International Journal of Innovative Computing, Information and Control
Volume **16**, Number **5**, October **2020**

ICIC International ©2020 ISSN 1349-4198
pp. 1787–1800

MINING PUBLIC OPINION ON RADICALISM IN SOCIAL MEDIA VIA SENTIMENT ANALYSIS

Ade Iriani[1], Hendry[1,*], Daniel Herman Fredy Manongga[1] and Rung-Ching Chen[2]

[1]Faculty of Information Technology
Satya Wacana Christian University
1-10 Notohamidjojo, Salatiga, Central Java 50715, Indonesia
ade.iriani@uksw.edu; *Corresponding author: hendry@uksw.edu; dmanongga@gmail.com

[2]Department of Information Management
Chaoyang University of Technology
No. 168, Jifeng East Road, Wufeng District, Taichung 413310, Taiwan
crching@cyut.edu.tw

Received May 2020; revised August 2020

Abstract. *The identification and classification of hate speech, extremism, and radicalism in social media are very important topics today because they have a wide negative impact on society. Hostile groups use this media to spread their hate speech, ideology, and recruitment of individuals. This study aims to propose a new method using deep learning to classify the utterances of hate, extremism, and radicalism in Indonesian which are posted on social media. This method uses word2vec as word embedding and the combination of Restricted Boltzmann Machine and back-propagation network as our basic classification method. The model can achieve 81.63% accuracy to predict radicalism and hate-speech. Our model outperforms the baseline classifier methods based on the comparison in experimental results.*
Keywords: Social media, Sentiment classification, Hate speech, Extremism, Radicalism, Deep learning

1. **Introduction.** Radicalism and hate speech in social media are more severe nowadays. With the weak regulation to regulate the use of social media, users freely type whatever topic they want on it. It is in contrast with the aim of social media as an instrument to facilitate communication and access to information. Many parties have tried to suppress the hate speech and radicalism on the social media, such as government, technology companies, civil organizations, and researchers [1–3].

Social media like Facebook, Twitter, Instagram, and YouTube are communication platforms that can be used for good and evil purposes. For good purposes, this media can contain positive and inspirational content. However, this media can also be used as a platform to spread fake news or potentially dangerous propaganda. Fake news can be intentionally made and spread [4] to create various threats to modern society including insecurity, chaos, or riots in local communities [5]. Through this media, various hostile groups, opposition, extremist, or jihad can increase their online presence to recruit their followers. In some cases, social media is also a place for radicalization, planning, and preparation of attacks [6, 7], for example, the influence of social media on a simple woman in Indonesia who planned a suicide bombing to the State Palace on December 11,

DOI: 10.24507/ijicic.16.05.1787

2016. This woman underwent a process of indoctrination of jihad through the Internet conducted by her future husband.

Expressions of hatred, extremism, and radicalism have enormous influence on society. For Indonesia, which has a multiethnic and religious society, issues such as ethnicity, religion, race, and intergroup pose a major threat to the integrity of Indonesian society. To reduce such problems on social media, various efforts have been made by government and non-government organizations, including the threat of severe penalties for the news disseminator. However, these efforts did not significantly reduce the amount of radical content on social media. Recent studies show the growth of forms of intolerance, extremism, and radicalism among Indonesian millennial Muslims as a result of the influence of social media [8].

Deep learning is a complex multivariate deep structure computation of machine learning [23]. Though deep learning can be characterized in several different ways, the most important is that deep learning can learn higher-order interactions among features using a cascade of many layers. The capability of deep learning to going deep training brings the breakthrough of neural network methodology to be applied in many domains of researches. Deep learning is applicable for classification problems in a specific domain of radicalism [10,13,22], user rating [23], fake news [4], and many more. In this manuscript, we use the classifier algorithm to classify the hate speech and radicalism in the Indonesian language.

One of the big challenges in dealing with such issues is determining the extent to which content on a social media site is dangerous and which content is still in the corridor of freedom of speech [9]. The method used by researchers to detect and classify extremist content and hate speech in social media is sentiment analysis. There are various variations of this method, but the baseline now is machine learning-based methods, such as, support vector machine, linear classifier, decision tree, and neural networks. The advantage of classification methods like this is the ease of implementation. However, the weakness is in the stability of configuration and parameter settings. Generalization and overfitting are derived from configuration settings. In this study we propose a new method of using deep learning to classify radicalism in Indonesian.

The remainder of the paper is organized as follows. Section 2 briefly describes related work with the previous works and previous researches contributions. Section 3 gives the problem definition and preliminary. Section 4 presents our proposed method. Section 5 describes the experimental results. We give conclusions in Section 6.

2. **Related Work.** The use of social media for malicious purposes has caught the attention of various researchers lately. This section reviews various studies using relevant sentiment analysis to classify and predict extremist and radical content on social media.

In general, sentiment classification methods can be categorized into three main categories, namely methods based on lexicon, methods based on machine learning, and mixed methods (hybrid) [10]. Lexicon methods are very dependent on the dictionary containing the tagged lexicon. Text input needs to be converted to a token to match the lexicon already in the dictionary. Whereas machine learning-based methods consist of data collection, pre-processing, training data, classification, and results. Machine learning techniques are very dependent on the selection of features. The selection of the appropriate features largely determines the accuracy of the classifier. Feature vectors can be unigram (single word phrases), bi-gram (two consecutive phrases), tri-gram (three consecutive phrases). The mixed-method (hybrid) is a combination of the two previous methods.

The use of lexicon-based methods in the analysis of radical content in online media has been carried out by several researchers. In [11], they built a sentiment analysis model to analyze Chinese texts using suffix trees and reciprocal information. In contrast to them, [12] uses lexical resources, such as SentiWordNet, WordNet, and the NLTK toolkit to classify terrorism data that appears on web forums. [13] also used a lexicon-based sentiment analysis, but to reduce data sparsity and improve the accuracy of sentiment detection, they proposed rule-based classification. In [14], they proposed cognitive psychology to detect misinformation on online social networks. To measure differences in retweeting behavior they use the Gini coefficient. With this method, the algorithm does not use any specific features of Twitter.

