Accredited by National Journal Accreditation (ARJUNA) Managed by
Ministry of Research, Technology, and Higher Education, Republic Indonesia with Second Grade (Peringkat 2, Sinta 2)
since year 2017 to 2021 according to the decree No. 10/E/KPT/2019.

Published online on the journal's webpage: **http://jurnal.iaii.or.id**

RESTI JOURNAL
(System Engineering and Information Technology)
Vol. 5 No. 5 (2021) 937-842 ISSN Electronic Media: 2580-0760

Implementation Word2Vec for Feature Expansion in Twitter Sentiment Analysis

Naufal Adi N[1], Erwin Budi Setiawan[2]
[1,2]Informatic, School of Computing, Telkom University
[1]naufaladin@student.telkomuniversity.ac.id, [2]erwinbudisetiawan@telkomuniversity.ac.id

Abstract

Twitter is a microblog-based social media site launched on July 13, 2006. In March 2020, 476.696 tweets about the government policy in COVID-19 spread on Twitter were captured by the Institute for Development of Economics and Finance (Indef). Government policy has a standard meaning, namely a decision systematically made by the government with specific goals and objectives relating to the public interest, whether carried out directly or indirectly. Sentiment analysis analyzes people's opinions, sentiments, evaluations, attitudes, and emotions from written language. In this decade, Sentiment Analysis is has become a trendy research area. The purpose of this paper is to focus how to implement word2vec using similarity word as a feature expansion for minimize the vocabulary mismatch in Twitter Sentiment Analysis using "word embeddings". This research contains 11.395 tweets for a dataset, where the dataset will be used in two classifications: Support Vector Machine Algorithm and Artificial Neural Network Algorithm. The output of Word2Vec will be used for feature expansion in this research, where the algorithm of expansion will check in each row in the corpus where has a similarity vector with that word and will replace the word with the similarity of this words if the value is 0. The dataset in Feature Expansion is using 142.545 articles from Indonesian media. The result of this research is ANN is better than SVM, where the ANN without feature expansion gets 68.89 % and using feature expansion gets 72.58 %. For SVM, the final accuracy without feature expansion is 63.95 %, and using feature expansion gets 68.56 %. This research proves that feature expansion can improve the final accuracy.

Keywords: Sentiment Analysis, SVM, ANN, Word2Vec, TF-IDF

1. Introduction

Many information is generated through social internet networks with various social media formats in an era like this. When an event occurs, many people will discuss it through social media. Many things happen on social media; they search or discuss the news as a routine that must be done every day. From everything that someone discusses on social media, we can determine the value of the discussion.

Twitter is a microblog-based social media site launched on July 13, 2006 [1]. In March 2020, 476.696 tweets about the government policy in COVID-19 spread on Twitter were captured by the Institute for Development of Economics and Finance (Indef). Government policy has a standard meaning, namely a decision systematically made by the government with specific goals and objectives relating to the public interest, whether carried out directly or indirectly [2]. Twitter allows people to share information in real-time based on their experience or feeling by limiting your thoughts and information to 280 characters, exceptionally brief, not continuously linguistically redress, and employing a parcel of word variations. The point is to be able to maximally trade by using as small characters as possible [3]. Be that as it may, these constraints cause clients to utilize huge sums of "noise" like emoticons, numerous abbreviations [4], abbreviated terms, incorrectly spelled words [3], and web slang words [5] in arrange to compress more data. Tweets are frequently abbreviated and so difficult to understand. The utilize of word varieties increment the probability of lexicon jumble and make the tweets troublesome to understand without a few kinds of context [5].

One research study shows that the results compared to 3 algorithm classifiers on topic classification: Support Vector Machine, Logistic Regression, and Naive Bayes. Using Support Vector Machine is 54 % using Naive Bayes is 52 %, and using Logistic Regression is 58 % [6]. In this paper the author also using word2vec as as a feature expansion and the result of this paper, feature expansion can increase the final accuracy. For algorithm SVM feature expansion using word2vec can increase

Accepted by editor: 24-07-2021 | Final Revision: 03-09-2021 | Online Publication: 25-10-2021

0.13 %, for Naive Bayes can increase 0.21 % and for Logistic Regression can increase 0.17 % [6]. To our knowledge, word2vec has not been explored in sentiment analaysis in Indonesian language tweets.

