IAES International Journal of Artificial Intelligence (IJ-AI)
Vol. 9, No. 2, June 2020, pp. 212~220
ISSN: 2252-8938, DOI: 10.11591/ijai.v9.i2.pp212-220

Sentiment analysis of informal Malay tweets with deep learning

Ong Jun Ying[1], Muhammad Mun'im Ahmad Zabidi[2], Norhafizah Ramli[3], Usman Ullah Sheikh[4]
[1]Department of Device Development Group, Intel Penang, Malaysia
[2,3,4]School of Electrical Engineering, Faculty of Engineering, Universiti Teknologi Malaysia, Malaysia

Article Info

Article history:

Received Jan 16, 2020
Revised Mar 6, 2020
Accepted Apr 17, 2020

Keywords:

Bahasa Indonesia
Convolutional neural network
Malay
Natural language processing
Sentiment analysis

ABSTRACT

Twitter is an online microblogging and social-networking platform which allows users to write short messages called tweets. It has over 330 million registered users generating nearly 250 million tweets per day. As Malay is the national language in Malaysia, there is a significant number of users tweeting in Malay. Tweets have a maximum length of 140 characters which forces users to stay focused on the message they wish to disseminate. This characteristic makes tweets an interesting subject for sentiment analysis. Sentiment analysis is a natural language processing (NLP) task of classifying whether a tweet has a positive or negative sentiment. Tweets in Malay are chosen in this study as limited research has been done on this language. In this work, sentiment analysis applied to Malay tweets using the deep learning model. We achieved 77.59% accuracy which exceeds similar work done on Bahasa Indonesia.

Corresponding Author:

Muhammad Mun'im Ahmad Zabidi,
Department of Electronics and Computer Engineering,
School of Electrical Engineering,
Faculty of Engineering, Universiti Teknologi Malaysia,
81310 UTM Johor, Malaysia.
Email: munim@utm.my

1. INTRODUCTION

The advancement of the Internet causes the world to face ever-growing volumes of data in various forms. Applications such as social media, review sites, forums, blogs, and others generate enormous heaps of data in the form of sentiment, opinion, user's views, arguments about different social events, politics, products and more. The sentiments expressed by users have a very great influence on the readers, product vendors, and even politicians. As a result, many companies depend on Internet users' feedback and opinions to market their products and services on social media such as Twitter, Facebook, Snapchat, and Instagram. Companies are interested in knowing what users think about their services or products. The changes in these business models had created a huge business opportunity for the sentiment analysis of the data [1]. Malay is Malaysia's national language. In Malaysia, there are a significant number of users who are using Malay to express their opinions and arguments on social media. There is a huge business opportunity for building a Malay sentiment analysis model. However, very limited research has been attributed to Malay sentiment analysis [2].

Data on social media are mostly unstructured. The large amounts of unstructured data make it very difficult for a human to extract and summarize the opinion contained therein. Often, human errors will occur thus reducing the accuracy of the data analysis. Automated sentiment analysis will greatly reduce the workload for a human to analyze the data [3-4]. To exploit this opportunity, many start-ups are now providing sentiment analysis services for the public. Likewise, many big corporations are also building their in-house capabilities in sentiment analysis. These practical applications and industrial interests have provided a strong motivation for research in sentiment analysis.

2. LITERATURE REVIEW
2.1. Natural language processing

Natural Language Processing is a field that intersects of artificial intelligence, computer science, and linguistics. It refers to an interaction between machines and humans in natural language [5]. The ultimate goal of NLP is to enable computers to understand language as well as humans do. NLP is divided into three categories, which are:

– Natural Language Understanding: Computers' ability to understand what we say.
– Natural Language Generation: The generation of natural language by a computer.
– Speech Recognition: The translation of spoken language into text.

NLP had been widely applied to different applications, such as predictive typing, sentiment analysis, spell checking, spam detection, and others.

2.2. Sentiment analysis

Sentiment analysis (SA) is also known as opinion mining [6]. It refers to build a system to identify and extract opinions from the text. It often able to extract the attribute of the expression, which includes:

– Subject: Topic that is talked about.
– Polarity: Positive or negative opinion.
– Opinion holder: Entity that expresses an opinion.

