Journal of Information System Research (JOSH)
Volume 3, No. 4, Juli 2022, pp 644−650
ISSN 2686-228X (media online)
https://ejurnal.seminar-id.com/index.php/josh/
DOI 10.47065/josh.v3i4.1875

Hate Speech Detection on YouTube Using Long Short-Term Memory and Latent Dirichlet Allocation Method

Andi Fadil Adiyaksa*, Donni Richasdy, Aditya Firman Ihsan

Fakultas Informatika, Program Studi Studi Informatika, Telkom University, Bandung
Jl. Telekomunikasi No. 1, Terusan Buahbatu - Bojongsoang, Sukapura, Kec. Dayeuhkolot, Kabupaten Bandung, Jawa Barat, Indonesia
Email: [1,*]andifadiladiyaksa@student.telkomuniversity.ac.id, [2]donnir@telkomuniversity.ac.id,
[3]adityaihsan@telkomuniversity.ac.id
Email Penulis Korespondensi: andifadil.adiyaksa1@gmail.com
Submitted: **16/07/2022**; Accepted: **28/07/2022**; Published: **31/07/2022**

Abstrak−Media sosial youtube salah satu media populer untuk semua kalangan untuk menjadi platform sebagai sarana informasi dan menyampaikan pendapat. Pendapat dapat dikategorikan kebencian jika pendapat tersebut menyerang sesuatu yang ditergetkan. Ujaran kebencian merupakan suatu perilaku, perkataan ataupun Tindakan yang dilarang, karena mengakibatkan terjadinya kekerasan kepada setiap individua dan kelompok. Mengekspresikan opini dalam bentuk ujaran kebencian merupakan masalah yang masih sangat sulit diatasi oleh pihak berwenang karena sudah sangat umum terjadi. Oleh sebab itu, penelitian ini dibangun sistem untuk mendeteksi ujaran kebencian pada kolom komentar youtube, dengan menggunakan metode *Long Short-Term Memory* dan *Latent Dirichlet Allocation*. Pada penelitian ini dilakukan beberapa metode yang bertujuan untuk mendapatkan nilai akurasi terbaik dan melakukan proses topic modeling menggunakan *Latent Dirichlet Allocation* menghasilkan total tiga topik yang berisi kata-kata yang sering muncul pada komentar youtube. Berdasarkan pengujian yang dilakukan diperoleh nilai akurasi terbaik sebesar 0,657 atau 66%.

Kata Kunci: Ujaran Kebencian; YouTube; Topic Modeling; Long Short-Term Memory; Latent Dirichlet Allocation

Abstract−YouTube social media is one of the popular media for all people to become a platform as a means of information and expressing opinions. Opinions can be categorized as hate if they attack something targeted. Hate speech is a behavior, word or action that is prohibited, because it causes violence to any individual and group. Expressing opinions in the form of hate speech is a problem that is still very difficult for the authorities to overcome because it is very common. Therefore, in this study a system was created to detect hate speech in the youtube comment column, using the Long Short-Term Memory and Latent Dirichlet Allocation. In this study, several methods were carried out that aimed to get the best accuracy value and carried out the topic modeling process using Latent Dirichlet Allocation to produce a total of three topics containing words that often appear in youtube comments. Based on the tests that have been obtained, the best accuracy is 0.657 or 66%.

Keywords: Hate Speech; YoutTube; Topic Modeling; Long Short-Term Memory; Latent Dirichlet Allocation

1. INTRODUCTION

Social media is not common among the general public, both adults, teenagers, and children[1].Social media users can do anything, such as communicate, give opinions, or communicate with friends or other people. On social media, the expression of the views or opinions on information in the form of images or videos is free, including views that contain hate speech[2]. In recent years, hate speech has become a more prevalent criminal offense, particularly on social media. By focusing on a person or group of people's conscious or unconscious fundamental features, it can cause harm to them[3]. The social media used is the youtube platform by taking comment data from one of the video content creators in Indonesia. To find out whether the video contains opinions or opinions that include hate speech, classification or grouping is needed to find out a text pattern from the youtube video comment.

There have been several studies related to the detection of hate speech that has been carried out previously. One of them is a study conducted by Elvira Erizal[4] entitled Detection of Hate Speech in Indonesian in the Instagram Comments Column with the Maximum Entropy Classification Method. 88.68% recall, and F1 score 88.62%. This study concludes that the iteration value affects system performance. The more iterations run, the better the system performance until the best parameter is obtained. Then the commission will not change in value.