The purpose of this paper is to focus how to implement word2vec using similarity word as a feature expansion for minimize the vocabulary mismatch in Twitter Sentiment Analysis using "word embeddings".

The rest of the paper is organized as follows. Section 2 desribes of the research method of sentiment analysis on Twitter. Section 3 descibes provides the results and discuccion of experiments and for the conclusion in Section 4.

2. Research Method

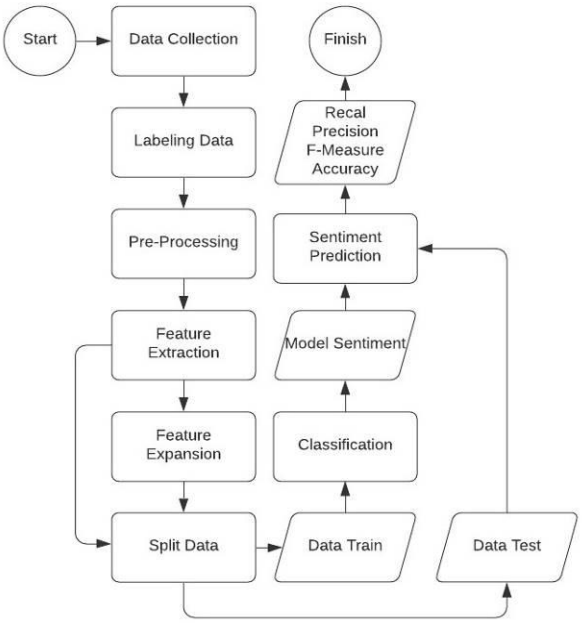

Figure 1. System Architecture

2.1. Sentiment Analysis

In this decade, Sentiment Analysis is has become a trendy research area. People share their daily life photos, status, and message on social media platforms like Twitter, Facebook, and Instagram [7]. Sentiment Analysis is analyzing people's opinions, sentiments, evaluations, attitudes, and emotions from written language [7]. Sentiment Analysis, is a subfield of Natural Language Processing (NLP), where Sentiment Analysis extracts personal information in text documents [8]. For the data, we can use from someone's post on social media. We can take the data automatically or manually [9].

2.2. Data Collection

In this research, the author retrieves data in the form of tweets from Twitter. In collecting data from Twitter, the author using automatic data crawling with a keyword. Crawling data is a process that was collecting data from the internet, either taking data on a small scale or a large scale. In this research author using Twitter API (Application Program Interface), which has been provided by the Twitter developer as a link from the system to Twitter so that the data can be retrieved and processed [9]. In collecting data, the author will use hashtags relating to government policy. Government policy has a standard meaning, namely a decision systematically made by the government with specific goals and objectives relating to the public interest, whether carried out directly or indirectly. Data that will use for Feature Extraction from tweets are 11.395 tweets about government policy. The author crawls the data using six keywords. Table 1 describes keywords and the total data that used in this research. Data for Feature Expansion uses data from Indonesian Articles where articles are taken from media in Indonesia like Kompas, CNN Indonesia, Tempo, Republika, Liputan 6, Detik.com, and Koran Sindo. Total articles for Feature Expansion are 142.545 articles.