Instead of classifying radical content, [15] investigated the sentiments of radical writers. Using Parts-Of-Speech designation, sentiment analysis, and the algorithm they developed, they tried to classify radical authors. Their results conclude that although there is no simple typology, their method is flexible enough to identify some of the properties of radical online users. In [30], they used student's native language feedback to do the sentiment analysis problem.

Due to the limitations of the lexicon-based method, some researchers turn to the machine learning method. In [16], they showed that in terms of accuracy the machine learning method is better than the lexicon method. Some other researchers who also use machine-based learning methods are [17] who uses a KNN-based machine learning classification system to identify conversations related to extremists. In [18], they used the Support Vector Machine (SVM) method to detect hatred on social networks. The use of SVM was also carried out by [19].

Although machine learning methods are better than lexicon methods, they are very dependent on the choice of data representation (features). The engineering of these features is very dependent on human abilities and prior knowledge about the problem at hand [20]. The SVM model requires the right kernel for each different data type. Linear classifications such as linear regression and moving averages are very dependent on gradient descent to classify data. Tree models face computational complexity when developing nodes and tree depths. Neural network models are becoming obsolete because of the proposed deep learning with more complex computational capabilities to extract more data.

With the advent of deep learning based on artificial neural networks, more and more researchers have begun to focus on this method. In [21], they used a deep learning method to detect harassment content on Social Media Sites. Similarly, [22] combined the Convolutional Neural Network (CNN) model with the Long Short Term Memory (LSTM) network. The CNN model is to identify content that contains extremist elements, while the LSTM model is to capture remote dependencies throughout the review. In this study we propose a new method of using deep learning to classify radicalism in Indonesian. We use word2vec as word embedding and the combination of Restricted Boltzmann Machine (RBM) and back-propagation network as our basic classification method.

3. **Problem Definition and Contributions.** The dataset consists of a set of user comments, $UC = uc_1, uc_2, \ldots, uc_n$, where n is the number of user comments. A label for each user comment or uc_i is expressed by labels $L = L[UC]_{2\times n}$. This matrix L denotes the label of radicalism of user comment uc_i where i is an index of the user comments. The element of L is a binary value, where 0 represents radical and 1 represents non-radical comment.

The typical baseline classifier for classification algorithms is Support Vector Machine (SVM), linear classifier (linear regression and moving average), tree (decision tree and random forest), and neural networks (multi layer perceptron, convolution neural network, and fully connected neural network). The advantage of these algorithms is the ease of use of implementation. The drawback of many classification algorithms is parameter tune-up and lack of stable configuration setting. The generalization and over-fitting are derived from the configuration setting. The classification algorithms tend to fit only to training data. The SVM model needs to use the correct kernel for different types of data. The linear classifier such as linear regression and moving average is very dependent on the gradient descent to classify the data. The tree models face the complexity of computation when developing nodes and tree depths. The neural network models become old fashioned since deep learning is proposed with higher complex computational capabilities to extract more data.

We propose a novel method using deep learning to classify radicalism and hate speech from user comments for the Indonesian language. We use word2vec as word embedding and the combination of Restricted Boltzmann Machine (RBM) and back-propagation network as our deep base classifier method. This paper makes several contributions as follows. 1) We propose sentiment analysis to predict radicalism and hate speech for the Indonesian language and implement 5 step pre-processing to reduce the noise, clean the data, and handle unusual comment style text. 2) We propose a combination of RBM and Back-Propagation network (RBMBP) to enhance the deep learning classifier capability and compare the results with baseline classifier methods such as SVM, MLP, and decision tree algorithms.

In this paper, we represent the radicalism value using a binary value as the output in the RBMBP model. The binary value represents the label data of our dataset. Label 1 means the user comment belongs to radicalism comment, while 0 is not. In this paper, we only consider the two classes of the user comment classification, and the main concern is to detect the radicalism or hate speech contained in the user comment. The binary classification method is also very intuitive and easy to explain. The computation complexity of the binary classification is lower compared to the multi-class classification method. Table 1 shows the relationship between user comments and label. The user comments will be read and go through pre-processing process such as cleaning, stemming, converting and removing unimportant words. Each user comment will be processed by word2vec to form

Table 1. User comments – Label

UC (in Indonesian)	**Label**
Soal jln Jatibaru,polisi tdk bs GERTAK gubernur .Emangny polisi tdk ikut pmbhasan? Jgn berpolitik. Pengaturan wilayah,hak gubernur. Persoalan Tn Abang soal turun temurun.Pelik.Perlu kesabaran. [USERNAME] [USERNAME] [URL] **Translate**: "Regarding the road Jatibaru, the police could not THREAT the governor. Did the police not take part in the discussion? Do not play politics. Regional regulation, governor's rights. Tn. Abang's problem is a matter of generation. Complicated. Need patience. [USERNAME] [USERNAME] [URL]"	1
Kepingin gudeg mbarek Bu hj. Amad Foto dari google, sengaja, biar teman-teman jg membayangkannya. Berbagi itu indah. **Translate**: "Want to eat gudeg mbarek Mrs. hj. Amad, Photo from google, intentionally, so my friends can imagine it. Sharing is beautiful."	0
...	...
Sesama cewe lho (kayaknya), harusnya bisa lebih rasain lah yang harus sibuk jaga diri, rasain sakitnya haid, dan paniknya pulang malem sendirian. Gimana orang asing? Wajarlah banyak korban yang takut curhat, bukan dibela malah dihujat. **Translate**: "Fellow girls (I think), you should be able to feel more who has to be busy taking care of yourself, feeling sick during menstruation, and panicking to go home late at night alone. What about strangers? Naturally many victims were afraid to confide, were not defended but were blasphemed."	1

a feature vector V as a matrix of weights value $V = [v_{uc,l}]_{m \times n}$, where m is the dimension of the vector and n is the total number of user comment uc with label l. Matrix V represents the word vector representing user comment toward label of radicalism and hate speech.

The deep learning model is combination between RBM net and Back-Propagation model (RBMBP). The deep learning model starts with RBM net. The input of the RBM net is the word vector V which is generated from word2vec model. The RBM net is well known for its capability to do feature extraction from the data [23]. We want to utilize this capability to extract the feature from word vector V and generate the feature word vector Ve as a matrix of $Ve = [ve_{uc,l}]_{m \times n}$. The back-propagation network will use the feature vector Ve to classify current user comment into radicalism and hate speech or not.