In a study conducted by Bagas Prakoso Putra[5] with the title Detection of Hate Speech Using the Convolutional Neural Network Algorithm on the Image. From this research, the data classification process has been carried out using the Convolutional Neural Network (CNN) algorithm by getting an average precision, recall, accuracy 99.46%, 97.99%, 99.8%. In this study, it is suggested to be able to add mobile application integration so that users are not only focused on the web but also on their respective smartphone devices so that it is easy to take data through photos or screenshots and can add hate speech recognition through a visual image directly without text intermediaries.

In a study conducted by Doni Riyanta[6] with the title Detection of Hate Speech on Twitter with Feature Expansion Using Fast text. This research has built a hate speech detection system using Feature Expansion with the Fast text method to create a corpus from IndoNews data and tweet data, which aims to overcome the problem of vocabulary mismatches in tweets and perform similarity retrieval Top 1, Top 5, and Top 10. The classification method used is Random Forest and Support Vector Machine (SVM). From the results of the research above, it can be concluded that weighting using TF-IDF and combined with Feature Expansion can increase the accuracy value

Journal of Information System Research (JOSH)
Volume 3, No. 4, Juli 2022, pp 644−650
ISSN 2686-228X (media online)
https://ejurnal.seminar-id.com/index.php/josh/
DOI 10.47065/josh.v3i4.1875

and F1-Score in the Random Forest classification with an accuracy value of 99.92%, which increases by 13.375% and the F1-Score value of 0.9992 which increased by 0.1343 in the corpus tweet in the Top 10 similarity. While in the Support Vector Machine (SVM) classification, the accuracy value obtained was 87.91% which increased by 0.457%, and for the F1-Score value of 0.8789, which increased by 0.0046 in the corpus tweet at similarity 10.

In a study conducted by Baidya Nath Saha[7] with the title LSTM based Deep RNN Architecture for Hate Speech and Offensive Content (HASOC) Identification in Indo-European Languages. In this study, the method from the LSTM base is using RNN using data from twitter and facebook with training data corpus for English and Hindi. The RNN base method from LSTM gets an accuracy of 0.50 or 50% for English and 0.54 or 50% for hindi.

In a study conducted by Yullia Diah Pitaloka[8] with the title Detection of Hate Speech Using the Word2vec Algorithm and DBN. The Deep Belief Network technique and Word2Vec feature extraction with improved accuracy before classification are used in this study. It is intended that by creating this program, the computer would be able to recognize hate speech in the text on the image. The deep learning network (DBN) technique is used in the system for categorizing hate speech. The accuracy rate in this study's results after preprocessing was 86.46%.

Therefore, his research above aims to detect hate speech on social media on the Indonesian YouTube platform and determine the modeling topic or topics discussed in the YouTube comments column, each of these topics contains 30 words that often appear in user comments using Latent Dirichlet Allocation. (LDA). The benchmark used in this study was derived from user or person comments on YouTube videos. Long-Short-Term Memory is the study methodology employed (LSTM). A system known as LSTM uses data that has been kept for a considerable amount of time and is intended to forecast, process, and categorize. One of the most recent variations on the Recurrent Neural Network (RNN) approach is this one. Because the LSTM's structure is more sophisticated to store and categorize information that is held for a long time, this approach is superior to increase performance and seems to address the RNN method's weaknesses, including its inability to predict words based on information maintained for a long period. an RNN against a system. This study aims to analyze the Latent Dirichlet Allocation (LDA) to determine a modeling topic that represents a document or text and Long Short-Term Memory (LSTM) method to resulting performance. The Confusion Matrix algorithms aim to measure performance for machine learning classification problems where the output can be in the form of two or more classes. Confusion Matrix is a table with four different combinations of predicted and actual values, with the hope of getting better deals than existing research.

2. RESEARCH METHODOLOGY

2.1 Research Flow

This study, there are several stages to creating a hate speech detection system on the Indonesian youtube platform. In making this classical system, there are several stages consisting of retrieval of data stored in the dataset, preprocessing, data separation, feature extraction, modeling, and evaluation. The system flowchart to be made can be seen in Figure 1.

