset that contains tweets sentences. Because the training objective is to predict the next sequence of input, it does not need labeled data samples so the dataset can be easily gathered from the Internet. In this experiment, it used 454,559 unlabeled data gathered from the twitter API.

Once the recurrent neural network model is trained, the neural network model is used as a feature extractor of text input. The model processes an input byte by byte, forming its hidden state that is regarded as a multidimensional dense vector representation of the input sentence. The vector is then used along with a classifier such as SVM and trained with labeled data to create a sentiment prediction model.

A. Dataset

Supervised learning required a huge amount of labeled data, which is hard to come by. This is especially true for Indonesian dataset. Although gaining a lot of popularity, most of the research conducted in Indonesian sentiment analysis sometimes use only 1,000 to 5,000 data in their experiment.

The article chooses twitter as our dataset because twitter is one of the most popular sources for sentiment analysis data. Tweets consist of a maximum of 280 characters. And due to the nature of Twitter itself, most tweets only consist of text, unlike other social media such as Facebook or Instagram. The data gathering methods from twitter is also relatively easy. With twitter API, researchers can access not only tweets but also location, languages, user information, etc., which makes it easier to gather any specific data needed. There is also hashtag in twitter that makes data gathering process for sentiment analysis on a certain topic easier. All of these things on top of the other make Twitter our choice for a sentiment analysis dataset source.

This dataset in this work is originated from Twitter and taken with Twitter API between September and December of 2018. Each of the tweets has a maximum character of 140 from the gathered data of 454,559 tweets, 10,806 tweets were selected to be labeled and used. The example of the dataset can be seen in Table I.

TABLE. I. Sample of the Labeled Indonesian Dataset

Sentiment	Indonesian Tweet
-1	lagu bosan apa yang aku save ni huhuhuhuhuhuhuhuhuhuuuuuuuuuuuuuu
-1	kita lanjutkan saja diam ini hingga kau dan aku mengerti tidak semua kebersamaan harus melibatkan hati
1	doa rezeki tak putus inna haa zaa larizquna maa lahu min na fadesungguhnya ini ialah pemberian kami kepada kamu
1	makasih loh ntar kita bagi hasil aku 99 9 sisanya buat kamu
0	yg selama ini nunggun video dari aku nih retweet yg mau full
0	barusan liat tulisan di belakang truk rela injek kopling demi kamu bisa shopping
-1	ku kira aku introvert ternyata karna nga punya uang aja jadi males ke mana
1	ya aku akan menjadi satu satunya bukan nomor satu tetapi satu satunya
-1	aku aja capek sama diriku sendiri apalagi kamu maaf ya '
0	aku nang kampus untuk ngopi ketemu wong2 podo kabeh takok e lapo mengubah sosyeti

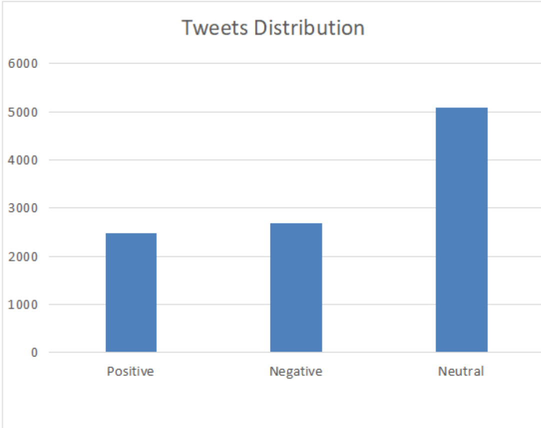

Fig. 1. The Distribution of the Sentiment in the Indonesian Dataset.

The selected tweets then labeled manually with three variables which is positive, labeled as 1, negatively labeled as -1, and neutral, labeled as 0. From the total of 10,806 tweets, 2482 are labeled as positive tweets, 2691 as negative tweets, and 5084 as neutral tweets; the distribution can be seen in Fig. 1.