Table 1. Tweet Data

Tweet	Total
#BenihLobster	1.315
#Covid-19	1.688
Indonesia Menyerah	1.303
Mosi Tidak Percaya	1.350
#OmnibusLaw	2.238
#PSBB	3.501

2.3. Labeling Data

To get a maximum result, data must be labeled with the correct value in the classification method. In the classification method, there has a function to differentiate the class of data. Training data is a set of data used to fit the parameters in the model [10]. Testing Data is a set of data used to evaluate a final model fit in training data. The dataset is manually labeled with 13 people, which contains two classes, positive and negative. Specifically total of the data in each class, shown in Table 3. Table 2 shows an example of class tweet after process in Sentiment Analysis. In this problem, the author assumes two classes there is positive class and a negative class

Table 2. Example of Tweet Class

Tweet	Sentiment
Kita sedang membela rakyat, bukan kepentingan kelompok saja. A Toh ^ A juga kita sama-sama rakyat lantas kenapa harus saling memberi tangis ?	Positive
Kebijakan pemerintah tentang ekspor benih lobster adalah kebijakan yang sangat merugikan negara dan rakyat	Negative

DOI: https://doi.org/10.29207/resti.v5i5.3325

Table 3. Labeling Distribution

Label	Total
Positive	5.770
Negative	5.625

2.4. Pre-Processing

The data obtained is unstructured data or is data that is not good if we use this data in an evaluation model. In the pre-processing process, unstructured data is changed into structured data. In this research author using four steps in pre-processing step.

Figure 2. Pre-Processing Workflow

(1) Case Folding: In the Case of Folding, all special characters in data will be removed (delimiter), and all letters on data will be converted into lowercase. (2) Tokenizing: Tokenizing is a process for separate words from space that will be used in Bag Of Words (BoW). In this process author using the word_tokenize NLTK library in Python. (3) Stopword Removal : This is a step for removing unimportant words like conjunction, disjunction, and preposition. (4) Lemmalization: Lemmatization is changing the words to be a basic word, or we can know it is stemming. In this process author using Python Library named is Sastrawi.

2.5. Word2Vec

Word2Vec is a feature expansion bag of word and skip-gram architectures that aims to process data that is not numeric and then processed into numeric data. Since the publishing of Word2Vec by Mikolov for word embedding, there are so many people used as features for several text classification tasks. Word2Vec constructs a vocabulary from training data and learns in vector representation. There are three layers in Word2Vec, input layer, researching (hidden layer), and output layer. There are two types of Word2Vec: Skip Gram and Continuous Bag of Word (CBOW).

The skip-gram model takes words as input and aims to predict the target context. Continuous Bag of Word is the opposite of Skip-gram, where the purpose is to predict the output when giving context from the input. Cosine similarities calculate the difference between vectors based on the cosine of the angle.

The algorithm will learn from the statistics from the resulting number of times each pairing shows up. To maximize of the likelihood in context words will be given to the center word and try to calculate the maximize of probability [11]. For calculated, the likelihood can be represented using this equation:

$$(\theta) = \prod_{t=1}^{T} \prod_{-m \le j \le m j \ne 0} P(W_{t+j} | W_t ; \theta) \quad (1)$$

In the first step, we must know where word wants to be a target, and then we declare the window. For each position t = 1, ... until T (corpus), predict the target words within a window of fixed size m, and given by wj in equation [12]. For calculation probability of context, a word can be represented using this equation:

$$P(W_o | W_i) = \frac{\exp(u_o^T v_c)}{\sum_{w=1}^{v} \exp(u_w^T v_c)} \quad (2)$$

V is a vector representation of the center word, and U represents the context word. For o is a given by central target word, and c is being obtained using the softmax function. The softmax function converts its probability by normalizing it over the whole vocabulary.

The output of Word2Vec is similar to words. Table 3 will show the example output of Word2Vec, where the target word is" Partai."