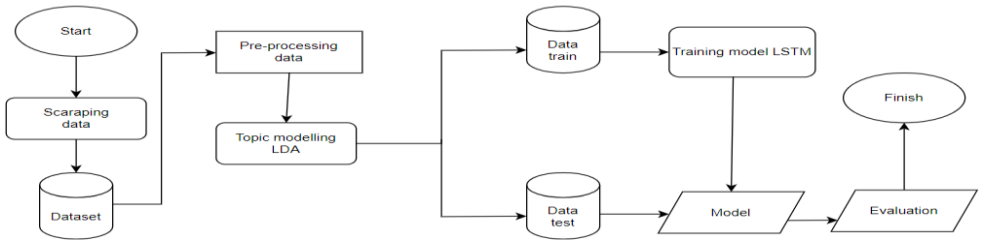

Figure 1. Stages of the system to be built

2.2 Dataset

The data used is a youtube video commentary from one of the content creators of Indonesia, Diva Studio, using data scraping or data extraction. Data scraping is an automation technology that allows someone to extract data from a website, database, enterprise application, or legacy system and then save it into a file in the form of a spreadsheet or excel[9]. The data collection also uses an API (Application Programming Interface) which is owned by several websites or applications, one of which is youtube, so that it allows someone to access data in a structured data format[10]. The data obtained from the scraping data amounted to 1056 youtube comment data, where each comment has a label. Label 0 indicates that the text is a category of hate speech, while Label 1 suggests that the text is a category of non-hate speech. In this category, there are 648 labeled 0 and 408 labeled 1.

2.3 Preprocessing

Preprocessing is the process of getting unstructured data from data capture suitable for processing[11].preprocessing stage, it aims to eliminate or clean words in text classification. This stage will change

Journal of Information System Research (JOSH)
Volume 3, No. 4, Juli 2022, pp 644−650
ISSN 2686-228X (media online)
https://ejurnal.seminar-id.com/index.php/josh/
DOI 10.47065/josh.v3i4.1875

the data so that it becomes better data which will later produce information in the form of text with good quality so that it is ready to be used in the following process. In this study, the preprocessing stages used were cleaning, case folding, tokenizing, and word change. Cleaning is removing punctuation marks, emoticons, and numbers from the text. Case folding is converting uppercase letters into lowercase letters in text. Tokenize is the process of cutting sentences from the previous stage based on each constituent word. Changing words is the process of changing words that has no meaning or a typo to have sense. An example of the preprocessing process starting from raw data, which will produce better data, is shown in Table 1.

Table 1. Preprocessing Process

Process Name	Input	Output
Cleaning	Astaga lihatlah, muka lu jalek BANGET0.	Astaga lihatlah muka lu jalek BANGET0
Case folding	Astaga lihatlah muka lu jalek BANGET0	astaga lihatlah muka lu jalek banget
Change Word	astaga lihat muka lu jalek banget	astaga lihat muka kamu jelek banget
Tokenizing	astaga lihat muka kamu jelek banget	[astaga,lihat,muka,kamu,jelek,banget]

2.4 LDA (Latent Direchlet Allocation)

LDA is a modeling topic technique that represents a document or text as a mixture of topics with a certain probability[12]. LDA refers to anything that is hidden in the data[13]. Topic modeling with LDA to find the topic being discussed in each document or sentence so that can use it to classify, summarize, and evaluate the similarities and relationships of the issues contained in a collection of documents or texts[14]. In LDA form, each produced topic is randomly combined to describe the document or text, and the subject is formed from the word distribution. The LDA's form, as shown in the picture below.

