The Korean Institute of Communications and Information Sciences

Available online at www.sciencedirect.com

ScienceDirect

ICT Express 8 (2022) 396–408

www.elsevier.com/locate/icte

Detection of fake news using deep learning CNN–RNN based methods

I. Kadek Sastrawan, I.P.A. Bayupati*, Dewa Made Sri Arsa

Department of Information Technology, Universitas Udayana, Badung, Indonesia

Received 3 July 2021; received in revised form 15 September 2021; accepted 6 October 2021
Available online 22 October 2021

Abstract

Fake news is inaccurate information that is intentionally disseminated for a specific purpose. If allowed to spread, fake news can harm the political and social spheres, so several studies are conducted to detect fake news. This study uses a deep learning method with several architectures such as CNN, Bidirectional LSTM, and ResNet, combined with pre-trained word embedding, trained using four different datasets. Each data goes through a data augmentation process using the back-translation method to reduce data imbalances between classes. The results showed that the Bidirectional LSTM architecture outperformed CNN and ResNet on all tested datasets.

Keywords: Fake news detection; Deep learning; CNN; Bidirectional LSTM; ResNet

1. Introduction

The development of the internet has a positive impact on society because it provides broad access to information. These developments also impact the company to digitize its products to adapt to technological developments to retain its customers [1]. This advancement brings a massive invention of online social networks (OSN), especially multimedia social networks (MSN), a type of OSN focusing on multimedia sharing experience [2]. Zhang et al. proposed a framework to address the rights management of contents in MSN, security, and ease of use. While online and multimedia social networks provide advantages in communication and technology, these innovations severely impact the social aspect. Zhang et al. [3] proposed a novel model of spatio-temporal access control for protecting the privacy and information security of users in OSN. Srinivasan & Dhinesh Babu [4] proposed a parallel neural network to identify rumor because it harms society [5,6]. This rumor must be validated and tends to lead to fake news. Even information that is considered accurate sometimes still presents fake news, whether intentional or not. Sahoo & Gupta [7] proposed a Chrome extension to detect malicious profiles in Twitter using various features on the profile itself and machine learning. Sahoo & Gupta [8] also proposed a Chrome extension to automatically detect fake news on Facebook using multiple features. Sahoo & Gupta [8] stated that there are four main features of fake news such as: news content, social content, target victim, and creater and spreader.

Recently, the spread of fake news has become more prevalent due to the ease of creating and distribute information on the internet. Fake news itself is not actual but is made real for a specific purpose [9]. A statistic on the American public's ability to distinguish between real and fake news explains that only 26% of the total respondents feel very capable of identifying between real and fake news [10]. This value is still low, and the community's insufficient ability to differentiate between real and fake news also plays a role in spreading fake news. One example of the spread of fake news known on the internet is fake news during the US presidential election in 2016 [11]. The spread of fake news has catastrophic implications and potential dangers in the political and social fields [12]. Therefore, research on detecting fake news is being intensified due to the effect of fake news spreads.

This study extensively analyzes several deep learning methods performances combined with the current state of the art word embeddings on benchmark fake news datasets. We utilized pre-trained word embedding such as Word2Vec, Glove, and fastText. These pre-trained word embeddings are popular

* Corresponding author.
E-mail addresses: sastrawanikadek@gmail.com (I.K. Sastrawan), bayupati@unud.ac.id (I.P.A. Bayupati), dewamsa@unud.ac.id (D.M.S. Arsa).

Peer review under responsibility of The Korean Institute of Communications and Information Sciences (KICS).

https://doi.org/10.1016/j.icte.2021.10.003

pre-trained word embedding and were chosen because they have been trained using a massive corpus to produce sufficient vocabulary. Each pre-trained word embeddings is combined with the respective deep learning methods, namely CNN, Bidirectional LSTM, and ResNet, to determine their performance in detecting fake news. CNN and Bidirectional LSTM were chosen because both methods are known to give good results in text processing. At the same time, ResNet was developed to increase CNN's resistance to vanishing gradients by adding a residual block. Furthermore, we also empirically show that the performance of deep learning in detecting fake news can be boosted by selecting proper word embeddings.

This paper is divided into several main sections: Section 2 describes the literature review and related works. Section 3 explains the stages of the research and the proposed method. Section 4 describes the data and devices used and experiment scenarios. Section 5 shows the experimental results obtained as well as discussions. Finally, Section 6 presents conclusions and plans for future work.

2. Related works

Various methods have been explored to detect fake news, such as research in [13], which is conducted using the N-gram and TF–IDF methods for feature extraction as well as various classifiers such as Stochastic Gradient Descent (SGD), Support Vector Machine (SVM), Linear Support Vector Machine (Linear SVM), K-Nearest Neighbor (KNN), and Decision Tree (DT). Ahmed et al. [13] use a dataset called ISOT Fake News where it can be accessed publicly, and they obtained an accuracy of 92% using the Linear SVM classifier. Moreover, Ozbay and Alatas [14] only used TF–IDF as their feature extraction method, a slightly different approach from the study [13]. Ozbay and Alatas [14] also tried to use 23 classifiers such as ZeroR, CV Parameter Selection (CVPS), Weighted Instances Handler Wrapper (WIHW), DT, and so on to detect fake news. They reported that their approach outperformed the results in [13] by obtaining accuracy, precision, recall, and F1-scores of 96.8%, 96.3%, 97.3%, and 96.8%.

Ahmad et al. [15] also conducted similar research to [13,14]. They compared individual learning algorithms and ensemble learning algorithms performance. Ahmad et al. [15] tested the performance of the Logistic Regression (LR), LSVM, Multilayer Perceptron (MLP), and KNN algorithms individually. They then compared them with ensemble learning such as Random Forest (RF), Voting Classifier, Bagging Classifier, and Boosting Classifier. Moreover, Ahmad et al. [15] compared those methods using several datasets, such as ISOT Fake News Dataset [13], Fake News Dataset [16], Fake News Detection Dataset [17], and a dataset formed from a combination of them. The results obtained from testing on the first dataset outperformed the study results [14], with accuracy, precision, recall, and F1-scores respectively by 99%, 99%, 100%, and 99% using the RF algorithm. The RF algorithm also gives good results on the third and fourth datasets as evidenced by the accuracy, precision, recall, and F1-scores of 95%, 98%, 93%, and 95% in the third dataset and 91%, 92%, 91%, and 91% in the fourth dataset, sequentially. In addition, Ahmad et al. study [15] obtained 94%, 94%, 95%, and 94% results for accuracy, precision, recall, and F1-score, respectively, in the second dataset using the Bagging Classifier algorithm combined with DT.

Kaliyar et al. [18] proposed a different approach to detecting fake news than [13–15]. Kaliyar et al. [18] chose a pre-trained word embedding called Glove, which was later combined with a Convolutional Neural Network (CNN), rather than TF–IDF and classical machine learning algorithms. They use Fake News Dataset [16], and as a result, their proposed method is outperformed Ahmad et al. study [15], where the accuracy, precision, recall, and F1-score are 98.36%, 99.40%, 96.88%, and 98.12%, respectively.

Research on fake news detection using the Fake News Detection Dataset [17] has also been conducted by Bahad et al. [19], even before the study of Ahmad et al. [15]. This research also uses GloVe pre-trained word embedding. It combines it with several deep learning architectures such as CNN, Recurrent Neural Network (RNN), Unidirectional Long Short-Term Memory (LSTM), and Bidirectional LSTM. The results obtained from these studies were varied. One of them was superior to Ahmad et al. [15] research results with a value of 98.75% accuracy using the Bidirectional LSTM. The study [19] also tested it using the Fake or Real News Dataset [20] and obtained 91.48% accuracy using Unidirectional LSTM.