Sentiment analysis generates great interest mainly due to having many practical applications. The constant expansion of the Internet produces large volumes of textual data expressing opinions. With sentiment analysis, this messy and unstructured information will be automatically transformed into structured data. These data contain the public opinion about services, products, brands and different topics that people could express opinions about. The data is very valuable for commercial applications like marketing analysis, product feedback, customer services, public relationship, and others. In short, the SA system helps companies to make sense of this sea of unstructured text by automating business processes, getting actionable insights, and saving time on manual data processing. This research focuses solely on the polarity extraction aspect of SA.

There are several ways or approaches to implement the sentiment analysis system. As shown in Figure 1, these approaches are often be categorized into two major categories, which are machine learning approaches and lexicon-based approaches [7].

– Machine Learning Approaches: Learn the features from labelled data, then apply it to a set of unknown data.
– Lexicon Based Approaches: Work on the dictionary with a list of words/sentences with different polarity, output polarity will be some of the polarities on different words/sentences [8].

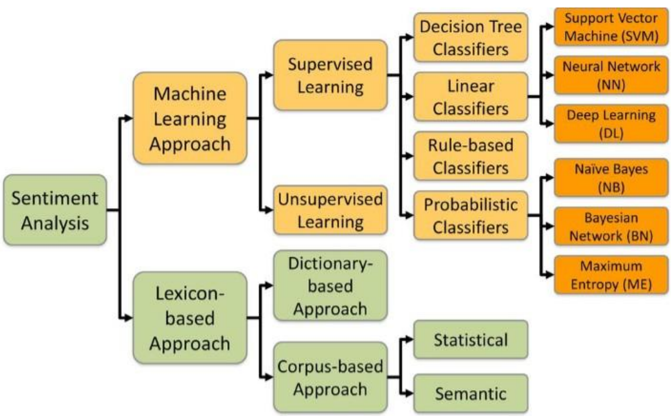

Figure 1. Sentiment classification techniques

Machine learning approaches rely on using algorithms to automatically extract the features from a data set. After the models have been sufficiently trained, it will be fed with unknown text and then returns with the corresponding polarity (positive, negative, or neutral). The sentiment analysis task is usually categorized in NLP under the text classification segment and therefore is a type of classifier.

2.3. Data pre-processing

Data pre-processing refers to the steps that need to be employed before it can be transformed into the numerical feature that can work with the machine learning algorithm [9]. The raw text comes with a variety of forms, for example, individual words, to sentences, multiple paragraphs, even with special characters and contains a lot of redundant information.

All these raw texts need to be cleaned up and trim down to the root format that can work well with the machine learning algorithm. Good text pre-processing enhances the classification accuracy and as well as training efficiency [10]. The different text pre-processing techniques that are often used are stated below:

- Tokenizer: To create a token for each word in the sentences/document.
- Capitalization: Changing all the words to lower case or upper case.
- Abbreviation Replacement: Expand the abbreviation to the original words.
- Stop word removal: Remove the meaningless words. For example: ’is’, ’the’, ’are’.
- Stemming: Process of removing the endings of the word. For example: changing 'studies' to 'study', changing 'studying' to 'study'.
- Lemmatization: Process of grouping the different inflected forms of a word. For example: changing 'studies' and 'studying' to 'study'.
- Spelling Correction: Recover the words from typos.
- Part of speech tagging: tag token for the type of the word. For example Nouns, Verbs, Adjectives.
- Remove numbers: Remove the numerical expression.
- Named entity recognition: label the words with respective entities. For example, name, location, dates, address.

Not all pre-processing techniques are suitable for every scenario. The techniques are chosen on a case by case basis.

2.4. Text to numerical representation

The computer only understands the data which is a numerical format, so do the machine learning algorithms. Before inputting the data for the machine learning algorithm to process, we need to convert the data from textual format to numerical format. This process is also known as text vectorization. The common techniques for text vectorization are:

- Bag Of Words (BoW): BoW is a basic and straightforward technique. With this method, the order and grammar of the words are discarded. It only tells whether a word is present in the document or not. Each word of the entire data set is corresponding to a column, if the particular exist in the input sentences, vector representation of this sentence has a 1 in the corresponding column for this word [11].
- Term Frequency – Inverse Document Frequency (TF-IDF): Each word within an input sentence is replaced with its TF-IDF score and created a vector out of these scores for each input sentence. The whole idea of this measure is to give more importance to the terms that are more specific to a certain class (TF) and reduce the importance of the term that is very frequent in the entire corpus (IDF) [12].
- Word Embeddings: A way of statistically extracting the meaning of a word from the text and representing it with a set of numbers. To be able to understand the meaning of a word, these models use contextual similarities. With this technique, words with similar meanings tend to have a similar representation. The word2vec method is commonly used for word embeddings [13].