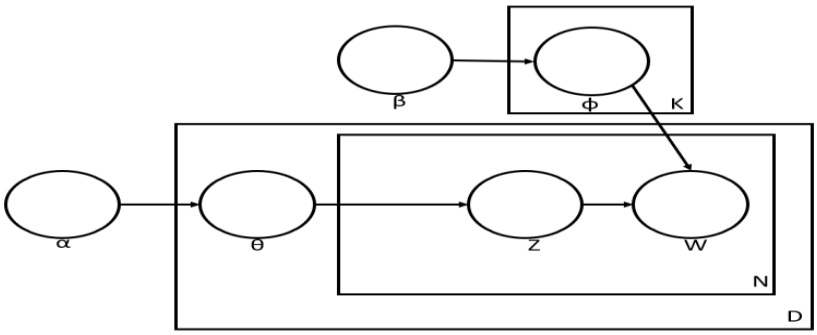

Figure 2. Illustration LDA Method

According to the LDA illustration in Figure 1, there have 2 parameters in the method LDA. Parameters α and β are inputs for the distribution of topics contained in the word set (corpus) of D document. The word set (corpus) of the D document's word distribution is determined by the inputs and. The parameter is used to control how subjects are distributed across the document; the greater the alpha value, the more topics are covered. The subject dispersed from document D is represented by the variable, the topic for word N in document D is represented by ZN, and the word under observation is represented by WN. The probability distribution of the words in the topic is represented by the variable. Determine the distribution of words on the topic by using the parameter. The topic has more words when the beta value is higher; conversely, when the beta value is lower, the topic contains fewer words.. So the topic contains a more unique set of words. In the φ variable represents the distribution of words on the topic – K[15]. In this process, the calculation of the distribution of topics in a document will be carried out using the formula (1) and the calculation of the distribution of words in a topic with the formula (2).

$$P(t_k|d_i) = \frac{n_{ik}+\alpha}{N_i+K\alpha} \quad (1)$$

$$P(tk|di) = \frac{m_{jk}+\beta}{\sum_{j\in v} m_{jk}+V\beta} \quad (2)$$

2.5 LSTM (Long Short-Term Memory)

An RNN development approach called LSTM, can categorize data in the form of time series[16]. LSTM resolves the vanishing gradient issue that arises in RNN with four gates. The presence of a route connecting the cell state Ct-1 with the cell state (Ct) is the fundamental tenet of LSTM[17]. The steps in the LSTM process:

a. Choosing which information will be erased at Ct-1 is the first stage in deciding which information will not be kept in state. Using a sigmoid, this procedure is referred to as the "forget gate." The output of this gate will be the coupled St-1 and Xt, which together will provide a value between 0 and 1. The Hadamart Product will be used to multiply this result by the cell state Ct-1. When the value is 0, the information won't be transmitted and

Journal of Information System Research (JOSH)
Volume 3, No. 4, Juli 2022, pp 644−650
ISSN 2686-228X (media online)
https://ejurnal.seminar-id.com/index.php/josh/
DOI 10.47065/josh.v3i4.1875

will be destroyed, whereas when the value is 1, the information will be continued. The formulation of the LSTM's first step is shown below.

$$f_t = \frac{\sigma}{1}(w_f \,.\, [s_{t-1}, x_t] + b_f) \tag{3}$$

The information that has to be added to cell state Ct is determined in the second stage. This phase uses the sigmoid function as the input gate and the tanh function as the intermediate gate to process the combined results of St-1 and Xt. To create the information that will be added to Ct, the outputs of the two functions are multiplied. The following is then included in the first step's forget gate's production:

$$i_t = \frac{\sigma}{1}(w_i \,.\, [s_{t-1}, x_t] + b_i) \tag{4}$$

$$C_t = \frac{tanh}{1}\,(\omega_c \,.\, [s_{t-1}, x_t] + b_c) \tag{5}$$

$$C_t = i_t * C_t + f_t * C_{t-1} \tag{6}$$

b. Finding the LSTM unit's output is the final step. The output gate, produced by combining the sigmoid calculations of St-1 and Xt, makes this output. This gate determines the size of the value of the following cell state in st. In addition, the value of the output gate is multiplied by the value of the tanh of Ct. The output of the LSTM unit is the multiplication result:

$$o_t = \sigma(w_0 \,.\, [s_{t-1}, x_t] + b_0) \tag{7}$$

$$s_t = o_t * tanh(C_t) \tag{8}$$

2.6 Evaluation

Evaluation is the last stage in this study that uses the confusion matrix. Each class is present in each dimension of the confusion matrix, which is a two-dimensional matrix. Columns usually represent the original class, while rows represent the predicted class[12]. To evaluate, use a confusion matrix to get values such as TP, TN, FP, and FN. After that, that value will be used to recall, F1 score, precision and accuracy using the following formula:

$$\text{Akurasi} = \frac{\text{TP} + \text{TN}}{\text{TP} + \text{FP} + \text{TN} + \text{FN}} \tag{9}$$

$$\text{Precision} = \frac{\text{TP}}{\text{TP} + \text{FP}} \tag{10}$$

$$\text{Recall} = \frac{\text{TP} + \text{TN}}{\text{TP} + \text{FN}} = \frac{\text{TN}}{\text{P}} \tag{11}$$

$$\text{F1} - \text{score} = \frac{2\ * (\text{Recall} * \text{Precision})}{\text{Recall} + \text{Precision}} \tag{12}$$