One year after the study in [19], Deepak & Chitturi [21] conducted a similar study using the same dataset, namely Fake or Real News Dataset [20]. Deepak & Chitturi [21] examine secondary features such as news domains, news writers, and headlines to measure fake news detection performance. Then, they utilize word embeddings such as a Bag of Words (BoW), Word2Vec, and GloVe combined with a Feed-forward Neural Network (FNN) and LSTM. As a result, Deepak & Chitturi [21] reported that the use of secondary features positively affects an increase in performance. When the secondary features were added, the FNN accuracy increased by 1%, from 83.3% to 84.3%. Surprisingly, the LSTM accuracy boosted significantly by 7.6% from 83.7% to 91.3% with these secondary features. Although there is a significant increase in performance, especially in LSTM, the results are still not superior to the study [12].

3. Research method

This research consists of two main phases. The first phase is the training phase, as shown in Fig. 1 begins by retrieving training data stored in the database. The training data then goes through a data cleansing process to clean up poor-quality data. The data augmentation process is then applied to the cleaned data to balance the data between the classes. The augmented data is then pre-processed and transformed into word vectors using pre-trained word embedding. Word vector generated by pre-trained word embedding then used to train the deep learning model: CNN, Bidirectional LSTM, and ResNet. Finally, the trained model is then stored in the database to be used in the testing phase.

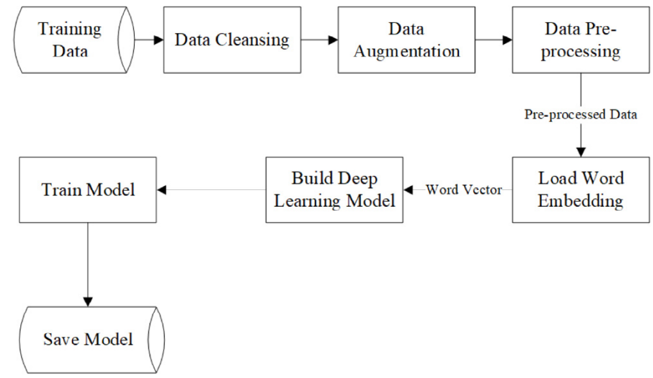

Fig. 1. Training phase.

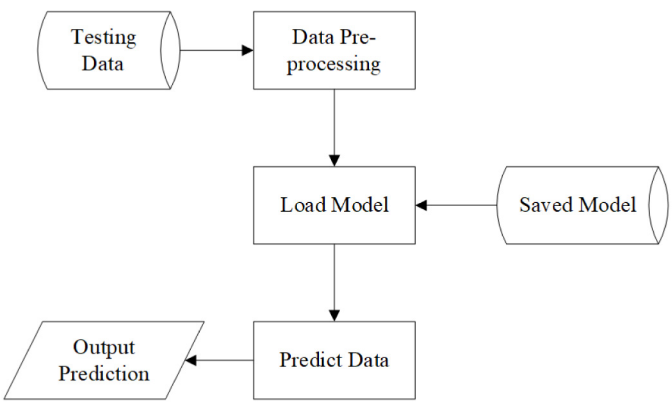

Fig. 2. Testing phase.

The second phase can be seen in Fig. 2. This phase is the testing process to evaluate the trained model, it is carried out by first taking the test data from the database. The test data goes straight to the same pre-processing stage as the training phase without going through the data cleansing and augmentation stages. The previously trained model is then taken from the database and used to predict the pre-processed test data. Finally, the prediction results will be displayed and used to evaluate the model's performance.

3.1. Data cleansing

Data cleansing or data cleaning is the process of correcting or removing low-quality data from the database [22]. Data cleansing in this study was carried out by deleting data with no content or label because it could interfere with the analysis and decision-making process.

3.2. Data augmentation

Data augmentation is a process that is usually performed to balance a dataset by creating synthetic data using the information in that dataset [23]. Data augmentation is often used in activities involving the learning process to reduce imbalance classes. This study applies back-translation as a data augmentation method where English data is translated into German and then translated back into English. The newly generated data is different from the original data but still has the same meaning. This study chooses Back-translation English ↔ German because the study [24] showed a substantial improvement.

3.3. Data pre-processing

The augmented data will go through pre-processing, where the text data is converted into a more understandable form to simplify the feature extraction process [25]. According to Etaiwi & Naymat [25], pre-processing consists of three stages: punctuation removal, stopword removal, and stemming or lemmatization. Additional steps, such as case folding and number removal, are sometimes carried out depending on the problem [26].

The pre-processing carried out in this study consisted of several stages, namely the tokenization process to facilitate processing [27], then case folding is applied to each word token by converting it to a lower case [28]. Then, characters non-alphabet will be removed from the token because they do not significantly impact the analysis process [29]. Words that are included in the stopword are also drawn to reduce computational load. Finally, lemmatization is carried out to convert each token into its common root word [30].

3.4. Word embedding

Word embedding or distributed word representation is a technique that maps words into number vectors, where words that have similar meanings will be close to each other when visualized [31]. That can be done because word embedding can capture a word's semantic and syntactic information in a vast corpus [32]. Word embedding is increasingly used in sentiment analysis research, entity recognition, part of speech tagging, and other text analysis-based research because it shows promising results [33]. Some examples of pre-trained word embedding are often used, namely Word2Vec, GloVe, and fastText.

Word2Vec [34] is one of the pre-trained word embedding used to obtain a word vector representation. Word2Vec provides two architectural options, namely Continuous Bag of Words (CBOW) and Continuous Skip-gram. CBOW predicts the vector of a word based on the context vector or the words around it regardless of the order of the words, while the skip-gram predicts the surrounding context vector based on the middle word.

Global Vector (GloVe) [35] is also a popular pre-trained word embedding that utilizes a matrix of the number of word occurrences in a corpus and a matrix factorization to obtain a vector representation of a word [36]. The GloVe works by forming a matrix of the number of word occurrences first from a vast corpus. After that, a factorization process is carried out on the matrix to obtain a vector representation of the word [37].

FastText [38] is a pre-trained word embedding that uses the improved CBOW Word2Vec architecture. Improvement was

Fig. 3. The CNN architecture with 16 layers and consists of an input layer, embedding layer, one-dimensional convolutional layers, max-pooling layers, concatenate layers, global max-pooling layer, dense layer, and output layer.

made by using more efficient computational algorithms and word order [39]. FastText also adds subword information [40] using the bag of character n-gram because it has a good impact on vectors representing words that rarely appear or are misspelled [41]. This study examines and compares the effect of pre-trained word embedding, namely Word2Vec, GloVe, and fastText, on the model's performance in detecting fake news.

3.5. Deep learning

Deep learning belongs to the branch of machine learning and is an advancement of classic machine learning. Unlike classic machine learning, which still requires human assistance in extracting its features [42], deep learning has the advantage of automatically learning raw data features. More specific features will be obtained from the formation of more general features [43]. Deep learning can do this because it uses a Deep Neural Network (DNN), consisting of a convolutional layer, pooling layer, and fully connected layer [44]. This study uses a deep learning method to detect fake news. Three different deep learning methods were tested, namely CNN, Bidirectional LSTM, and ResNet, to determine which method has the best performance in detecting fake news.

Convolutional Neural Network (CNN) is a deep learning architecture that uses convolutional layers to map input data features. The layers are arranged by applying different filter sizes to produce different feature mappings. CNN can obtain information about the input data based on the feature mapping results [45]. A pooling layer usually accompanies the convolutional layer to produce the exact output dimensions even with different filters. The pooling layer also lightens the computational load by reducing the output dimensions without losing essential information [46].