4. **Proposed Methodology.** In this section, we present our approach to mining the public opinion for radicalism in social media via sentiment analysis. Figure 1 shows our system architecture. The system starts by extracting user comments from social media platforms. The system applies text filtering to filtering the relevant text from the topic. The filtering process is used to reduce the sparsity of the dataset. The remaining words will be stored in the document repository. The system implements the pre-processing step to remove stop-word, carry out tagging, word stemming, converting, lower-casing, and noise-cleaning. In this paper, we remove all the sarcasm manually and label it as

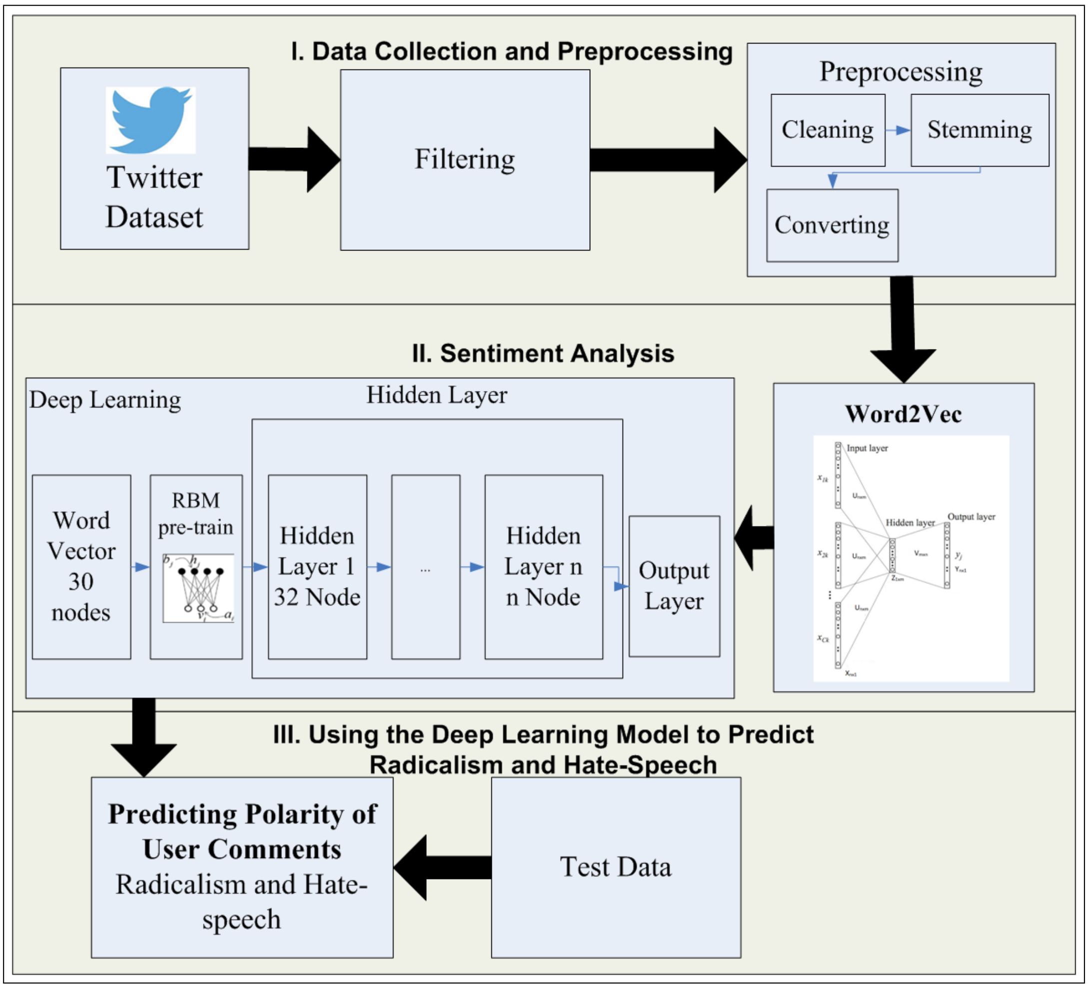


FIGURE 1. The system architecture

noise. We remove all the sarcasm because to deal with sarcasm is already a challenging problem.

4.1. **Pre-processing.** The pre-processing step includes the following steps, which present the corpus data in a more structured form to easily extract features and opinion words.

4.1.1. *Removing stop-words and cleaning.* The first step of the pre-processing step is to remove the stop-words and do the cleaning of the texts. The most common words in a text, such as *atau, dan, adalah*, and *dengan* make the accuracy of the sentiment analysis decrease significantly. Noise also makes the model hard to fit in the training phase [23]. We implement noise-cleaning by filtrating comments which are less than ten words and do not contain opinion words. The system also detects the opposite impression, for example, "not bad" phrase has "good" meaning.

Some user comments have symbols (@, ?, !, *, etc.) and emojis (:D, :'(, etc.) in it. The system removes all of these that do not have a contribution to the sentiment. Sentiment analysis in the transportation area considers numbers in their analysis, as they need the user sentiment in some route or transportation means [24]. In this paper, the system removes all the numbers contained in the user comments.

4.1.2. *Word stemming and lemmatization.* Stemming and lemmatization are a process of converting a word to its basic form. Stemming is a process to prepare text, words, and documents for further processing. In natural language processing, many texts are derived from another text as their use in the speech. This change is called inflected language. Indonesian language is different from the English language. Indonesian language does not have grammatical categories such as tense. The verb in the Indonesian language does not have irregular and regular word tense. Word stemming helps the system to achieve the root forms of inflected words.

Lemmatization is an important process in the sentiment analysis [25]. It employs a lexicon to determine the lemma inside the sentence. The lexical information of each word can be obtained after this process is finished. Because of the importance of the stemming and lemmatization, our system implements it. We implement katadasaR [26] package in R to stem the Indonesian words.

4.1.3. *Converting characters and lower-casing.* The last step of pre-processing is to convert unusual words on social platforms to express their opinion. Many social media users' typing comments are influenced by gender, friendship style, and culture in different regions. The words are usually not a standard use, and contain many local dialect also. In this step, the system will replace the unusual words with the standard use, and delete the unknown words based on our dictionary. The more variance of the words with the same meaning also makes the dataset sparser. Machine learning model needs more time and complex computational resources to process the sparse dataset.