Table 4. Similar Words Of Partai

Rank	Word	Similarity (%)
1	Parpol	69.2
2	PDIP	65.9
3	PPP	61.2
4	PAN	60.4
5	PKS	58.1
6	PKB	58.0
7	Kader	57.4
8	KMP	56.5
9	PKPI	55.2
10	Parta	55.1

Expansion Algorithm

```
Input: dataset
Output: dataset with expansion
for word in dataset do
   if row[word] == 0
      x                                    =
mst_similar(positive=[word])[:rank]
      x = list(x)
      if list(x) > 0
        row[word] = 1
      else
        row[word] = 1
      end if
    end if
end for
```

The output of Word2Vec will be used for feature expansion in the next step, where the algorithm of expansion will check in each row in the corpus where has a similarity vector with that word. An example given a tweet "Partai tersebut memenangkan pemilu". Suppose the value of "Partai" is zero, the feature expansion algorithm will check corpus similarity. When checking in the corpus and has similarity, we choose to rank no 1 in similarity, then the feature value of "Partai" in the tweet representation is assigned to 1.

2.6. TF-IDF

Term Frequency – Inverse Document Frequency (TF-IDF) is a machine learning method for Natural Language

DOI: https://doi.org/10.29207/resti.v5i5.3325

Processing (NLP) where reflects how important words or documents are in a collection or corpus [13]. Common words in one or a group of smaller documents tend to have higher TF-IDF numbers than words in general like articles. The frequency with which words will appear in the document shows how important the words are in the document. The number of times a document contains this word indicates how common it is the word. The word's weight gets a more significant value if it often appears in a document and gets smaller if it appears in multiple documents [14]. Term Frequency is a frequency of occurrence of the term of i in document j divided by absolute terms in document j can be represented using this equation:

$$tfij = \frac{f_d(i)}{\max f_d(j)_{j \in d}} \quad (3)$$

For Inverse Document Frequency (IDF) has a function to reduce the weight of a term if they appear widely scattered throughout the document, can be represented using this equation:

$$idf(t, D) = \log\left(\frac{N}{df(t)+1}\right) \quad (4)$$

Where df(t) is a comprehensive document that has term t, add 1 to avoid dividing by 0 if the df(t) value is not present in the corpus. If the value of Term Frequency and Inverse Document Frequency is already, the next step is calculating the Term Frequency – Inverse Document Frequency (TF-IDF) using this equation:

$$TF - IDF = tf_{ij} * idf(t, D) \quad (5)$$

The result of TF-IDF can be represented by vector data in the form of a sparse matrix with dimension (n samples, n feature), where n feature is the value of TF-IDF and n samples is a comprehensive document.

2.7. Artificial Neural Network

Artificial Neural Network (ANN) is part of deep learning, where deep learning is part of machine learning. Artificial Neural Network (ANN) is created through inspiration from the human biological brain, consisting of 60 trillion interconnected neurons [15]. Artificial Neural Network has four main steps: initialization, activation, weight training, and iteration in a classification task. Artificial Neural Network has three layers, input layer, hidden layer (can be more than 1), output layer (multiple layer perceptron).

Every information element in a neural network begin with a neuron connected by a link, every link has a weight, and every neuron consists of more than one weight. From the input neuron until output, the links will move to other nodes, known as feed-forward (FF) neural networks. Back Propagation (BP) is an error correction or backward process for checking the fault from the output layer and check in to the hidden layer [16].

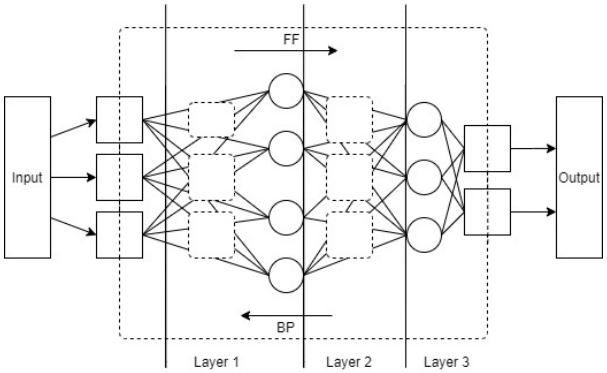

Figure 3. ANN Architecture

2.8. Support Vectore Machine

Support Vector Machine: Vapnik introduced Support Vector Machine (SVM). After that, it became the classification method often used because of the high number of enthusiasts who used it [17]. Support Vector Machine is part of a supervised learning method for classification and regression. Support Vector Machine has a high performance in producing accuracy figures between other classification algorithms based on other studies [18]. The Support Vector Machine's primary function is to check the hyperplane between two classes because there can be many hyperplanes. However, the objective is to find a hyperplane with the highest margin, which means the maximum distance between two classes.