The CNN architecture in this study was adapted from the research of Kaliyar et al. [18] and can be seen in Fig. 3. One input layer with 1000 dimensions is connected to the embedding layer, whose dimensions are determined by the pre-trained word embedding dimensions. The embedding layer is connected to three one-dimensional convolution layers with different kernel sizes, where each convolution layer will produce a different feature mapping. Each convolution layer is connected to the max-pooling layer for feature compression and controlling overfitting. A concatenate layer will combine the features obtained by each max-pooling layer. The concatenate layer is then connected with two convolution layers and a max-pooling layer for further feature extraction. Finally, the

Table 1
Datasets used in this study.

Dataset name	Cleaned	Augmented	Pre-processed	Fake news	Real news	Total
ISOT fake news dataset [13]	No	No	No	23.502	21.417	44.919
Fake news dataset [16]	No	No	No	10.413	10.387	20.800
Fake or real news dataset [20]	No	No	No	3.154	3.161	6.315
Fake news detection dataset [17]	No	No	No	2.135	1.870	4.005

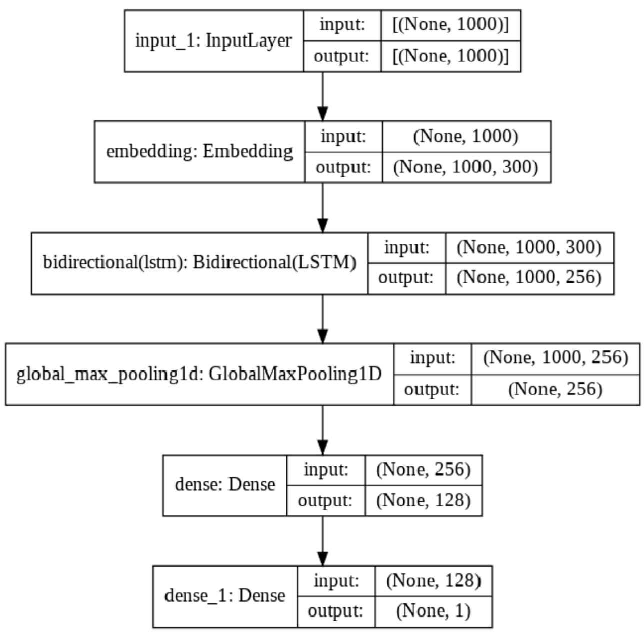

Fig. 4. Bidirectional LSTM architecture. It consists of an input, embedding, bidirectional LSTM, global max-pooling, dense, and output layers.

global max-pooling layer is connected to the fully connected layer and the output layer.

Bidirectional LSTM is a Recurrent Neural Network (RNN) type architecture consisting of two Long Short-Term Memory (LSTM) positioned in different directions. The architecture aims to improve the memory capabilities of LSTMs by providing context information from the past and future [47].

Fig. 4 describes the Bidirectional LSTM architecture used in this study, where there is an input layer of 1000 dimensions, followed by an embedding layer. The architecture continues with the Bidirectional LSTM layer, where the LSTM is identical for both directions. Furthermore, the global max-pooling layer, fully connected layer, and the output layer predict the class.

Residual Network (ResNet) [48] is a deep learning architecture that utilizes a residual block to minimize vanishing gradients. The residual block can forward the signal directly to a layer without passing through the previous layers [49].

ResNet architecture that used in this study is shown in Fig. 5. ResNet architecture is built using CNN architecture in Fig. 3, where the two sets of convolution layers and the max-pooling layer are replaced with residual block. The addition of this residual block is expected to increase architectural resistance to vanishing gradient and make the previously learned information re-learned and last longer.

4. Experiment setup

The experiments in this study were carried out using four different datasets, which were also used in previous studies, namely ISOT Fake News Dataset [13], Fake News Dataset [16], Fake or Real News Dataset [20], and Fake News Detection Dataset [17]. The characteristics of each dataset described in Table 1 indicate that the four datasets have different amounts of data. That can be useful in determining the ability of a deep learning model to handle large or small amounts of data. Unfortunately, each dataset has not gone through the stages of data cleansing, data augmentation, and data pre-processing because there are still empty data rows, data imbalances, remaining stop words, and affixed words. Therefore, the process of data cleaning, data augmentation, and data pre-processing is necessary so that the data is ready for further analysis. Datasets that has been cleaned, augmented and pre-processed are also publicly accessible [50].

Experiments were carried out using the help of the TensorFlow, NLTK, pandas, and scikit-learn libraries and devices that have an Intel Core i7-7700HQ CPU, NVIDIA GeForce GTX1050 GPU, and 16 GB RAM. Some of the tests carried out in this study to determine the effect on the performance of the resulting model, including testing the data augmentation method, optimizer method, batch size hyperparameter, and final testing. Each value in testing is examined on four different datasets using three deep learning methods, namely CNN, Bidirectional LSTM, and ResNet, combined with one of the three pre-trained word embeddings such as Word2Vec, GloVe, and fastText. The evaluation process of each value in testing uses visualization in a box or bar plot. The data augmentation method tests two values, namely data with augmentation and data without augmentation. There are seven methods tested in the optimizer method: SGD, RMSprop, Adam, Adadelta, Adagrad, Adamax, and Nadam. The next test is the batch size hyperparameter, where five values are tested, specifically 32, 64, 128, 256, and 512. Finally, a test is carried out that combines the values or methods that have been selected from previous tests to determine the final performance of the deep learning model.

5. Result and discussions

5.1. Data augmentation

The first test was conducted to determine the effects of the data augmentation process. The box plot, shown in Fig. 6, is used to visualize the data augmentation method's test results. The augmented data have higher minimum values than the data without augmentation by 3.8%. Besides that, the augmented