3. RESULTS AND DISCUSSION

Testing the text classification of Indonesian youtube video comments, there are 1056 datasets which are divided into two categories or labels, including label 0 is the text that is categorized as containing hate speech while label 1 is the text that is not classified as having hate speech. This labeling is carried out by two different people, which is done manually. The test scenario of this final project focuses on the preprocessing and classification stages. The first scenario is to test the data sharing. The goal is to find out the best performance of the data sharing. The second scenario is to do a test with classification, namely by changing the number of batch sizes or the number of data samples distributed to the Neural Network. The goal is to determine whether batch size can affect classification performance and display topics generated by LDA.

The next step is to model the topic using the LDA method, The next step is to model the topic using the LDA method, this method uses preprocessing tokenizing, to find each word that is being discussed or frequently appears in youtube comments stored in 3 the topics that can be seen in Figure 4 and the contents of those topics in Figure 5. After modeling the topic using the LDA method, the next step is to classify the LSTM model. LSTM has 3 models or gates that have their respective duties and functions, namely the forget gate functions to reduce or eliminate irrelevant information to get more actual information, after getting complete information, enter the next gate, namely the input gate according to its name on This gate is in charge of inputting information and its job is

Journal of Information System Research (JOSH)
Volume 3, No. 4, Juli 2022, pp 644−650
ISSN 2686-228X (media online)
https://ejurnal.seminar-id.com/index.php/josh/
DOI 10.47065/josh.v3i4.1875

to add information that has been selected on the previous gate, and the last stage functions to produce a complete data information at this stage is called the output gate.

3.1 Results and discussion of topics generated by LDA

In Figure 3 below are the topics generated by the LDA generated by 1056 datasets which are then manually labeled and preprocessed. In the preprocessing process, data is collected using a tokenizing technique, namely, cutting sentences into each word so that the LDA process is more effective in scanning data. The topic modeling process using LDA resulted in a total of three topics. The research will use the second topic to determine what aspects are contained in topic number two.

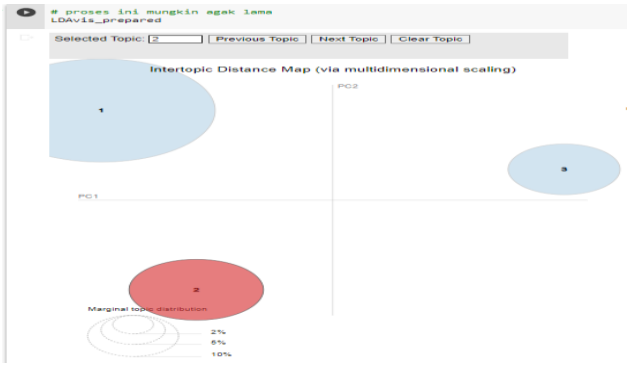

Figure 3. Illustration Determination of aspects based on the results of the LDA

In topic two, there is a collection of words, as shown in Figure 4 below. In determining the aspect, you can see from the top terms or word rankings most often discussed in the topic.

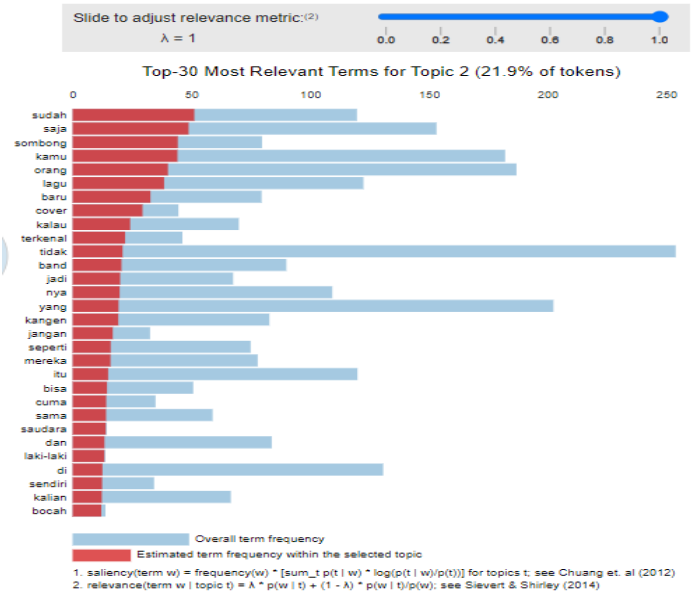

Figure 4. Illustration of a collection of words on topic 3 of the results of the LDA

3.2 Results and discussion of the effect split data

In the first scenario, testing was carried out on modeling, which focused on split data. In the split train data and test data. Training data will later be used to train the algorithm in finding the appropriate model, while testing data will be used to test and determine the model's performance obtained at the testing stage. In this test, the LSTM classification is used to determine the effect of split data on the accuracy value. The results of the first scenario in Table 2.