2.5. Classification algorithms

Many classification algorithms exist in machine learning. For sentiment analysis, statistical models like Support Vector Machine (SVM), Naïve Bayes (NB) or neural networks are commonly used.

- Naïve Bayes (NB): Use Bayes’s Theorem (probabilistic algorithms) to predict the category of a text [14].
- Support Vector Machine (SVM): Text is represented as a point in a multidimensional space. The points will be mapped to different categories. New text is mapped onto the same space and predicted to belong to categories based on the region that lies [15].
- Artificial Neural Networks (ANN) and Deep Learning (DL): Using a diverse set of algorithms to imitate how the actual human brain works by employing artificial neural networks to process data [16].

Convolutional Neural Network (CNN) such as shown in Figure 2 is a type of ANN that is commonly used in image recognition. Initially, it was designed to process the pixel data, but recently CNN also has been applied to text classification tasks and achieve very good performance [17-19].

The architecture of CNN pretty much like a multi-layer perceptron that has been designed for reduced processing requirements. In general, CNN will consist of an input layer, a hidden layer, and output layers. The hidden layer includes multiple convolutional layers, normalization layer, pooling layer and fully connected layer [16]. CNN has forward and reverse transmission. In forward transmission, the data input structures undergo multiple layers of processes and output structures in the output layer, an activation function is needed in every layer. In reverse transmission, the probability of error is calculated using a given model result and the forward transmission result, and its transfer to the error function back to each respective layer, lastly the gradient descent technique is applied to tune the bias parameter and network weights to obtain better accuracy.

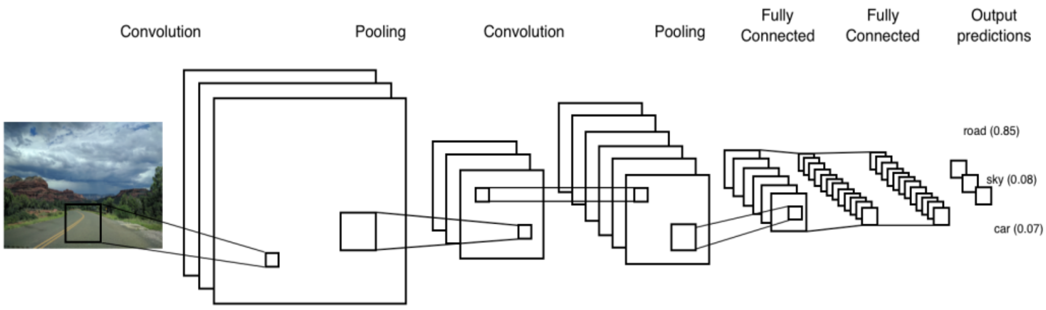

Figure 2. Convolutional Neural Network [16].

Training of CNN is done in two stages which are the training stage and testing stage. During the training stage, a set of labelled data needs to be provided. This stage will involve the convolution process, subsampling and multiplying the end outcome of the subsampling with certain artificial neural weights during the training process. While the testing stage is to apply input data and test on the trained model [10].

3. METHODOLOGY

Figure 3 shows the methodology used for this work. First, the dataset is prepared. It then goes through pre-processing, labelling and conversion to numerical representation. After these steps are done the data is ready use to train and test on the CNN model. Lastly, the performance of the sentiment analysis task is analyzed.

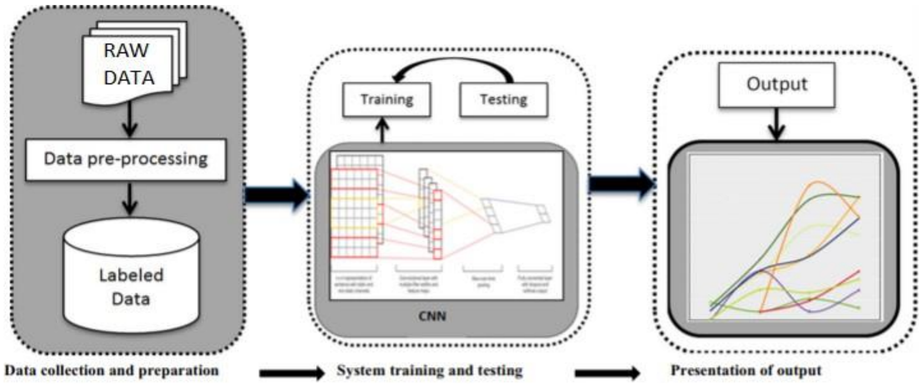

Figure 3. Data processing

3.1. Proposed CNN architecture

The proposed architecture shown in Figure 4 and Table 1 is the CNN architecture adopted from [20] with the addition of a dropout layer to reduce the overfitting problem.