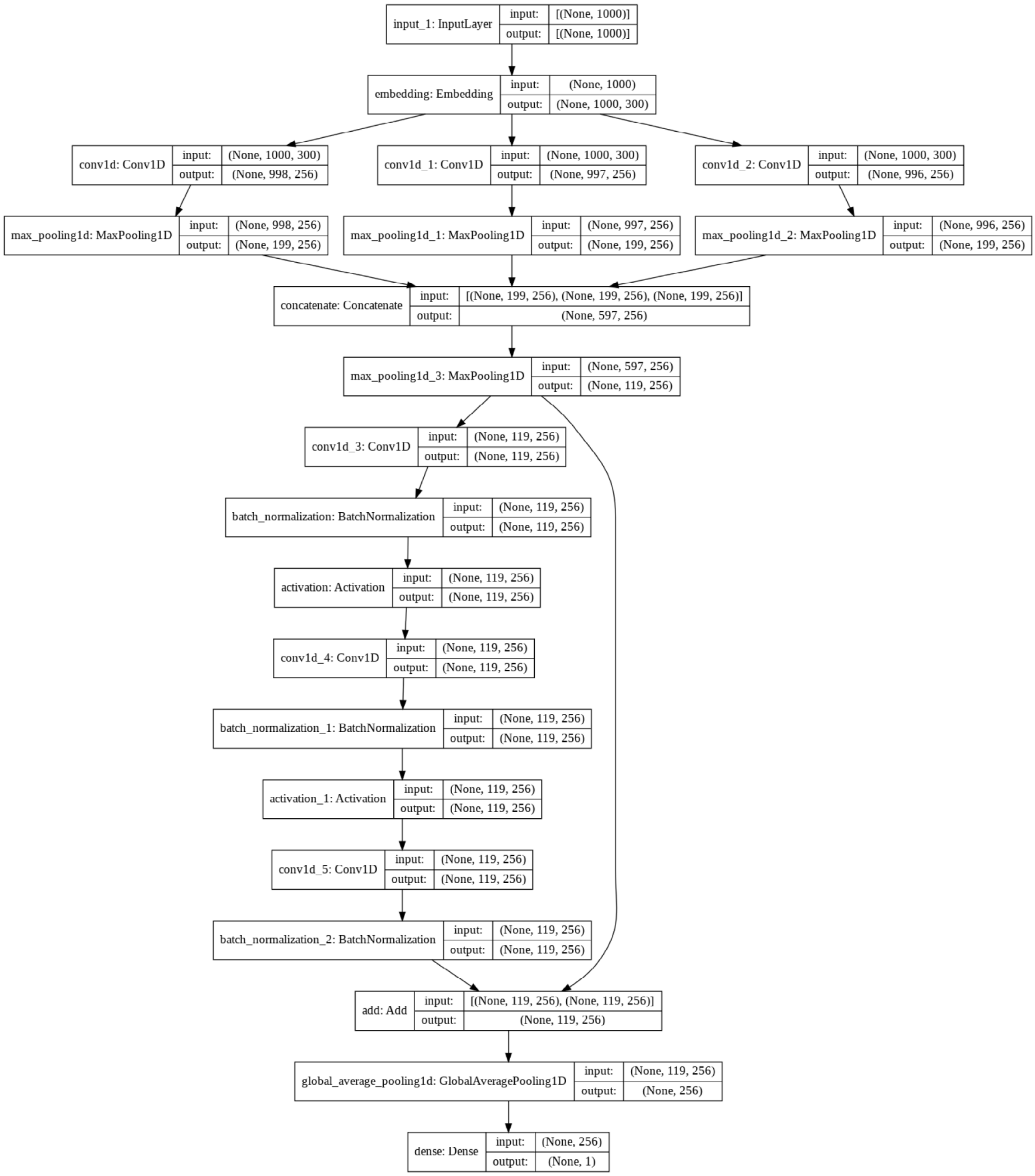

Fig. 5. ResNet architecture with 21 layers. It is constructed by input, embedding, convolutional, max pooling, concatenate, batch normalization, activation, add, global average pooling, and output layers.

data also has higher first and second quartile values by 0.7% and 0.1%. Even though the third and fourth quartile values were slightly lower, the augmented data has a higher mean and smaller box sizes, indicating more concentrated and consistent results.

5.2. Hyperparameter tuning

The hyperparameter tuning process is carried out by testing two values, namely the optimizer and batch size, to improve the performance of the deep learning model. Fig. 7 shows the results of testing the optimizer method in the form of a bar plot. Adamax gets the highest performance with a lower loss rate, so Adamax is used as the optimizer method for the final test. Finally, the batch size test results in Fig. 8 show that the batch size 64 value is the most optimal by obtaining the highest value at the minimum, first quartile, median, average, and maximum values.

The final test process involves four different datasets. It is carried out on each of the deep learning architectures that have been compiled using the selected hyperparameter then combined with pre-trained word embedding. Evaluation metrics,

Table 2
Comparison of the proposed methods with the state of the arts on ISOT fake news dataset.

Author	Word embedding model	Classification model	Accuracy	Precision	Recall	F1-score
Ahmed et al. [13]	TF–IDF	Linear SVM	92%	–	–	–
Ozbay & Alatas [14]	TF–IDF	Decision tree	96.8%	96.3%	97.3%	96.8%
Ahmad et al. [15]	LIWC	Random forest	99%	99%	100%	99%
Proposed model	fastText	CNN	99.88%	99.89%	99.88%	99.88%
Proposed model	GloVe	ResNet	99.90%	99.91%	99.90%	99.90%
Proposed model	**GloVe**	**Bidirectional LSTM**	**99.95%**	**99.95%**	**99.95%**	**99.95%**

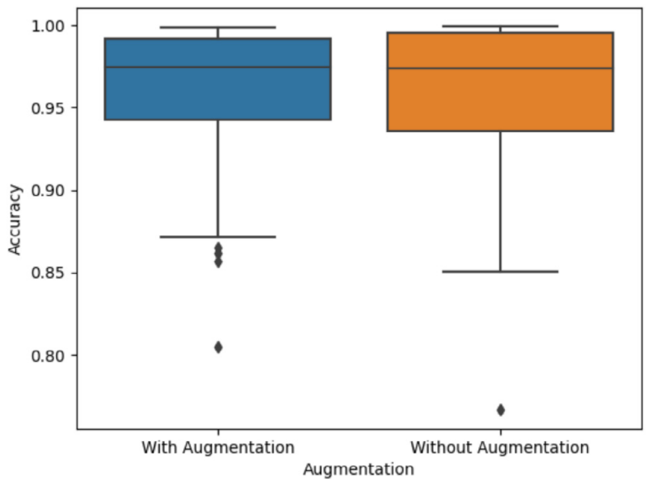

Fig. 6. Data augmentation result.

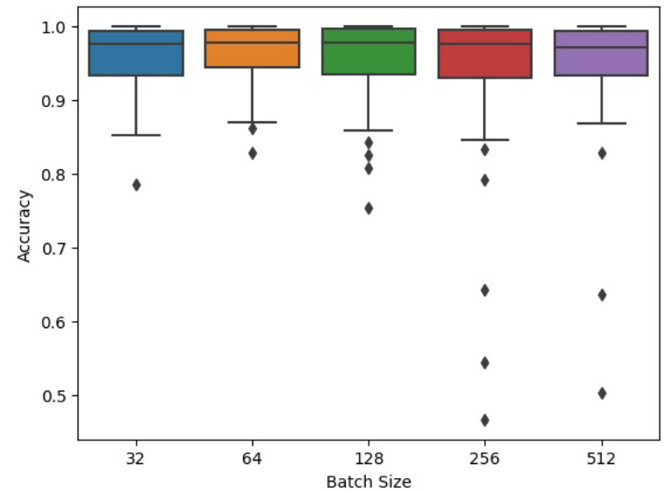

Fig. 8. Batch size result.

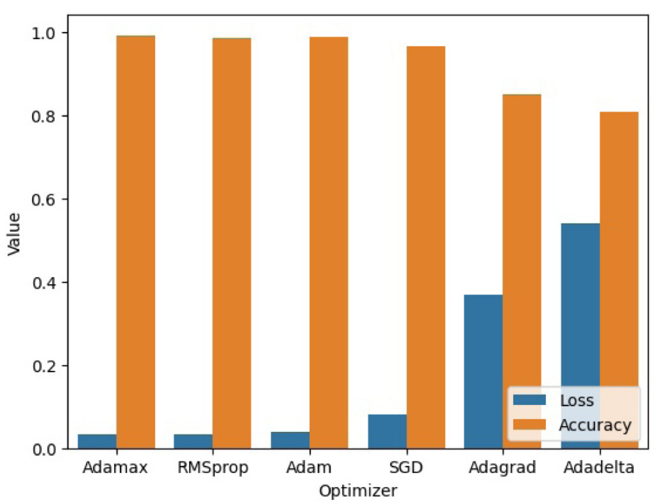

Fig. 7. Optimizer result.

such as accuracy, precision, recall, F1-score, and confusion matrix, are used to assess the model's performance.

5.3. ISOT fake news dataset

ISOT Fake News Dataset [13] is the first dataset used in the final test. The test results on the ISOT Fake News Dataset [13] are shown in Fig. 9 in the form of a comparison of training time and test performance and a confusion matrix. Based on Fig. 9(a), the training process using ISOT Fake News Dataset [13] takes 280 s to 706 s, where CNN is the fastest and Bidirectional LSTM is the longest. In the test performance shown in Fig. 9(b), Bidirectional LSTM outperforms the other two models by obtaining the highest at minimum to the maximum value. A summary of the prediction results of each model is shown in the form of a confusion matrix in Fig. 9(c), (d), and (e), where each model more predicts news as fake news.