The system also deals with a different style of typing case. To avoid confusion with case sensitive problems in some programming language, the system will lower casing all the words in the pre-processing step, and make all the words in lower-case type. All the pre-processing step is to make the system fitter to classify the main problem, which is predicted radicalism and hate speech comment in the Indonesian language.

4.2. **Word2vec model.** The next step of the system is the topic modeling and word embedding. In this process, the feature vector is generated. We used topic modeling and word embedding algorithms such as word2vec [27], Continues Bag of Words (CBoW) [28], and skip-gram [29]. In this paper, we implement word2vec to do the word embedding. In this step, the system will train a model to reconstruct linguistic contexts of words.

Word2vec takes a large corpus of words and produces a vector space. Word2vec can utilize either of two model architectures between the CBoW and skip-gram model. In the CBoW architecture, the model predicts the current word from the bag-of-words surrounding the context words. The order of the words does not influence the prediction result. Opposite from CBoW, in the skip-gram architecture, the model uses the current word to predict the surrounding window of context words. The skip-gram architecture weighs the closest context words heavier than the far context words. The CBoW is a faster algorithm while skip-gram is slower but does a better job for infrequency words.

Word2vec is a neural network architecture that studies word embedding from a large corpus of data [24]. The model works to create a feature vector in a high-dimensional space. In this paper, we trained and used CBoW because the architecture is faster to handle many words. Figure 2 shows the CBoW model to predict the current word context based on surrounding context words. In the CBoW model, input to the model is the sum of one-hot encoded vectors of the context words within the window size. The size of the input (X) will be $n \times 1$. The set of X contains $x_{1k}, x_{2k}, \ldots, x_{Ck}$. The model will train two types of weight matrices U and V. The hidden layer Z consists of $1 \times m$ dimension. The output layer Y_j consists of $n \times 1$ dimension. The CBoW model applies the logarithmic loss function as is shown in Equation (1).

$$loss = -\frac{1}{N}\sum_{i=1}^{N} y_i \cdot \log(p(y_i)) + (1 - y_i) \cdot \log(1 - p(y_i)) \tag{1}$$

where N is total number of nodes, y_i is the predicted word in the output node, and i is the index of nodes. The CBoW model applies softmax activation function only in the last

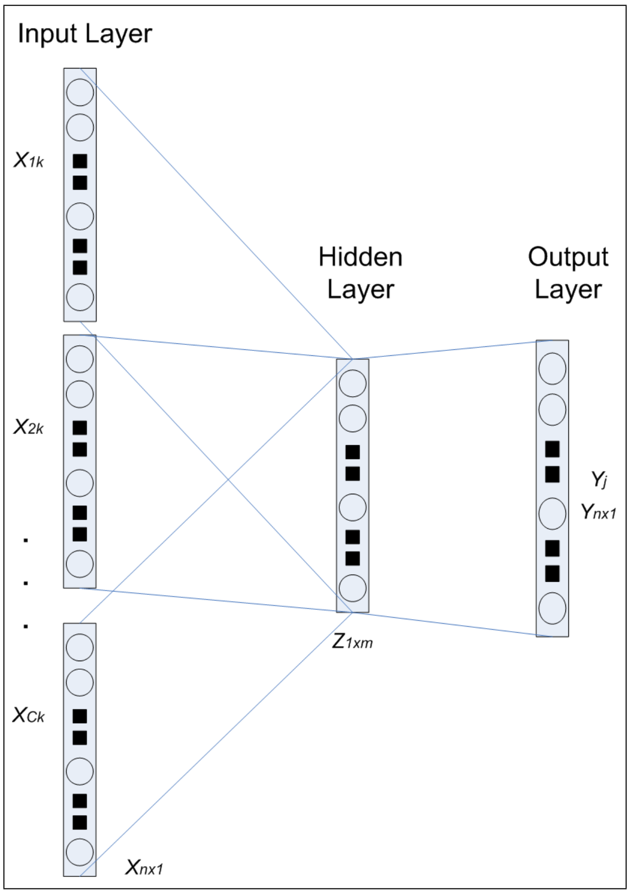


FIGURE 2. Architecture of the word2vec

layer since it gives the output vector as probabilities of occurrence for each word in the corpus. The softmax activation function is shown by Equation (2).

$$\sigma(x_j) = \frac{e^{x_j}}{\sum e^{x_i}} \tag{2}$$

where σ is a softmax function of x_j, i is index for input nodes, and j is index for output nodes.

In this paper, we applied the word2vec feed forward propagation training model to calculating the weights. The input vector X will be transposed to make the $1 \times n$ dimension. In the beginning, the value of vector U will be a randomly initialized matrix of size $n \times m$. Vector Z is a dot matrix between X and U with the size of $1 \times m$. To achieve the output value, the system initialized vector V with a randomly initialized matrix of size $m \times n$. The dot product between new vector Z and matrix V will generate a new vector A. Softmax function will be applied on transpose of vector A giving new vector Ve of dimension $n \times 1$ as the final output.

4.3. **Deep learning model.** In this section we will describe the deep learning model we use to classify the word vector Ve into output class. The vector Ve is represented by matrix $Ve = [ve_{uc,l}]_{m \times n}$, where n represents the dimension of the vector, and m represents the total user comments. Matrix Ve is the word vector from the word2vec model, and becomes the input of the RBM model in the first part of the deep RBMBP model.