There are 1:1:2 ratio of positive, negative, and neutral tweets, but considering the main purpose of this dataset is general sentiment analysis, it is concluded that the balance between each category is sufficient to be used even besides general sentiment analysis. The tweets are saved in CSV format with two columns. These tweets also have been lightly processed to remove noise so it can be conveniently used, the noise removed are symbols, URL links, username, and hashtag. The dataset can be downloaded as a common creative copyleft license at http://ugm.id/idsadataset

IV. Result and Discussion

It is conducted experiments to evaluate the effectiveness of the model for generating text representation from the byte-level recurrent neural network. It is shown comparison results between the proposed model and other typical sentiment analysis models: SVM classifier with TF-IDF features, and sentiment lexicons using AFINN [24]. The performance is evaluated using standard evaluation metrics: accuracy. Accuracy is defined as:

$$Acc = \frac{true\_negatives + true\_positives}{true\_positives + false\_positives + false\_negatives + true\_negatives}$$

To get sentiment value using AFINN, it is translated the sentiment lexicons dictionary from English to Indonesian using Microsoft translation service. Sentiment analysis is performed by cross-checking the string tokens (words, emojis) with the translated AFINN list and getting their respective scores.

TABLE. II. Effectiveness Comparison among our Model and other Typical Approaches

Method	Accuracy
1-gram TF-IDF vector	0.528
byte-level recurrent neural model	0.543
AFINN sentiment lexicon	0.455

Table II shows the results of our model on our labeled tweet datasets and TF-IDF features with an SVM classifier. TF-IDF representation provides 52.8 % of accuracy, while byte-level generated features give 54.3% of accuracy. There is 2.84 % improvement when using byte-level generated features compared to typical TF-IDF features. The result can be improved by concatenating the feature vectors of TF-IDF and the character level word embedding and making use of principal component analysis dimensionality reduction technique. AFINN sentiment lexicon methods give 45.48 % of accuracy.

V. Conclusion

In this work, it is proposed the use of byte-level recurrent neural networks with multiplicative long short-term memory cells for generating a representation of sentences, which are combined with a classifier (such as SVM) to generate a prediction of sentiment. The hybrid representation addition with sentiment lexicon could improve accuracy.

It cannot be said that the proposed methodology performance beat the state-of-the-art. On the other hand, state-of-the-art approaches, require a considerable amount of human work, which are labeled dataset and sentiment lexicon dictionaries. The proposed methodology is simple and does not rely on human-generated resources so it can be scalable to a larger dataset. However, it requires huge computational resources to conduct this methodology, as it has to process a huge amount of unlabeled data.

In the future, the research will aim to conduct experiments towards the following directions, in order to improve its performance: (a) improvement of the use pre-processing methods of text, (b) Apply the methodology into a larger dataset (e.g. contains millions of data samples).

Acknowledgment

The authors would like to thank you to Microsoft Rinna for their kind supports in providing grants and facilities for conducting this research.

References

[1] E. Cambria, B. Schuller, Y. Xia, and C. Havasi, "New Avenues in Opinion Mining and Sentiment Analysis," IEEE Intell. Syst., vol. 28, no. 2, pp. 15–21, 2013.

[2] I. Chaturvedi, E. Cambria, R. E. Welsch, and F. Herrera, "Distinguishing between facts and opinions for sentiment analysis: Survey and challenges," Inf. Fusion, 2018.

[3] X. Zhang and X. Zheng, "Comparison of Text Sentiment Analysis Based on Machine Learning," 2016 15th Int. Symp. Parallel Distrib. Comput., pp. 230–233, 2016.

[4] J. Wehrmann, W. Becker, H. E. L. Cagnini, and R. C. Barros, "A character-based convolutional neural network for language-agnostic Twitter sentiment analysis," 2017 Int. Jt. Conf. Neural Networks, pp. 2384–2391, 2017.

[5] Z. Rezaei and M. Jalali, "Sentiment analysis on Twitter using McDiarmid tree algorithm," 2017 7th Int. Conf. Comput. Knowl. Eng. ICCKE 2017, vol. 2017-Janua, no. Iccke, pp. 33–36, 2017.