The test results comparison on the ISOT Fake News Dataset [13] described in Table 2 shows that the proposed models outperformed other studies [13–15]. Bidirectional LSTM + GloVe model obtaining 99.95% accuracy, precision, recall, and F1-score. The ResNet + GloVe model followed with a performance of 99.91% for precision and 99.90% for accuracy, recall, and F1-score. Finally, the CNN + fastText model has a performance gain of 99.89% precision and 99.88% for accuracy, recall, and F1-score.

5.4. Fake news dataset

The second dataset used in the final test was the Fake News Dataset [16], where the results and the comparison can be seen in Fig. 10 and Table 3. The training process using Fake News Dataset [16] shown in Fig. 10(a) takes 190 s to 387 s, where CNN is the fastest and Bidirectional LSTM is the longest. Bidirectional LSTM still outperforms the other two models by obtaining the highest at minimum to the maximum value in the test performance shown in Fig. 10(b). The confusion matrix of Bidirectional LSTM on Fig. 10(d) also has minor

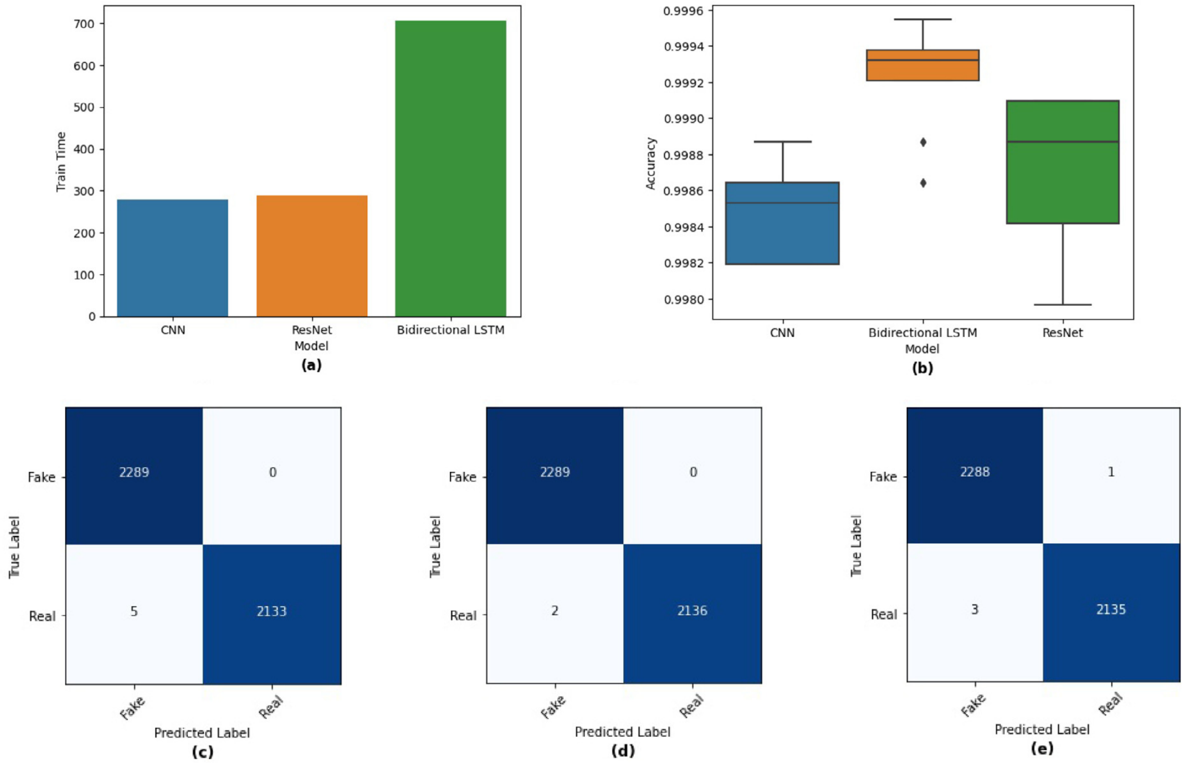

Fig. 9. Experiment results on ISOT Fake News Dataset: (a) training time for each deep learning methods, (b) the accuracies for all methods in box plot, (c) CNN confusion matrix, (d) Bidirectional LSTM confusion matrix, (e) ResNet confusion matrix.

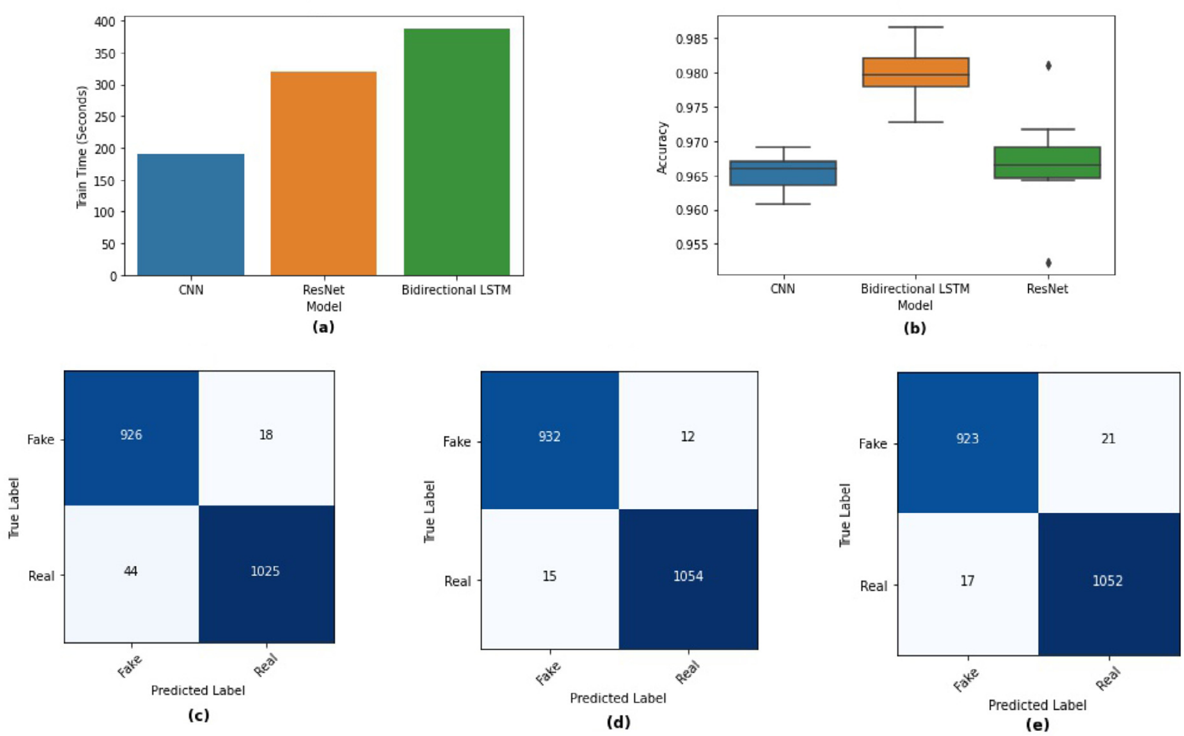

Fig. 10. Experiment results on Fake News Dataset: (a) training time for each deep learning methods, (b) the accuracies for all methods in box plot, (c) CNN confusion matrix, (d) Bidirectional LSTM confusion matrix, (e) ResNet confusion matrix.

Table 3
Comparison of the proposed methods with the state of the arts on fake news dataset.