RBM is a fully connected neural network that is powerful for extracting features from data. The system utilized the strength of the RBM to extract important features from vector Ve. The RBM input layer consists of 30 nodes, which come from the 30 most popular word2vec's radicalism and hate-speech words. The RBM will train the input vector into new input vector for the back-propagation input. The new input vector is called map feature vector, where the values are generated from the extraction of the RBM. The standard RBM model consists of a visible layer v_i, with a hidden layer h_j, and a matrix of weights $W = [W_{i,j}]$. The bias in the RBM exist in both layers, a_i is the bias for the visible layer and b_j is the bias for the hidden layer. The energy function for a configuration (v, h) is defined by Equation (3).

$$E(v, h) = -\sum_{ij} v_i W_{ij} h_j - \sum_i a_i v_i - \sum_j b_j h_j \tag{3}$$

The energy function is a type of probability distribution over the input vector, defined as $P(v)$, as shown in Equation (4).

$$P(v) = \frac{1}{Z} \sum_h e^{-E(v,h)} \tag{4}$$

where Z is partition of all probabilities configurations. We use logarithmic softmax function to calculate the output value from the RBM model. The RBM used two softmax functions for each layer. The first calculation is between input layer and hidden layer. The probabilities of nodes value in the hidden layer from the input layer are calculated using a softmax function shown by Equation (5).

$$p(h_j = 1|v) = \sigma\left(\sum_i w_{ij} v_i + b_j\right) \tag{5}$$

where $p(h_j = 1|v)$ is the probabilities of nodes value in hidden layer from the visible layer v. $\sigma(\cdot)$ is sigmoid function, w_{ij} is the weights matrix, v_i is the visible layer, and b_j is the bias generated from the hidden layer. The system needs to reconstruct the hidden layer into output layer with the following Equation (6).

$$p\left(v_i'|h\right) = \sigma\left(\sum_j w_{ij} + a_i\right) \tag{6}$$

where $p\left(v_i'|h\right)$ is the probabilities of nodes value in reconstruct visible layer from hidden layer. $\sigma(\cdot)$ represents the sigmoid function, w_{ij} is the weights matrix, v_i' is the reconstruct visible layer, and a_i represents bias which is generated from reconstruction of the new visible layer.

The reconstruct visible layer generated by RBM model is called the feature map vector $Ve' = \left[ve'_{uc,l}\right]_{m\times n}$. The feature map vector is an input for the Back-Propagation (BP) model. The BP model will now classify the feature map vector into binary class, radicalism or not. Equation (7) shows the output nodes calculation using gradient descent method.

$$o_j = f(net_j) = \sigma\left(\sum_{k-1}^{n} w_{kj} o_k\right) \tag{7}$$

$$f(x) = \begin{cases} x, & \text{if } x > 0 \\ 0.01x, & \text{otherwise} \end{cases} \tag{8}$$

$$f(net_j) = \frac{1}{1+e^{-net_j}}, \quad net_j = \sum_{k-1}^{n} w_{kj} x_k - \theta_j \tag{9}$$

where σ is activation function, k is the number of neurons in the previous layer, j is the neuron in the current layer, and n is the number of inputs units to the neuron. w_{kj} represents the weight between k and j. In this paper, we implement the Rectified Linear Unit (ReLU) activation function between the input layer and the hidden layer, among hidden layers. For the hidden layer to the output layer, we implement the softmax activation function to predict the binary class output. The ReLU activation function is shown by Equation (8), while the softmax activation function is shown by Equation (9). The computation is repeated until the satisfying target output is reached.

5. **Experiments and Results.** In this section, we present our results and discuss the validation procedure of the proposed method.

5.1. **Data collection and filtration.** Dataset is collected from twitter APIs. The APIs allowed the system to collect the tweets from the users automatically. The twitter APIs allow user to make queries for what type of data they want to extract. We crawled the tweets related to emotions. Radicalism and hate-speech related to emotions such as anger, fear, and anxiety. The dataset consists of 4,400 user comments. However, REST APIs permit a single customer to employ only 350 queries per 15 min, and it retrieves 3,200 of the most-recent tweets per query [24].

Social media has vast amounts of data. The volume of social media data will be grown double by 2025. The capabilities of the classic knowledge-based data filtering system cannot work with large amounts of social media data. In this paper, the filtering process is conducted by crawling the specific queries which contain negative emotions. We assumed radicalism and hate-speech belonging to negative emotions will minimize the scope of the user comments which are needed to retrieve.

5.2. **Performance evaluation.** In this section, we describe the configuration and parameter setting for the baseline evaluation methods using different classifiers. We used Waikato Environment for Knowledge Analysis (Weka) to implement the baseline methods. We compare the classification algorithms with our proposed model. The configuration of the baseline model is shown as follows.

- Support Vector Machine (SVM): The kernel for SVM we used is Libsvm. This kernel classifies data with a linear kernel. The training process permits sparse data so the training dataset comprises non-zero values. The parameter setting of the training kernel is as follows. The batch size is 100, cacheSize is 40, cost is 1, degree is 3, and kernel type is radial basis function.
- Multilayer Perceptron (MLP): The MLP we utilized is Weka library MLP and uses a sigmoid function and a squared error and mean absolute error function for the evaluation of the classification approaches. The parameter setting of the training is as follows. The batch size is 100, consists of one hidden layer, the learning rate is 0.3, momentum is 0.2, and training epochs is 500.
- Random Forest: The Random Forest algorithm that is multi-modal decision tree with the best tree will be picked as the output result. The parameter setting of the trees is as follows. The batch size is 100, maxdepth is 0, seed is 1, and number of iterations is 100.
- RBMBP: The proposed method hyper-parameter configuration setting is as follows. The batch size is 100, the epochs is 100, the number of hidden layers is 5 with the number of nodes being 30, 30, 20, 10, 5, respectively. The optimizer is adam with learning rate 0.0001, and weight decay is 0.000001. The activation function of each hidden layer is ReLU, and the activation function of the output layer is sigmoid function. The dropout is only applied in the first hidden layer with 0.5 rate value.

For each classifier, we compare the precision, recall, F-score, Root Mean Square Error (RMSE), Mean Absolute Error (MAE) using the testing dataset. All these baseline methods will be compared with RBMBP.

5.3. **Results.** In this section, we present results based on the experiments described in performance evaluation. Figure 3 shows the bag of words from the word2vec model. The words like "leceh:[Eng: insult]", "harassment", "ancam:[Eng: threat]", "mingkem:[Eng: shut up]", "perang:[Eng: war]", "bom:[Eng: bomb]", are examples of the radical words in Indonesian. The system records all words in the form of a matrix-vector. The system has a bag of words for each word, and we select only the top 30 words in the matrix. The top 30 words will be the input vector to the RBM model.

The system consists of two main parts. The first part is to learn features from the matrix-vector of word2vec model. The second part is to classify the feature map vector into the system's target class. Using the RBM the system will construct the feature map vector. Figure 4 shows a preview of the feature map vector which is generated from the RBM model. The value for each node in the feature map vector is normalized in range 0 to 1. Each node represents the weight of the word in the user comment. This feature map vector will become the input for the second part of the system.