Table 2. Effect of Split Data

Split Data Data train: Data test	Classification Accuracy LSTM
60:40	0.575
70:30	0.540
80:20	0.622
90:10	**0.656**

Journal of Information System Research (JOSH)
Volume 3, No. 4, Juli 2022, pp 644−650
ISSN 2686-228X (media online)
https://ejurnal.seminar-id.com/index.php/josh/
DOI 10.47065/josh.v3i4.1875

Based on the results, table 2 that the distribution of 90 train data and test data 10 using the LSTM method classification gets an accuracy of 0.656 or 66%. Based on the experiment above, it can be concluded that the more data trained, the better the accuracy.

3.3 Results and discussion of the effect batch size

In the second scenario, the test was carried out using the classification technique carried out in the previous plan, namely by using the LSTM accuracy classification using 90:10 data. The quantity of data samples sent to the neural network is known as the batch size. The data distribution in this test was done in the manner described in the preceding scenario, namely 90:10 with batch size values of 16, 32, and 64. This experiment was run to find out how batch size affected the accuracy value. Table 3 provides the test findings.

Table 3. Effect of Batch Size

Batch Size	Classification Accuracy LSTM Data 90:10
16	0.547
32	0.528
64	**0.657**

Using the classification with 90:10 data, the batch size 64 obtained an accuracy of 0.657, and it can be shown that there was an increase of 0.001 based on the results of the tests performed in Table 3. According to the aforementioned experiment, the batch size influences whether the accuracy value increases or decreases.

So in the scenarios that have been carried out on the LDA method in determining topics. There are 3 topics, each of which has 30 words that are often spoken in youtube comments, and the LSTM classification method gets the best accuracy from data sharing and batch size changes of 66%.

4. CONCLUSION

Based on scenario testing that has been carried out to detect hate speech in comment data on youtube Indonesia uses LSTM and LDA, it can be concluded that the best system performance is the result of split data for data train and data test with a total data sharing of 90:10, with an entire division of 90% for data train and 10% for data test and using batches size 64. The system's performance in data sharing using the LSTM classification produces an accuracy of 0.656. With these tests, it is proven that data sharing can improve accuracy the lower the test data and the higher the classification results obtained. On the contrary, the higher the test data, the lower the classification results. This is because the spread of data on the test data produces different classification values. The use of batch size is proven that a high value of batch size affects the accuracy value, which creates an accuracy of 0.657 or 66%. This is because a high batch size is used. After all, it allows computational acceleration. For further research, more datasets are needed so that the resulting accuracy value is more effective. Because labeling is done manually, it would be preferable if specialists were brought in to decide whether a phrase contains hate speech or not. This would ensure that the data used for testing is more accurate and unbiased.

REFERENCES

[1] B. R. Amrutha and K. R. Bindu, “Detecting hate speech in tweets using different deep neural network architectures,” *2019 Int. Conf. Intell. Comput. Control Syst. ICCS 2019*, no. Iciccs, pp. 923–926, 2019, doi: 10.1109/ICCS45141.2019.9065763.

[2] S. S. Syam, B. Irawan, and C. Setianingsih, “Hate speech detection on twitter using long short-term memory (LSTM) method,” *2019 4th Int. Conf. Inf. Technol. Inf. Syst. Electr. Eng. ICITISEE 2019*, pp. 305–310, 2019, doi: 10.1109/ICITISEE48480.2019.9003992.

[3] À. A. Carracedo and R. J. Mondéjar, “Profiling Hate Speech Spreaders on Twitter,” *CEUR Workshop Proc.*, vol. 2936, pp. 1801–1807, 2021.

[4] E. Erizal, B. Irawan, and C. Setianingsih, “Hate speech detection in Indonesian language on instagram comment section using maximum entropy classification method,” *2019 Int. Conf. Inf. Commun. Technol. ICOIACT 2019*, pp. 533–538, 2019, doi: 10.1109/ICOIACT46704.2019.8938593.