Author	Word embedding model	Classification model	Accuracy	Precision	Recall	F1-score
Ahmad et al. [15]	LIWC	Bagging + Decision tree	94%	94%	95%	94%
Proposed model	fastText	CNN	96.92%	96.86%	96.98%	96.91%
Proposed model	fastText	ResNet	98.11%	98.11%	98.09%	98.1%
Kaliyar et al. [18]	GloVe	CNN	98.36%	99.4%	96.88%	98.12%
Proposed model	**fastText**	**Bidirectional LSTM**	**98.65%**	**98.64%**	**98.66%**	**98.65%**

Table 4
Comparison of the proposed methods with the state of the arts on fake or real news dataset.

Author	Word embedding model	Classification model	Accuracy	Precision	Recall	F1-score
Proposed model	fastText	ResNet	88.88%	89.36%	89.11%	88.88%
Deepak & Chitturi [21]	Word2Vec	LSTM	91.3%	–	–	–
Bahad et al. [19]	GloVe	Unidirectional LSTM	91.48%	–	–	–
Proposed model	fastText	CNN	91.9%	91.88%	91.93%	91.89%
Proposed model	**GloVe**	**Bidirectional LSTM**	**94.6%**	**94.58%**	**94.64%**	**94.59%**

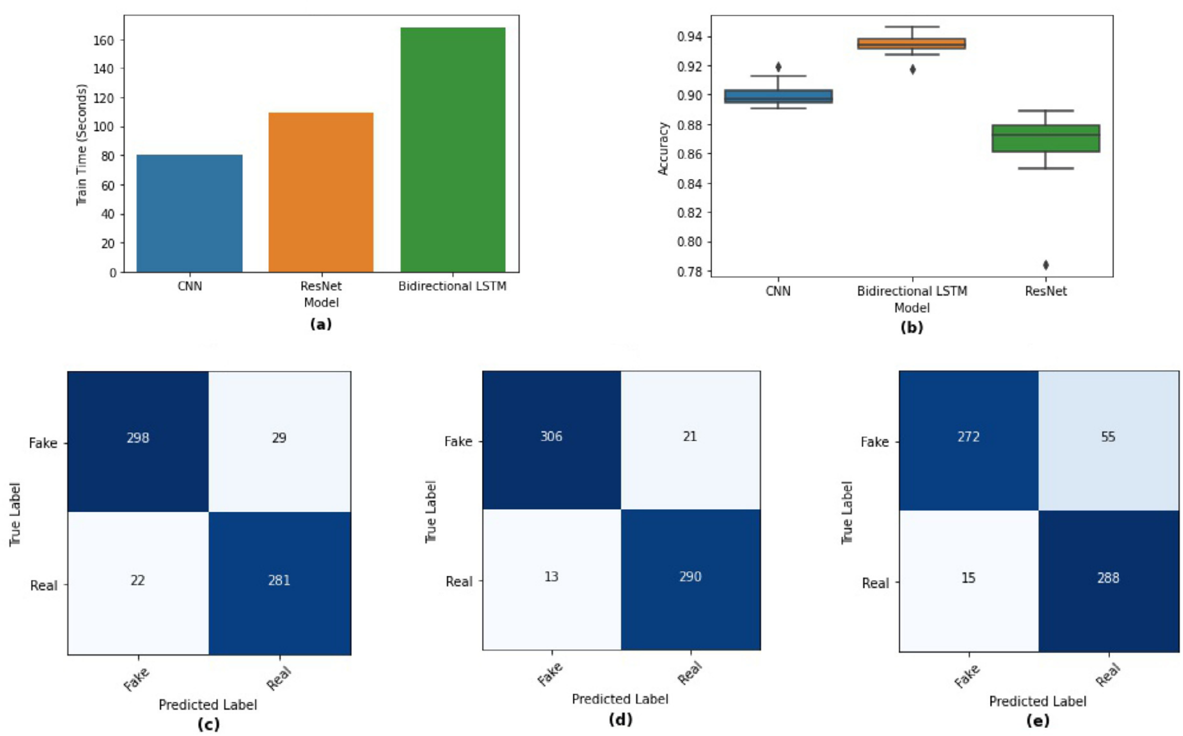

Fig. 11. Experiment results on Fake or Real News Dataset: (a) training time for each deep learning methods, (b) the accuracies for all methods in box plot, (c) CNN confusion matrix, (d) Bidirectional LSTM confusion matrix, (e) ResNet confusion matrix.

False Positive (FP), and False Negative (FN) compared to the other model's confusion matrix on Fig. 10(c) and (e).

The comparison of results on the Fake News Dataset [16] described in Table 3 shows that the proposed models with Bidirectional LSTM outperformed other studies [15,18]. The test results show that the Bidirectional LSTM + fastText model provides the best performance with accuracy and an F1-score of 98.65%, 98.64% precision, and a recall of 98.66%. The ResNet + fastText model obtains performance of 98.11% for accuracy and precision, 98.09% recall, and 98.1% F1-score, lower than Kaliyar et al. [18] result. The last proposed model with higher performance than Ahmad et al. [15] has 96.92%, 96.86%, 96.98%, and 96.91% accuracy, precision, recall, and F1-score, respectively, are owned by the CNN + fastText model.

5.5. Fake or real news dataset

Fake or Real News Dataset [20] is the third dataset used in the final test, where the results and the comparison can be seen in Fig. 11 and Table 4. Based on Fig. 11(a), the Fake or Real News Dataset [20] training process takes 80 s to 168 s, where CNN is the fastest and Bidirectional LSTM is the longest. In the test performance shown in Fig. 11(b), Bidirectional LSTM obtaining the highest at minimum to the maximum value with

Table 5
Comparison of the proposed methods with the state of the arts on fake news detection dataset.

Author	Word embedding model	Classification model	Accuracy	Precision	Recall	F1-score
Ahmad et al. [15]	LIWC	Random forest	95%	98%	93%	95%
Proposed model	GloVe	CNN	98.24%	98.32%	98.09%	98.2%
Bahad et al. [19]	GloVe	Bidirectional LSTM	98.75%	–	–	–
Proposed model	GloVe	ResNet	98.99%	99.05%	98.89%	98.97%
Proposed model	**fastText**	**Bidirectional LSTM**	**99.24%**	**99.19%**	**99.26%**	**99.23%**