[6] Ike P. Windasari, F. N. Uzzi, and iman satoto Kodrat, "Sentiment Analysis on Twitter Posts: An analysis of Positive or Negative Opinion on Gojek," in IEEE International Conference on Information technology, Computer, and Eletrical Engineering, 2017.

[7] T. Ghorpade and L. Ragha, "Featured based sentiment classification for hotel reviews using NLP and Bayesian classification," Proc. - 2012 Int. Conf. Commun. Inf. Comput. Technol. ICCICT 2012, pp. 1–5, 2012.

[8] S. Pasarate and R. Shedge, "Comparative study of feature extraction techniques used in sentiment analysis," 2016 Int. Conf. Innov. Challenges Cyber Secur., no. Iciccs, pp. 182–186, 2016.

[9] M. Taboada, J. Brooke, M. Tofiloski, K. Voll, and M. Stede, "Lexicon-Based Methods for Sentiment Analysis," Comput. Linguist., 2011.

[10] S. Baccianella, A. Esuli, and F. Sebastiani, "SentiWordNet 3.0 : An Enhanced Lexical Resource for Sentiment Analysis and Opinion Mining SentiWordNet," Analysis, vol. 10, pp. 1–12, 2010.

[11] K. W. Church and P. Hanks, "Word Association Norms, Mutual Information, and Lexicography," in Proceedings of the 27th annual meeting on Association for Computational Linguistics -, 1989, vol. 16, no. 1, pp. 76–83.

[12] T. Mikolov, G. Corrado, K. Chen, and J. Dean, "Efficient Estimation of Word Representations in Vector Space," Proc. Int. Conf. Learn. Represent. (ICLR 2013), pp. 1–12, 2013.

[13] R. Kiros et al., "Skip-Thought Vectors," 2015.

[14] A. Radford, R. Jozefowicz, and I. Sutskever, "Learning to Generate Reviews and Discovering Sentiment," Apr. 2017.

[15] S. Rosenthal, N. Farra, and P. Nakov, "SemEval-2017 Task 4: Sentiment Analysis in Twitter," Proc. 11th Int. Work. Semant. Eval., pp. 502–518, 2017.

[16] S. Mohammad, F. Bravo-Marquez, M. Salameh, and S. Kiritchenko, "SemEval-2018 Task 1: Affect in Tweets," Proc. 12th Int. Work. Semant. Eval., pp. 1–17, 2018.

[17] K. Baktha and B. K. Tripathy, "Investigation of recurrent neural networks in the field of sentiment analysis," Proc. 2017 IEEE Int. Conf. Commun. Signal Process. ICCSP 2017, vol. 2018-Janua, pp. 2047–2050, 2018.

[18] Franky, O. Bojar, and K. Veselovská, "Resources for Indonesian Sentiment Analysis," Prague Bull. Math. Linguist., vol. 103, no. 1, pp. 21–41, 2015.

[19] A. Bakliwal, P. Arora, and V. Varma, "Hindi Subjective Lexicon: A Lexical Resource for Hindi Polarity Classification," eighth Int. Conf. Lang. Resour. Eval., 2012.

[20] V. Perez-Rosas, C. Banea, and R. Mihalcea, "Learning Sentiment Lexicons in Spanish," Proc. Eighth Int. Conf. Lang. Resour. Eval., 2012.

[21] J. Deriu et al., "Leveraging Large Amounts of Weakly Supervised Data for Multi-Language Sentiment Classification," 2017.

[22] N. F. F. Da Silva, L. F. S. Coletta, and E. R. Hruschka, "A Survey and Comparative Study of Tweet Sentiment Analysis via Semi-Supervised Learning," ACM Comput. Surv., vol. 49, no. 1, pp. 1–26, 2016.

[23] B. Krause, L. Lu, I. Murray, and S. Renals, "Multiplicative LSTM for sequence modelling," 2016.

[24] F. Å. Nielsen, "A new ANEW: Evaluation of a word list for sentiment analysis in microblogs," CEUR Workshop Proc., vol. 718, pp. 93–98, 2011.