[5] B. P. Putra, B. Irawan, C. Setianingsih, F. T. Elektro, U. Telkom, and D. Learning, “Convolutional Neural Network Pada Gambar Hatespeech Detection Using Convolutional Neural Network Algorithm Based on Image,” no. 3, 2019.

[6] D. Riyanta and E. B. Setiawan, “Deteksi Ujaran Kebencian pada Twitter dengan Feature Expansion Menggunakan Fasttext,” *e-Proceeding Eng.*, vol. 8, no. 6, pp. 12449–12458, 2021.

[7] B. N. Saha and A. Senapati, “Hate speech and offensive content identification: LSTM based deep learning approach @ HASOC 2020,” *CEUR Workshop Proc.*, vol. 2826, pp. 290–297, 2020.

[8] D. Pitaloka, M. Nasrun, S. Si, and C. Setianingsih, “DETEKSI UJARAN KEBENCIAN MENGGUNAKAN ALGORITMA WORD2 VEC DAN DEEP BELIEF NETWORK (DBN) DETECTION OF HATE SPEECH USING WORD2 VEC AND DEEP BELIEF NETWORK (DBN) ALGORITHM,” no. 3, 2019.

[9] N. Rochmawati, H. B. Hidayati, Y. Yamasari, H. P. A. Tjahyaningtijas, W. Yustanti, and A. Prihanto, “Analisa Learning Rate dan Batch Size pada Klasifikasi Covid Menggunakan Deep Learning dengan Optimizer Adam,” *J. Inf. Eng. Educ.*

Journal of Information System Research (JOSH)
Volume 3, No. 4, Juli 2022, pp 644−650
ISSN 2686-228X (media online)
https://ejurnal.seminar-id.com/index.php/josh/
DOI 10.47065/josh.v3i4.1875

Technol., vol. 5, no. 2, pp. 44–48, 2021, doi: 10.26740/jieet.v5n2.p44-48.

[10] H. Faris, I. Aljarah, M. Habib, and P. A. Castillo, "Hate speech detection using word embedding and deep learning in the Arabic language context," *ICPRAM 2020 - Proc. 9th Int. Conf. Pattern Recognit. Appl. Methods*, no. Icpram 2020, pp. 453–460, 2020, doi: 10.5220/0008954004530460.

[11] D. A. N. Taradhita and I. K. G. D. Putra, "Hate speech classification in Indonesian language tweets by using convolutional neural network," *J. ICT Res. Appl.*, vol. 14, no. 3, pp. 225–239, 2021, doi: 10.5614/itbj.ict.res.appl.2021.14.3.2.

[12] P. H. Saputro, M. Aristin, and Dy. L. Tyas, "Klasifikasi Lagu Daerah Indonesia Berdasarkan Lirik Menggunakan Metode Tfidf Dan Naïve Bayes," *J. Teknol. Inform. dan Terap.*, vol. 4, no. 1, pp. 45–50, 2017.

[13] S. H. Mohammed and S. Al-Augby, "LSA & LDA topic modeling classification: Comparison study on E-books," *Indones. J. Electr. Eng. Comput. Sci.*, vol. 19, no. 1, pp. 353–362, 2020, doi: 10.11591/ijeecs.v19.i1.pp353-362.

[14] Z. Wan, "What Do Programmers Discuss about Blockchain ?," *IEEE Trans. Softw. Eng.*, 2019.

[15] M. D. R Wahyudi, A. Fatwanto, U. Kiftiyani, and M. Galih Wonoseto, "Topic Modeling of Online Media News Titles during COVID-19 Emergency Response in Indonesia Using the Latent Dirichlet Allocation (LDA) Algorithm," *Telematika*, vol. 14, no. 2, pp. 101–111, 2021, doi: 10.35671/telematika.v14i2.1225.

[16] Z. F. Hu, X. T. Si, Y. Luo, S. S. Tang, and F. Jian, "Speaker recognition based on 3dcnn-lstm," *Eng. Lett.*, vol. 29, no. 2, pp. 463–470, 2021.

[17] S. Boumerdassi, R. Milocco, L. Saidane, and N. Puech, *Machine learning for networking*, vol. 77, no. 5–6. Paris, France: First International Conference, 2022.