Figure 5 shows the training results of the RBMBP model. The model becomes convergence after 50 iterations. The model uses Mean Square Error (MSE) loss function, and achieves minimum MSE loss value at 0.0607. The model applied optimizer adam, and l1 and l2 regularization. Using regularization makes the model will keep the training to avoid overfitting. This is possible because the model implements weight penalization whenever the loss value goes below validation loss value. The system also uses accuracy

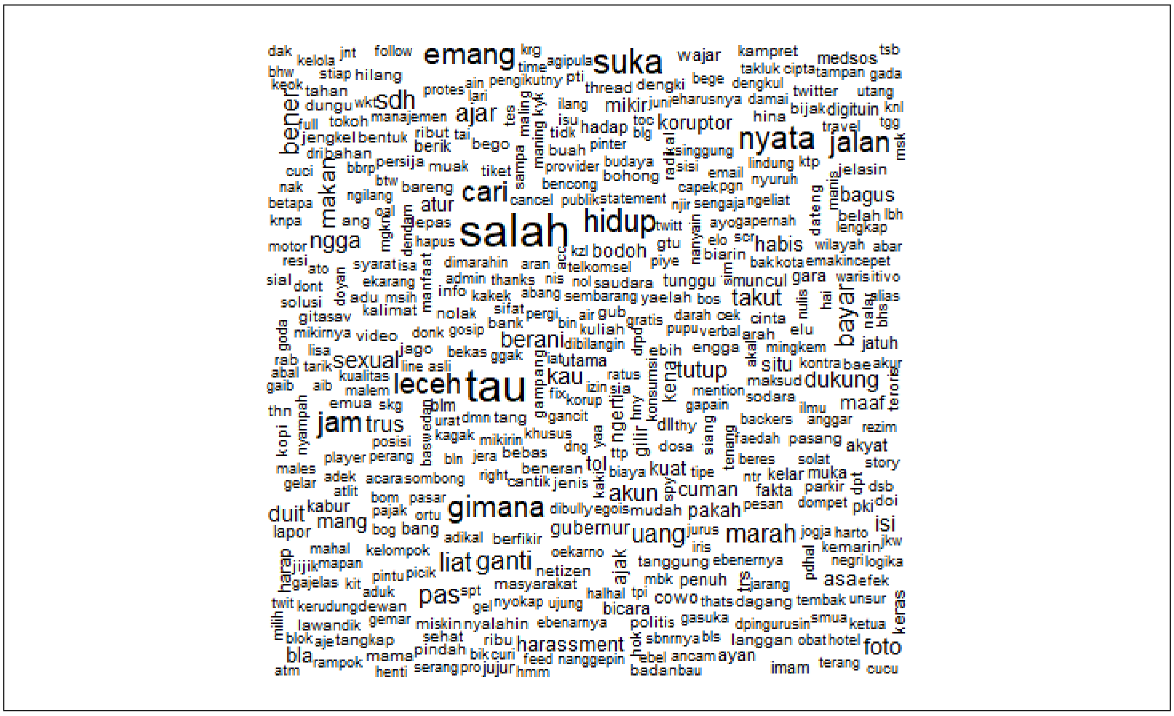
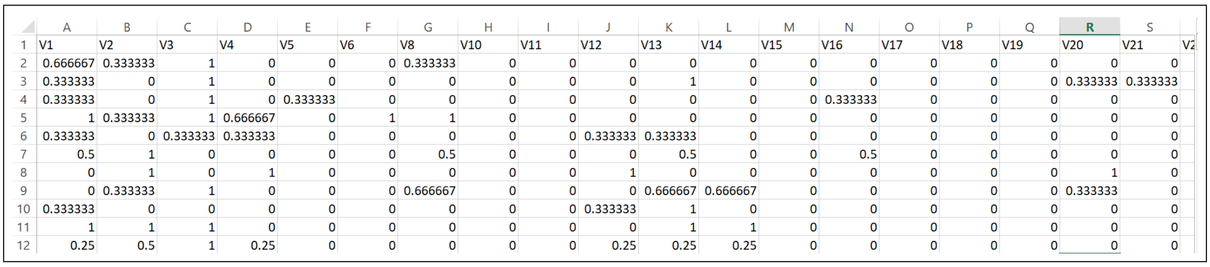


FIGURE 3. The bag of words from word2vec



	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	
1	V1	V2	V3	V4	V5	V6	V8	V10	V11	V12	V13	V14	V15	V16	V17	V18	V19	V20	V21	V2
2	0.666667	0.333333	1	0	0	0	0.333333	0	0	0	0	0	0	0	0	0	0	0	0	
3	0.333333	0	1	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0.333333	0.333333	
4	0.333333	0	1	0	0.333333	0	0	0	0	0	0	0	0	0.333333	0	0	0	0	0	
5	1	0.333333	1	0.666667	0	1	1	0	0	0	0	0	0	0	0	0	0	0	0	
6	0.333333	0	0.333333	0.333333	0	0	0	0	0	0.333333	0.333333	0	0	0	0	0	0	0	0	
7	0.5	1	0	0	0	0	0.5	0	0	0	0.5	0	0	0.5	0	0	0	0	0	
8	0	1	0	1	0	0	0	0	0	1	0	0	0	0	0	0	0	1	0	
9	0	0.333333	1	0	0	0	0.666667	0	0	0	0.666667	0.666667	0	0	0	0	0	0.333333	0	
10	0.333333	0	0	0	0	0	0	0	0	0.333333	1	0	0	0	0	0	0	0	0	
11	1	1	1	0	0	0	0	0	0	0	1	1	0	0	0	0	0	0	0	
12	0.25	0.5	1	0.25	0	0	0	0	0	0.25	0.25	0.25	0	0	0	0	0	0	0	

FIGURE 4. Preview of the feature map vector

metrics to evaluate the prediction. The highest accuracy that the system can achieve is 81.63%.