$$\frac{\overrightarrow{w}.\vec{x} + b}{||\vec{w}||} \geq a \quad (6)$$

Kernel Trick or Generalized dot product is calculating dot product of two vector to check how much this can make an effect on each other. Kernel functions also used to get dot products to solve SVM constrained optimization.

2.9. Evaluation Performance

In this research, the author will compare the Support Vector Machine (SVM) Algorithm and Artificial Neural Network (ANN) Algorithm based on four components. F-Measure, Precision, Recall and Accuracy is the component to judge in terms of performance, which of the two algorithms is better.

Table 5. Confusion Matrix

	Class = Yes	Class = No
Class = Yes	True Positive	False Negative
Class = No	False Positive	True Negative

(1) True Positive (TP): is a positive value based correctly predicted, where the value in class is yes and the result of predict is also yes. (2) True Negative (TN): is negative value based correctly predicted, where the value in class

DOI: https://doi.org/10.29207/resti.v5i5.3325

is no and the result of predict is also no. (3) False Positive (FP): is a positive value based correctly predicted, where the value in class is no and the result of predict is yes. (4) False Negative (FN): is a negative value based correctly predicted, where the value in class is no and the result of predict is also no. (5) Recall is correct positive observation in all observation, where in actual class the prediction is positive.

$$Recall = \frac{TP}{TP + FN} \quad (7)$$

F-Measure is a mean between precision and recall, score of f- measure is for balance between precision and recall. The important of f-measure is to optimize the system to favors precision to recall, where if one of them has more positive influence in the result.

$$FM = \frac{2 * Precision * Recall}{Precision + Recall} \quad (8)$$

Precision is correct positive prediction in all positive observation.

$$Precision = \frac{TP}{TP + FP} \quad (9)$$

Accuracy is the result of classification with true data divided by total true and false data, be careful where the data not good or skewed data. Labelling data can improve result accuracy.

$$Accuracy = \frac{TP + TN}{TP + FP + TN + FN} \quad (10)$$

3. Result and Discussion

The objective of this experiment it was find the accuracy in classification method using word2vec toward various data sources. The classification accuracy is characterized as the rate of accurately classified instances utilizing 10-fold cross approval. The tests were changed by utilizing 1, 5, and 10-top highlights for each subject during highlight choice.

This research was carried out by three scenarios in each classification method. The first scenario used SVM and ANN for each method as the baseline, where basline column describe the results without performing feature expansion and tf-idf method. The second scenario will used tf-idf for weighting word and will combine with basline. The last scenario is combine basline, tfidf and word2vec as feature expansion.

3.1. Classification SVM and ANN

Table 6. Performance of SVM (Basline)

Feature	Precision	Recall	FM	Accuracy (%)
1	62.1	74.3	67.2	63.76
5	62.3	74.4	67.3	63.83
10	62.4	74.5	67.5	63.95

Table 6 shows the performance of SVM as a basline in classification method. The highest accuracy is in Top 10, where SVM can get 63.95 %.

Table 7. Performance of SVM (Basline)

Feature	Precision	Recall	FM	Accuracy (%)
1	64.4	70.2	66.9	65.66
5	64.7	70.3	66.9	65.82
10	64.9	70.4	66.8	65.97

The performance of SVM as basline and combine with tf-idf shown in Table 7. The highest accuracy is in Top 10, where SVM combine TFIDF can get 65.97 %. The accuracy was improve, before combine with tfidf the highest accuracy is 63.95 after combine improve 2.02 %.