1. Each word is mapped to a word vector representation

– The entire tweet can be mapped to a matrix of size $s \times d$, where s = number of words and d = dimension of the embedding space
2. Zero padding
– To make sure all tweets have the same matrix dimension.
3. Max pooling layer
– Extracts the most important feature for each convolution.
– Combine all the cmax of each filter into one vector.
4. Drop out layer
– Explicitly altering the network architecture at training time
5. Softmax layer
– Give out final classification probabilities

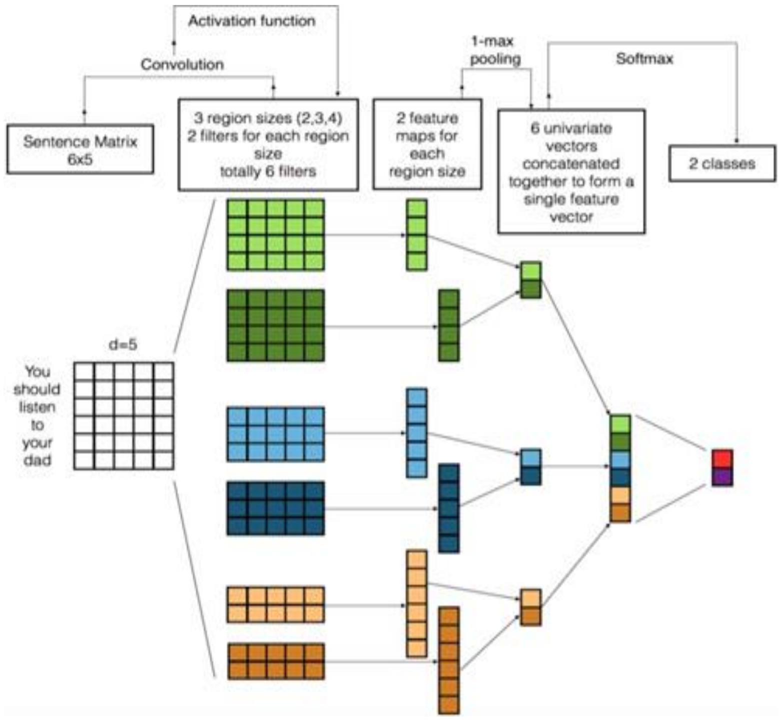

Figure 4. Proposed CNN architecture

Table 1. Baseline CNN configuration.

Parameter	Value
Filter Region	2,3,4
Feature Map	100
Pooling Layer	Max Pooling
Activation Function	ReLu
Drop Out Layer Rate	0.5

3.2. Data pre-processing

The raw text comes in a variety of forms, for example, individual words, to sentences, multiple paragraphs, even with special characters and contains a lot of redundant information. In our case, the data is referred to as raw tweets. All these raw texts need to be cleaned up to fit with the machine learning algorithm well. The data pre-processing steps proposed are as follows:
1. Split hashtag to words
2. Capitalization
3. Remove number, html tag, url
4. Stemming, lemmatizing (normalization)
5. Stop word removal

6. Tokenization
7. Data annotation/labelling (positive, negative, neutral)

3.3. Text to numerical representation

Before input, the data for the machine learning algorithm to process, the data in the textual format is converted to numerical format using Word2Vec. Word2Vec is a two-layer neural net that processes text. Its input is a text corpus and its output is a set of vectors: feature vectors for words in that corpus. While Word2vec is not a deep neural network, it turns text into a numerical form that deep nets can understand. The purpose and usefulness of Word2vec are to group the vectors of similar words in vector space. That is, it detects similarities mathematically. Word2vec creates vectors that are distributed numerical representations of word features, features such as the context of individual words.