We also make a comparison of RBMBP among baseline classifier methods. Table 2 shows the comparison of the prediction results. The first column is the name of the classifiers. The next three columns are measurement metrics: Root Mean Square Error (RMSE), MAE, and Accuracy. We also measure precision and recall to measure F-Measure which are represented in columns five, six, and seven respectively. Table 2 shows RBMBP outperforms base-line classifiers. In measurement metric RBMBP gets the best loss value in RMSE and MAE at 0.236 and 0.0735 respectively, and LibSVM gets the second best loss value in RMSE and MAE at 0.3009 and 0.0955. MLP gets the RMSE at 0.3041, and MAE at 0.131. Random forest gets RMSE at 0.3642, and MAE at 0.133. For accuracy value RBMBP gets the best value at 81.63%, while libSVM gets the second best at 80.9%. MLP gets accuracy value at 77.4%, and random forest at 73.5%.

We also measure the F-Measure for measuring classification performance. F-Measure shows the balance between precision and recall. Precision represents how precise or accurate the model to predict the positive results by total predicted positive. While recall represents how many of the actual positives labeling it as positive. In the experiments,

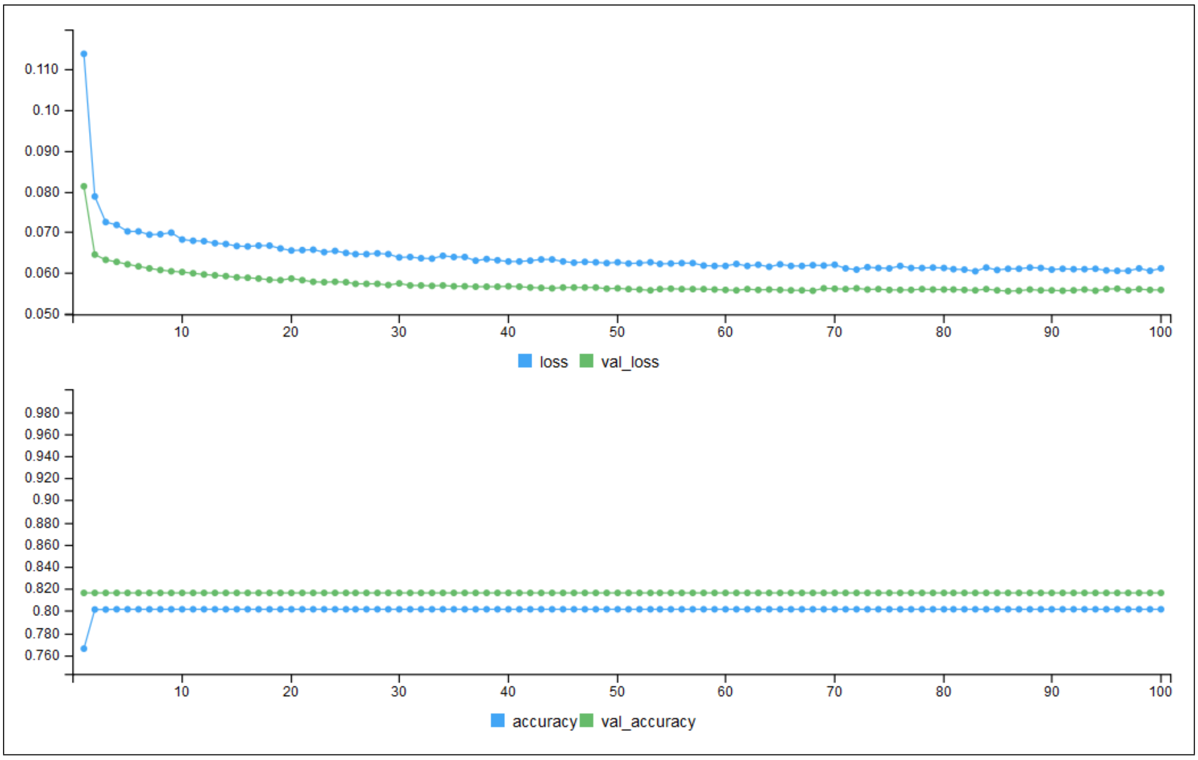


FIGURE 5. The training loss and accuracy value

TABLE 2. Comparison of results between methods (test phase)

Methods	RMSE	MAE	Accuracy	Precision	Recall	F-Measure
MLP	0.3041	0.131	0.774	0.746	0.774	0.757
libSVM	0.3009	0.0955	0.809	0.655	0.809	0.724
Random forest	0.3642	0.133	0.735	0.753	0.735	0.744
RBMBP	**0.236**	**0.0735**	**0.8163**	**0.8015**	**0.8051**	**0.8033**

RBMBP gets the best value for precision, recall, and F-Measure at 0.8015, 0.8051, and 0.8033. Random forest gets the second-best in precision at 0.753, MLP gets the precision at 0.746, and libSVM at 0.655. LibSVM gets the second-best recall value at 0.809, MLP gets the third at 0.774, and the last is the random forest with 0.735. In the F-Measure metric, the second-best is MLP at 0.757, the third is the random forest, and the last is libSVM at 0.724.

The system detects 2,747 radicalism or hate speech comments. From the detection results, the 40.08% express anger, 23.62% express fear, and 36.29% expressing sadness towards conditions. From comments which express anger, the "jihad", "bom", "rusuh", and "demo" words appear in the most part. In the comments which express fear, the "sakit", "sulit", and "haram" appear in the most part. In the comments which express sadness, the "meninggal", "hancur", and "bodoh" appear in the comments.

6. Conclusions.

We proposed a sentiment analysis of public opinion on radicalism. We utilized the deep learning methods via a combination of the RBM and back-propagation model. From the experiments, RBMBP outperforms other classifiers showing by the confusion metrics table. Using the RBM model, the word polarity of the word vector is enhanced by the feature map word vector. The sparsity in each node in the word vector is reduced, causing the training process to get better results.

In future work, we will explore the related between cities, cultures, and religions in the radicalism and hate-speech of sentiment analysis. We also like to explore topics and trust among users, as users are relatively more open in their communities rather than in public spaces.

Acknowledgment. This work is supported by internal research funds Satya Wacana Christian University. The authors also gratefully acknowledge the helpful comments and suggestions of the reviewers, which have improved the presentation.