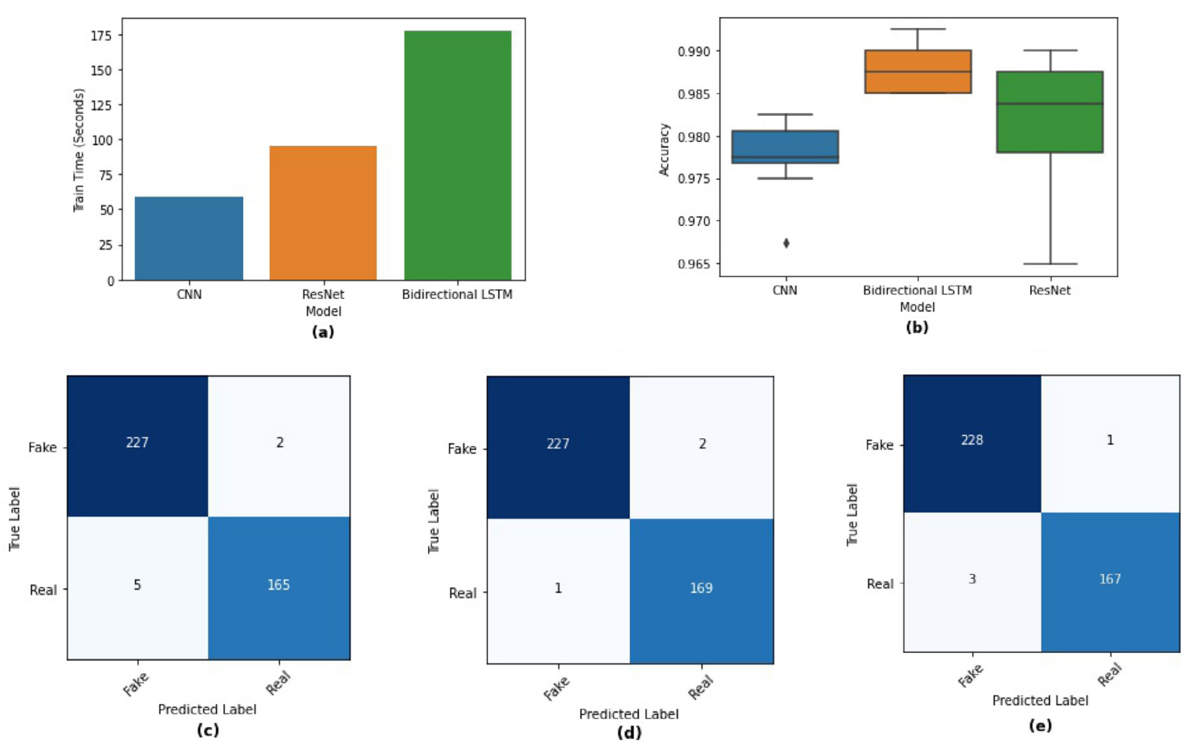

Fig. 12. Experiment results on Fake News Detection Dataset: (a) training time for each deep learning methods, (b) the accuracies for all methods in box plot, (c) CNN confusion matrix, (d) Bidirectional LSTM confusion matrix, (e) ResNet confusion matrix.

a lower FP and FN shown in Fig. 11(d) compared to the other model's confusion matrix in Fig. 11(c) and (e).

The test results comparison on the Fake or Real News Dataset [20] described in Table 4 shows that the proposed models with CNN and Bidirectional LSTM outperformed other studies [19,21]. The Bidirectional LSTM + GloVe model has the highest performance of 94.6% accuracy, 94.58% precision, 94.64% recall, and 94.59% F1-score. That performance was followed by the CNN + fastText model with accuracy, precision, recall, and F1-score values of 91.9%, 91.88%, 91.93%, and 91.89%. The lowest performance was obtained by the ResNet + fastText model with accuracy, precision, recall, and F1-score values of 88.88%, 89.36%, 89.11%, and 88.88%.

5.6. Fake news detection dataset

The last dataset used in the final test was the Fake News Detection Dataset [17]. The test results on the Fake News Detection Dataset [17] are shown in Fig. 12 in the form of a comparison of training time and test performance and a confusion matrix. The training process using Fake News Detection Dataset [17] shown in Fig. 12(a) takes 59 s to 177 s, where CNN is the fastest and Bidirectional LSTM is the longest. In the test performance shown in Fig. 12(b), Bidirectional LSTM still obtaining the highest at minimum to the maximum value. The confusion matrix of Bidirectional LSTM in Fig. 12(d) also has lower FP and FN compared to the other model's confusion matrix in Fig. 12(c) and (e).

The test results comparison on the Fake News Detection Dataset [17] described in Table 5 shows that the proposed models with ResNet and Bidirectional LSTM outperformed other studies [15,19]. The Bidirectional LSTM + fastText model again outperforms other models with a performance of 99.24% accuracy, 99.19% precision, 99.26% recall, and 99.23% F1-score. The test results also show that ResNet + GloVe is in second place with a performance of 98.99% accuracy, 99.05% precision, 98.89% recall, and 98.97% F1-score. The CNN + GloVe can only outperform Ahmad et al. [15] result with an accuracy value of 98.24%, 98.32% precision, 98.09% recall, and an F1-score of 98.2%.

Table 6 shows some examples of fake news and the prediction of each model toward the news. The first news has

Table 6
Examples of fake news.

News	Models	Prediction
Question dr ron paul outer limit radio dr paul serve twelve term u house representative threetime candidate u president devote political career defense individual liberty sound money noninterventionist foreign policy judge andrew napolitano call thomas jefferson day serve flight surgeon u air force dr paul move texas begin civilian medical practice deliver four thousand baby career obstetrician dr paul serve congress carol paul wife fifty year five child many grandchild greatgrandchildren ron paul new york post write politician buy special interest people public life thick thin rain shine stick principles added congressional colleague ron paul one ron paul never vote legislation unless propose measure expressly authorize constitution also congressman ron paul never vote raise tax ron paul never vote unbalanced budget ron paul never vote federal restriction gun ownership ron paul never vote raise congressional pay ron paul never take governmentpaid junket ron paul never vote increase power executive branch ron paul vote patriot act ron paul vote regulate internet ron paul vote iraq war ron paul participate lucrative congressional pension program ron paul chairman ron paul institute peace prosperity nonprofit educational charity host daily ron paul liberty report special thanks daniel mcadams chris rossini dylan charles chris duane mp audio link please check dr paul late book revolution ten year learn out limit radio website	CNN Bidirectional LSTM ResNet	Real Fake Fake
Suge knight claim tupac still alive twentyone year ago legendary u rapper tupac amaru shakur die shot street la vega september since conspiracy theory image repeatedly emerge prove rapper still alive man sit car tupac shot fire speaks make bizarre story year interview american tv station fox music mogul suge explain life imprisonment murder explain former best friend live anymore suge knight interview leave hospital pac laugh make joke ca nt understand someone state health change good bad ask believe rapper still alive suge reply tell never know pac people believe death incidentally music mogul alone opinion tupac shakur could still alive former policeman recently claim pay million euro fake tupac death official put follow word record world need know do ashamed exchange honesty money ca nt die without world knowing include popular story tupac tupac life cuba godmother political activist assata olugbala shakur suge knight kill tupac tupac back rapper kasinova tha spread music p diddy notorious big order tupac murder tupac member illuminati illuminati kill tupac become powerful tupac abducted alien fbi kill tupac related article suge knight finally admit tupac alive video elvis presley alive people convince elvis alive new photo emerge	CNN Bidirectional LSTM ResNet	Fake Fake Real
Cop try failed sue blacklivesmatter cop try failed sue blacklivesmatter reader think story fact add two cent news louisiana police officer try sue blacklivesmatter damage judge hold back let know ridiculous claim source httpwwwcarbonatedtvnewscoptriedtosueblacklivesmatterandwasscoldedinstead	CNN Bidirectional LSTM ResNet	Fake Fake Fake

long content and uses a proper noun, Bidirectional LSTM and ResNet successfully classified the news as fake news, but CNN failed due to the content length and the noun because fake news tends to have short content and use a common noun.

The second news also has long content and using proper nouns, reported speech and provocative sentences. CNN and Bidirectional LSTM successfully classified the news as fake news due to the provocative sentence, but ResNet failed.

The last news has short content, external link, and common nouns so that each model can easily classify the news as fake news.

6. Conclusions

This study applies a data augmentation process to each dataset using the back-translation method to reduce the class imbalance. Tests are conducted to determine the impact of data augmentation on the performance of the resulting model. The test results show that data augmentation has a positive effect, especially in improving model performance consistency.

Several deep learning methods such as CNN, Bidirectional LSTM, and ResNet were also evaluated in this study using four different datasets. Each deep learning method is combined with pre-trained word embeddings, namely Word2Vec, GloVe, and fastText. Based on the test results, Bidirectional LSTM outperformed CNN and ResNet on all test datasets. GloVe and fastText also gave good results because they each could excel in two different datasets.

We evaluate the combination of deep learning methods with popular word embedding in depth using several datasets. The datasets have gone through a cleansing, augmentation, and pre-processing process and is publicly accessible. In the future, we would like to implement our method to detect fake news in Indonesia because there is still much room for improvement in the current Indonesian fake news detection system. Further data collection and adjustments to Indonesian text processing methods are needed to address these challenges.