3.4. Training strategy

After the data pre-processing step and model are defined, the next step to determine is the training strategies to train the model. A Graphics Processing Unit (GPU) is chosen to train the model. Using a GPU saves substantial amounts of time since a GPU can support a higher number of parallel tasks compare to the Central Processing Unit (CPU) due to the higher number of cores. At the software level, Keras package implements the training procedure [21]

Next, another important item on deep learning is the data set used to train the model. In this study, two different data sets were used separately to train the English SA model and Malay's SA model. For the English SA model, the sentiment140 data set was used [22]. This data set contains about 16 million English tweets with positive and negative labels. It is widely used in research purposes. For the Malay SA model, the MALAYA data set was used [23]. It contains about 6 million Malay Tweets with positive and negative labels.

For each of the data set, the training is configured to set 80% of the data to become the training data while the 20% that left will be separate half become validate data and another half become testing data. Besides, the training is configured to use 100 epochs.

4. RESULTS AND ANALYSIS

4.1. Text to numerical representation result

This section reports the results from the Word2Vec numeric conversion. After the Word2Vec model is trained, the weight of the words (numeric format) will be fixed and would not be changed. The trained Word2Vec model becomes one of the layers in the final CNN sentiment analysis model. The main purpose and usefulness of Word2vec are to group the vectors of similar words in vector space. As shown in Figure 5, Word2Vec shows the mathematical similarities for the top 20 most similar terms for different input words.

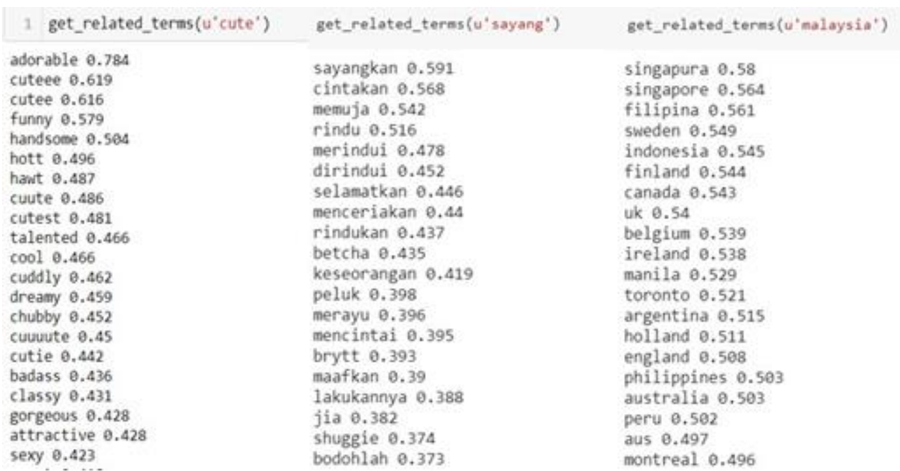

Figure 5. Word2Vec similarities results

4.2. CNN model hyper-parameter tuning

The hyper-parameters are the settings can be tuned to optimize the deep learning model. In this project there are few hyper-parameter had been determined and tuned. These are the filter region size, feature maps and drop out layer rate. Tables 2-6 reports the various adjustments made to the architecture during training.

Table 2. Results of difference single filter region sizes

Iteration	Region Size	Accuracy
T1	1	76.00%
T2	3	77.30%
T3	5	76.93%
T4	7	76.52%
T5	10	76.42%
T6	15	76.68%
T7	20	76.91%
T8	30	76.58%

Table 3. Results of different multi-filter region sizes

Iteration	Multi-Region Size	Accuracy
T1	2,3,4	76.56%
T2	3,4,5	76.85%
T3	4,5,6	76.74%
T4	5,6,7	76.09%
T5	7,8,9	76.59%
T6	3,4,5,6	76.85%
T7	6,7,8,9	76.80%
T8	3,3,3	77.37%
T9	3,3,3,3	76.48%

Table 4. Results of difference feature maps

Iteration	Feature Maps	Accuracy
T1	10	76.35%
T2	50	77.12%
T3	100	77.37%
T4	200	77.59%
T5	400	77.15%
T6	600	77.56%
T7	1000	77.26%
T8	2000	77.37%

Table 5. Result of different drop out rate

Iteration	Region Size	Accuracy
T1	0.1	76.17%
T2	0.2	77.37%
T3	0.3	77.17%
T4	0.4	77.34%
T5	0.5	77.59%
T6	0.6	77.40%
T7	0.7	76.99%
T8	0.8	76.91%
T8	0.9	77.44%

After going through many iterations of the hyper-parameter tuning. The best CNN model configuration that obtained are shown in Table 6.