Table 8. Performance SVM (Basline) + TFIDF + Feature Expansion

Feature	Precision	Recall	FM	Accuracy (%)
1	72.3	60.5	66.0	68.43
5	72.5	60.6	66.1	68.50
10	72.8	60.7	66.2	68.56

The performance of SVM as basline and combine with tf-idf and combine again using feature expansion shown in Table 8. The highest accuracy is in Top 10, where SVM combine tfidf can get 68.56 %. The accuracy was improve, before combine with feature expansion the highest accuracy is 65.97 after combine improve 2.59 %.

Table 9 shows the performance of ANN as a basline in classification method. The highest accuracy is in Top 10, where SVM can get 68.89 %.

Table 9. Performance of ANN (Basline)

Feature	Precision	Recall	FM	Accuracy (%)
1	70.5	65.8	67.8	68.82
5	70.7	65.9	67.9	68.85
10	70.9	66.0	68.0	68.89

The performance of ANN as basline and combine with tf-idf shown in Table 7. The highest accuracy is in Top 10, where ANN combine tfidf can get 70.69 %. The accuracy was improve, before combine with TFIDF the highest accuracy is 68.89 after combine improve 1.8 %.

Table 10. Performance of ANN (Basline) + TFIDF

Feature	Precision	Recall	FM	Accuracy (%)
1	75.6	63.7	68.0	69.92
5	75.8	63.8	68.1	70.15
10	76.0	63.9	68.1	70.69

The performance of ANN as basline and combine with tf-idf and combine again using feature expansion shown in Table 8. The highest accuracy is in Top 10, where ANN combine tfidf can get 72.58 %. The accuracy was

DOI: https://doi.org/10.29207/resti.v5i5.3325

improve, before combine with feature expansion the highest accuracy is 70.69 after combine improve 1.89 %.

Table 11. Performance of ANN (Basline) + TFIDF + Feature Expansion

Feature	Precision	Recall	FM	Accuracy (%)
1	80.5	59.0	68.1	72.45
5	80.7	59.1	68.2	72.52
10	80.9	59.2	68.3	72.58

3.2. Discussion

Table 12. Comparasion of Experiment's Results

Conditions	Accuracy (%)
SVM	63.95
SVM + TFIDF	65.97 (+ 2.02)
SVM + TFIDF + Feature Expansion	68.56 (+ 2.59)
ANN	68.89
ANN + TFIDF	70.69 (+ 1.8)
ANN + TFIDF + Feature Expansion	72.58 (+ 1.89)

Based oh these experiment's results on Table 6-11, the implementation of tfidf as weighting word and implementation of word2vec as feature expansion can improve the final accuracy. In Table 12 show the final result for each method tfidf it's important, where Term Frequency – Inverse Document Frequency (TF-IDF) as wighting word reflects how important words or documents are in a collection or corpus [17]. For word2vec as a feature expansion it's also important because the highest improve accuracy is when adding word2vec as a feature expansion, because the algorithm will learn from the statistics from the resulting number of times each pairing shows up. To maximize of the likelihood in context words will be given to the center word and try to calculate the maximize of probability [15].

4. Conclusion

This research has purpose to implement the word2vec as a feature expansion in Sentiment Analysis.This research uses dataset from twitter, where the author gets 11.395 tweets for dataset, and 142.545 articles from Indonesian media for data in feature expansion. We apply the experiment using Support Vector Machine (SVM) algorithm and Artificial Neural Network (ANN) algorithm. Based the final result ANN algorithm always among the top performer, whether to used word2vec as a feature expansion or not. Applaying tfidf as weighting word also help improve the accuracy, it's be good when we combine the tfidf and word2vec as a feature expansion.

The limitation of this research is the dataset size of Twitter is still small, this research only gets 11.395 of tweets. Therefore, for the future research can improve the performance by the largest of dataset, and also can applay the different methods in classification method.

References

[1] A. Fauzi, E. B. Setiawan, and Z. K. A. Baizal, "Hoax News Detection on Twitter using Term Frequency Inverse Document Frequency and Support Vector Machine Method," 2019, doi: 10.1088/1742-6596/1192/1/012025.