REFERENCES

[1] A. Guiora and E. A. Park, Hate speech on social media, *Philosophia*, vol.45, pp.957-971, 2017.
[2] C. Carlson, Censoring hate speech in U.S. social media content: Understanding the user's perspective, *Communication Law Review*, vol.7, no.1, pp.24-45, 2017.
[3] M. Mondal, L. A. Silva and F. Benevenuto, A measurement study of hate speech in social media, *Proc. of the 28th ACM Conference on Hypertext and Social Media*, New York, USA, pp.85-94, 2017.
[4] Z. Zhao, J. Zhao, Y. Sano, O. Levy, H. Takayasu, M. Takayasu, D. Li, J. Wu and S. Havlin, Fake news propagates differently from real news even at early stages of spreading, *EPJ Data Science*, vol.9, no.7, 2020.
[5] M. Mendoza, B. Poblete and C.Castillo, Twitter under crisis: Can we trust what we RT?, *Proc. of the 1st Workshop on Social Media Analytics*, New York, pp.71-79, 2010.
[6] J. Mei and R. Frank, Sentiment crawling: Extremist content collection through a sentiment analysis guided web-crawler, *Proc. of the 2015 IEEE/ACM International Conference on Advances in Social Networks Analysis and Mining*, Calgary, Alberta, pp.1024-1027, 2015.
[7] Q. Schiermeier, Terrorism: Terror prediction hits limits, *Nature*, vol.517, no.7535, p.419, 2015.
[8] J. Menchik, Productive intolerance: Godly nationalism in Indonesia, *Comparative Studies in Society and History*, vol.56, no.3, pp.591-621, 2014.
[9] J. Franco and N. Warburton, Should there be limits on hate speech?, *SAGE Journals*, vol.42, no.2, pp.150-152, 2013.
[10] H. Thakkar and D. Patel, Approaches for sentiment analysis on Twitter: A state-of-art study, *arXiv: 1512.01043*, 2015.
[11] D. Zeng, D. Wei, M. Chau and F. Wang, Domain-specific Chinese word segmentation using suffix tree and mutual information, *Information Systems Frontiers*, vol.13, pp.115-125, 2011.
[12] T. Chalothorn and J. Ellman, Using SentiWordNet and sentiment analysis for detecting radical content on web forums, *The 6th Conference on Software, Knowledge, Information Management and Applications (SKIMA 2012)*, Chengdu, China, 2012.
[13] M. Z. Asghar, A. Khan, S. Ahmad, M. Qasim and I. A. Khan, Lexicon-enhanced sentiment analysis framework using rule-based classification scheme, *PLoS ONE*, vol.12, no.2, 2017.
[14] K. P. Kumar and G. Geethakumari, Detecting misinformation in online social networks using cognitive psychology, *Human-Centric Computing and Information Sciences*, vol.4, no.14, 2014.
[15] R. Scrivens, G. Davies and R. Frank, Searching for signs of extremism on the web: An introduction to sentiment-based identification of radical authors, *Behavioral Sciences of Terrorism and Political Aggression*, vol.10, no.1, pp.39-59, 2018.
[16] S. A. Azizan and I. A. Aziz, Terrorism detection based on sentiment analysis using machine learning, *Journal of Engineering and Applied Sciences*, vol.12, no.3, pp.691-698, 2017.
[17] Y. Wei, L. Singh and S. Martin, Identification of extremism on Twitter, *Proc. of the 2016 IEEE/ACM International Conference on Advances in Social Networks Analysis and Mining (ASONAM'16)*, pp.1251-1255, 2016.
[18] S. Malmasi and M. Zampieri, Detecting hate speech in social media, *arXiv: 1712.06427v2*, 2017.
[19] K. Bedjou, F. Azouaou and A. Aloui, Detection of terrorist threats on Twitter using SVM, *Proc. of the 3rd International Conference on Future Networks and Distributed Systems (ICFNDS'19)*, Association for Computing Machinery, New York, NY, USA, 2019.
[20] Y. Bengio, A. Courville and P. Vincent, Representation learning: A review and new perspectives, *IEEE Trans. Pattern Analysis and Machine Intelligence*, vol.35, no.8, pp.1798-1828, 2013.
[21] H. Chen, S. McKeever and S. J. Delany, A comparison of classical versus deep learning techniques for abusive content detection on social media sites, in *Social Informatics*, S. Staab, O. Koltsova and D. Ignatov (eds.), 2018.

[22] S. Ahmad, M. Z. Asghar, F. M. Alotaibi and I. Awan, Detection and classification of social media-based extremist affiliations using sentiment analysis techniques, *Human-Centric Computing and Information Sciences*, vol.9, no.24, 2019.

[23] R.-C. Chen and Hendry, User rating classification via deep belief network learning and sentiment analysis, *IEEE Trans. Computational Social Systems*, vol.6, no.3, pp.535-546, 2019.

[24] F. Ali, D. Kwak, P. Khan, S. El-Sappagh, A. Ali, S. Ullah, H. K. Kye and K. S. Kwak, Transportation sentiment analysis using word embedding and ontology-based topic modeling, *Knowledge-Based Systems*, vol.174, pp.27-42, 2019.

[25] F. Ali, E. K. Kim and Y. G. Kim, Type-2 fuzzy ontology-based opinion mining and information extraction: A proposal to automate the hotel reservation system, *Applied Intelligence*, vol.42, pp.481-500, 2015.

[26] Nurandi, *An R Package of Word Stemming for Bahasa Indonesia Using Nazief & Adriani's Algorithm*, https://github.com/nurandi/katadasaR, Accessed on May 02, 2020.

[27] T. Mikolov, I. Sutskever, K. Chen, G. Corrado and J. Dean, Distributed representations of words and phrases and their compositionality, *Proc. of the 26th International Conference on Neural Information Processing Systems – Volume 2*, Red Hook, NY, USA, pp.3111-3119, 2013.

[28] B. Liu, Text sentiment analysis based on CBOW model and deep learning in big data environment, *Journal of Ambient Intelligence and Humanized Computing*, vol.11, pp.451-458, 2018.

[29] D. Guthrie, B. Allison, W. Liu, L. Guthrie and Y. Wilks, A closer look at skip-gram modelling, *Proc. of the 5th International Conference on Language Resources and Evaluation*, Genoa, Italy, 2006.

[30] M. Aldagdoog, S. Kother and S. Alzaghair, Sentiment analysis to monitor student feedback in their native language, *ICIC Express Letters*, vol.13, no.10, pp.899-906, 2019.