Table 6. Best configuration after hyper-parameter tuning

Feature	Parameter
Filter Region	3,3,3
Feature Map	200
Pooling Layer	Max Pooling
Activation Function	ReLu
Drop Out Layer Rate	0.5

4.3. Model performance and accuracy

After tuning all the parameters, the English and Malay Sentiment Analysis CNN models were be trained and the final accuracy results were obtained. The results of our models against other methods are listed in Table 7.

Table 7. Accuracy comparison with published works.

Model	Accuracy
SVM [25]	75.86%
Naive Bayes [25]	77.45%
Indonesian CNN without Normalizer [25]	69.92%
Indonesian CNN with Normalizer [25]	65.45%
Indonesian LSTM with Normalizer [25]	73.22%
This work English SA CNN	81.87%
This work Malay SA CNN	77.59%

Very limited work on Malay sentiment analysis has been done using deep learning approaches. Therefore, the comparison can only be done against a similar work in Bahasa Indonesia by [24]. There is another work done in Bahasa Indonesia which achieved 100% but the dataset was too small (less than 300 samples) [25]. Although there are some limitations were determined in this comparison, but it can serve as an indicator to provide a general view on the current standing of this project.

The accuracy obtained from our models is comparable and slightly better than [24]. From the table, SVM and Naïve Bayes are the machine learning approaches with accuracies close to the results obtained in this work. However, deep learning approaches have a significant advantage due to its data-driven characteristic meaning that the more data is fed the higher its accuracy. On the other hand, other machine learning algorithms will hardly improve their accuracy after reaching a certain value.

5. CONCLUSION

In conclusion, the deep learning approach has been successfully applied in the task of Malay sentiment analysis. The developed sentiment analysis model was able to classify the tweet text into two sentiment categories which are positive and negative. The deep learning architecture that was used to build the Malay Sentiment Analysis model was based on the Convolutional Neural Network (CNN). To feed the text data to the Convolution Neural Network, the Word2Vec word embedding model was built from scratch to convert the text input to numerical representation. The model has also been going through the hyper-parameter tuning to achieve its optimal performance. The Malay sentiment analysis CNN model has been successfully built and been validated to achieved accuracy up to 77.59%.

There are some potential enhancement and modification can be done on the current design. Firstly, the current design uses static word embedding, which means after the word embedding model was trained, the weight for the words (numerical representation) would be fixed. Dynamic word embedding is a possible enhancement. Using dynamic word embedding, the weight of the word embedding model can be tuned while on the CNN training state. Lastly, the different text pre-processing techniques can explore to enhance the model. For text classification, text-processing is equally important with the deep learning model building. A little modification done on this stage will highly influence the outcome.

ACKNOWLEDGEMENTS

This work is supported by Universiti Teknologi Malaysia (UTM). This work is financially supported by the Geran Universiti Penyelidikan (GUP) 15J35. The authors would like to express their deepest gratitude to the Research Management Center (RMC), UTM for making this research a success.

REFERENCES

[1] Gao, Shanshan, Jinxing Hao, and Yu Fu. "The application and comparison of web services for sentiment analysis in tourism." *12th International Conference on Service Systems and Service Management (ICSSSM)*. IEEE, 2015.
[2] Handayani, D., Bakar, N. S. A. A. and Yaacob, H. *Sentiment Analysis for Malay Language: Systematic Literature Review*. 2018 International Conference on Information and Communication Technology for the Muslim World (ICT4M). IEEE. 2018. 305-310.
[3] E. Cambria, B. Schuller, Y. Xia, and C. Havasi, “New avenues in opinion mining and sentiment analysis.” *Intelligent systems, IEEE*, vol. 28, no. 2, pp. 15-21, March 2013.
[4] B. Pang, Lillian Lee, and S. Vaithyanathan. "Thumbs up?: sentiment classification using machine learning techniques." *Proc. ACL-02 conference on Empirical methods in natural language processing, Association for Computational Linguistics*, vol 10. 2002.
[5] J. -Y. Nie, D. Zhao and Y. Feng. *Natural Language Processing and Chinese Computing*. Springer. 2014.
[6] B. Pang and L. Lee. "Opinion mining and sentiment analysis." *Foundations and Trends® in Information Retrieval*, vol. 2: No. 1-2, pp. 1-135, 2008.
[7] V. Korde and C.N. Mahender, “Text classification and classifiers: A survey.” *International Journal of Artificial Intelligence & Applications*, vol. 3, no. 2, p.85, 2012.
[8] M. Taboada, J. Brooke, M. Tofiloski, K. Voll, and M. Stede. Lexicon-based methods for sentiment analysis. *Computational linguistics*, vol. 32, no2, pp. 267-307, 2011.
[9] J. Yao. “Automated Sentiment Analysis of Text Data with NLTK.” *Journal of Physics: Conference Series, IOP Publishing*, vol. 1187, no. 5, 2019.
[10] A. Kulkarni and A. Shivananda. *Natural Language Processing Recipes: Unlocking Text Data with Machine Learning and Deep Learning Using Python*. Apress; 2019.
[11] ZS Harris, “Distributional structure”, *Word*, vol. 10, no. 2-3, pp. 146-162, 1954.
[12] J. Ramos. “Using tf-idf to determine word relevance in document queries,” *Proc.First instructional conference on machine learning*, vol. 242, pp. 133-142, 2003.
[13] T. Mikolov, K. Chen, G. Corrado and J. Dean, “Efficient estimation of word representations in vector space,” *arXiv preprint arXiv:1301.3781*, 2013,