[2] Herabudin, "Studi kebijakan pemerintah dari filosofi ke implementasi," *Pustaka Setia: Bandung*, 2014.

[3] W. Wu, B. Zhang, dan M. Ostendorf, "Automatic Generation of Personalized Annotation Tags for Twitter Users," Comput. Linguist., no. June, pp. 689–692, 2010.

[4] S. Mukherjee, A. Malu, a R. Balamurali, dan P. Bhattacharyya, "TwiSent : A Multistage System for Analyzing Sentiment," Cikm 12, pp. 2531–2534, 2012.

[5] M. A. Zingla, L. Chiraz, Y. Slimani, C. Berrut, M. A. Zingla, L. Chiraz, Y. Slimani, dan C. B. Statistical, "Statistical and Semantic Approaches for Tweet Contextualization To cite this version : Statistical and Semantic Approaches for Tweet Contextualization," Proceeding 19th Int. Conf. Knowl. Based Intell. Inf. Eng. Syst., vol. 60, pp. 498 – 507, 2015.

[6] Setiawan, Erwin B., Dwi H. Widyantoro, and Kridanto Surendro. "Feature expansion using word embedding for tweet topic classification." 2016 10th International Conference on Telecommunication Systems Services and Applications (TSSA). IEEE, 2016.

[7] B. Liu, "Sentiment analysis and opinion mining," *Synth. Lect. Hum. Lang. Technol.*, 2012, doi: 10.2200/S00416ED1V01Y201204HLT016.

[8] S. Symeonidis, D. Effrosynidis, and A. Arampatzis, "A comparative evaluation of pre-processing techniques and their interactions for twitter sentiment analysis," *Expert Syst. Appl.*, 2018, doi: 10.1016/j.eswa.2018.06.022.

[9] J. Eka Sembodo, E. Budi Setiawan, and Z. Abdurahman Baizal, "Data Crawling Otomatis pada Twitter," 2016, doi: 10.21108/indosc.2016.111.

[10] "5th International Conference on Big Data Innovations and Applications, Innovate-Data 2019," Communications in Computer and Information Science. 2019.

[11] Q. Chen and M. Sokolova, "Unsupervised Sentiment Analysis of Objective Texts," 2019, doi: 10.1007/978-3-030-18305-9_45.

[12] R. Velvizhi, C. Rajabhushanam, and S. R. S. Vidhya, "Opinion mining for travel route recommendation using Social Media

[13] V. Amrizal, "Penerapan Metode Term Frequency Inverse Document Frequency (Tf-Idf) Dan Cosine Similarity Pada Sistem Temu Kembali Informasi Untuk Mengetahui Syarah Hadits Berbasis Web (Studi Kasus: Hadits Shahih BukhariMuslim)," *J. Tek. Inform.*, 2018, doi: 10.15408/jti.v11i2.8623.

[14] B. M. Yunus, M. Irfan, W. B. Zulfikar, and W. Darmalaksana, "Similarity detection for hadith of Fiqh of women using cosine similarity and boyer moore method," *Int. J. Adv. Trends Comput. Sci. Eng.*, 2020, doi: 10.30534/ijatcse/2020/11912020.

[15] M. Negnevitsky, *Artificial Intelligence, A Guide to Intelligent Systems (Second Edition)*. 2015.

[16] A. G. Farizawani, M. Puteh, Y. Marina, and A. Rivaie, "A review of artificial neural network learning rule based on multiple variant of conjugate gradient approaches," 2020, doi: 10.1088/1742-6596/1529/2/022040.

[17] V. N. Vapnik, "The nature of statistical learning theory. Statistics for Engineering and Information Science," *Springer-Verlag, New York*, 2000.

[18] K. Mouthami, K. N. Devi, and V. M. Bhaskaran, "Sentiment analysis and classification based on textual reviews," 2013, doi:10.1109/ICICES.2013.6508366.

DOI: https://doi.org/10.29207/resti.v5i5.3325