[14] G.I. Webb, J.R. Boughton, and Z. Wang. "Not so naive Bayes: aggregating one-dependence estimators," *Machine learning*, vol. 58, no. 1, pp.5-24, 2005.
[15] V. Vapnik, *The nature of statistical learning theory*. Springer science & business media, 2013.
[16] M.M. Lopez and J. Kalita. "Deep Learning applied to NLP," *arXiv preprint arXiv:1703.03091*. 2017.
[17] Y. Kim. "Convolutional neural networks for sentence classification," *arXiv preprint arXiv: 1408.5882*,2014.
[18] N. Kalchbrenner, E. Grefenstette and P. Blunsom. "A convolutional neural network for modelling sentences," *arXiv preprint arXiv: 1404.2188*,2014.
[19] S. Liao, J. Wang, R. Yu, K. Sato and Z. Cheng. "CNN for situations understanding based on sentiment analysis of twitter data," *Procedia computer science*, vol. 111, pp. 376-381, 2017.
[20] Y. Zhang and B. Wallace. "A sensitivity analysis of (and practitioners guide to) convolutional neural networks for sentence classification," *arXiv preprint arXiv: 1510.03820*,2015.
[21] F. Chollet, *Keras*, GitHub repository, https://github.com/fchollet/keras, 2015.
[22] P. Chikersal, S. Poria and E. Cambria. "SeNTU: sentiment analysis of tweets by combining a rule-based classifier with supervised learning," *Proc. 9th International Workshop on Semantic Evaluation (SemEval 2015)* pp. 647-651, 2015.
[23] H. Zolkepli. *Malaya, Natural-Language-Toolkit library for bahasa Malaysia, powered by Deep Learning Tensorflow*. GitHub repository, https://github.com/huseinzol05/malaya, 2018.
[24] T. A. Le, D. Moeljadi, Y. Miura and T. Ohkuma. "Sentiment Analysis for Low Resource Languages: A Study on Informal Indonesian Tweets," *Proc. of the 12th Workshop on Asian Language Resources (ALR12)*, pp. 123-131, 2016.
[25] 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," *Journal of Physics: Conference Series. IOP Publishing.* vol. 971, 2018.

BIOGRAPHIES OF AUTHORS

Ong Jun Ying received his M.Eng (2019) in Computer and Microelectronics System from Universiti Teknologi Malaysia. His research work is mainly on machine learning, artificial intelligence and Natural Language Processing. He joined Intel Corp in 2017 and currently a System On Chip Design Engineer.

Muhammad Mun'im Ahmad Zabidi rececived his MS in Computer Engineering (1988) from the University of Bridgeport, USA. His research work is mainly on artificial intelligence, digital signal processing and embedded systems design. He is currently a Senior Lecturer at Universiti Teknologi Malaysia

Norhafizah Ramli received her MEE degree in Electrical-Electronics Engineering from Universiti Teknologi Malaysia in 1992. Her research interest is mainly in Biomedical and Digital Electronics. She is currently a Senior Lecturer at Universiti Teknologi Malaysia

Usman Ullah Sheikh received his PhD degree (2009) in image processing and computer vision from Universiti Teknologi Malaysia. His research work is mainly on computer vision, machine learning and embedded systems design. He is currently a Senior Lecturer at Universiti Teknologi Malaysia.