CRediT authorship contribution statement

I. Kadek Sastrawan: Writing - original draft, Writing - review & editing. **I.P.A. Bayupati:** Writing - original draft, Writing - review & editing. **Dewa Made Sri Arsa:** Writing - original draft, Writing - review & editing.

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Acknowledgment

All authors approved the version of the manuscript to be published.

References

[1] P.C. Verhoef, T. Broekhuizen, Y. Bart, A. Bhattacharya, J. Qi Dong, N. Fabian, M. Haenlein, Digital transformation: A multidisciplinary reflection and research agenda, J. Bus. Res. 122 (2019) 889–901, http://dx.doi.org/10.1016/j.jbusres.2019.09.022.

[2] Z. Zhang, R. Sun, C. Zhao, J. Wang, C.K. Chang, B.B. Gupta, CyVOD: A novel trinity multimedia social network scheme, Multimedia Tools Appl. 76 (2017) 18513–18529, http://dx.doi.org/10.1007/s11042-016-4162-z.

[3] L. Zhang, Z. Zhang, T. Zhao, A novel spatio-temporal access control model for online social networks and visual verification, Int. J. Cloud Appl. Comput. 11 (2021) 17–31, http://dx.doi.org/10.4018/IJCAC.2021040102.

[4] S. Srinivasan, L.D. Dhinesh Babu, A parallel neural network approach for faster rumor identification in online social networks, Int. J. Semant. Web Inf. Syst. 15 (2019) 69–89, http://dx.doi.org/10.4018/IJSWIS.2019100105.

[5] M. Takayasu, K. Sato, Y. Sano, K. Yamada, W. Miura, H. Takayasu, Rumor diffusion and convergence during the 3.11 earthquake: A Twitter case study, PLoS One 10 (2015) e0121443, http://dx.doi.org/10.1371/journal.pone.0121443.

[6] S. Wen, M.S. Haghighi, C. Chen, Y. Xiang, W. Zhou, W. Jia, A sword with two edges: Propagation studies on both positive and negative information in online social networks, IEEE Trans. Comput. 64 (2015) 640–653, http://dx.doi.org/10.1109/TC.2013.2295802.

[7] S.R. Sahoo, B.B. Gupta, Hybrid approach for detection of malicious profiles in twitter, Comput. Electr. Eng. 76 (2019) 65–81, http://dx.doi.org/10.1016/j.compeleceng.2019.03.003.

[8] S.R. Sahoo, B.B. Gupta, Multiple features based approach for automatic fake news detection on social networks using deep learning, Appl. Soft Comput. 100 (2021) 106983, http://dx.doi.org/10.1016/j.asoc.2020.106983.

[9] K. Park, H. Rim, Social media hoaxes, political ideology, and the role of issue confidence, Telemat. Inform. 36 (2019) 1–11, http://dx.doi.org/10.1016/j.tele.2018.11.001.

[10] Statista, Level of confidence in distinguishing between real news and fake news among adults in the United States in 2016 and 2019, 2020, https://www.statista.com/statistics/657090/fake-news-recogition-confidence/ (accessed November 20, 2020).

[11] K. Shu, A. Sliva, S. Wang, J. Tang, H. Liu, Fake news detection on social media: A data mining perspective, ACM SIGKDD Explor. Newsl. 19 (2017) 22–36, http://dx.doi.org/10.1145/3137597.3137600.

[12] S. De Paor, B. Heravi, Information literacy and fake news: How the field of librarianship can help combat the epidemic of fake news, J. Acad. Librariansh. 46 (2020) 102218, http://dx.doi.org/10.1016/j.acalib.2020.102218.

[13] H. Ahmed, I. Traore, S. Saad, Detection of Online Fake News using N-Gram Analysis and Machine Learning Techniques, in: Lect. Notes Comput. Sci. (Including Subser. Lect. Notes Artif. Intell. Lect. Notes Bioinformatics), 2017, pp. 127–138, http://dx.doi.org/10.1007/978-3-319-69155-8_9.

[14] F.A. Ozbay, B. Alatas, Fake news detection within online social media using supervised artificial intelligence algorithms, Phys. A Stat. Mech. Appl. 540 (2020) 123174, http://dx.doi.org/10.1016/j.physa.2019.123174.

[15] I. Ahmad, M. Yousaf, S. Yousaf, M.O. Ahmad, Fake news detection using machine learning ensemble methods, Complexity 2020 (2020) 1–11, http://dx.doi.org/10.1155/2020/8885861.

[16] UTK Machine Learning Club, Fake news, 2018, https://www.kaggle.com/c/fake-news/overview (accessed June 3, 2020).

[17] Jruvika, Fake news detection, 2017, https://www.kaggle.com/jruvika/fake-news-detection/version/1 (accessed June 3, 2020).

[18] R.K. Kaliyar, A. Goswami, P. Narang, S. Sinha, FNDNet – A deep convolutional neural network for fake news detection, Cogn. Syst. Res. 61 (2020) 32–44, http://dx.doi.org/10.1016/j.cogsys.2019.12.005.

[19] P. Bahad, P. Saxena, R. Kamal, Fake news detection using bi-directional LSTM-recurrent neural network, Procedia Comput. Sci. 165 (2019) 74–82, http://dx.doi.org/10.1016/j.procs.2020.01.072.

[20] G. McIntire, Fake or real news, 2017, https://github.com/joolsa/fake_real_news_dataset (accessed June 3, 2020).

[21] S. Deepak, B. Chitturi, Deep neural approach to fake-news identification, Procedia Comput. Sci. 167 (2020) 2236–2243, http://dx.doi.org/10.1016/j.procs.2020.03.276.

[22] W. Ji, L. Wang, Big data analytics based fault prediction for shop floor scheduling, J. Manuf. Syst. 43 (2017) 187–194, http://dx.doi.org/10.1016/j.jmsy.2017.03.008.

[23] J. Lemley, S. Bazrafkan, P. Corcoran, Smart augmentation learning an optimal data augmentation strategy, IEEE Access 5 (2017) 5858–5869, http://dx.doi.org/10.1109/ACCESS.2017.2696121.

[24] R. Sennrich, B. Haddow, A. Birch, Improving neural machine translation models with monolingual data, 2015, ArXiv Prepr. arXiv:1511.06709, http://arxiv.org/abs/1511.06709.

[25] E. Elakiya, N. Rajkumar, Designing preprocessing framework (ERT) for text mining application, in: 2017 Int. Conf. IoT Appl, IEEE, 2017, pp. 1–8, http://dx.doi.org/10.1109/ICIOTA.2017.8073613.

[26] M.J. Denny, A. Spirling, Text preprocessing for unsupervised learning: Why it matters, when it misleads, and what to do about it, Polit. Anal. 26 (2018) 168–189, http://dx.doi.org/10.1017/pan.2017.44.

[27] K. Welbers, W. Van Atteveldt, K. Benoit, Text analysis in r, Commun. Methods Meas. 11 (2017) 245–265, http://dx.doi.org/10.1080/19312458.2017.1387238.

[28] Y.A. Gerhana, A.R. Atmadja, W.B. Zulfikar, N. Ashanti, The implementation of K-nearest neighbor algorithm in case-based reasoning model for forming automatic answer identity and searching answer similarity of algorithm case, in: 2017 5th Int. Conf. Cyber IT Serv. Manag., IEEE, 2017, pp. 1–5, http://dx.doi.org/10.1109/CITSM.2017.8089233.

[29] R. Hafeez, S. Khan, I.A. Khan, M.A. Abbas, Does preprocessing really impact automatically generated taxonomy, in: 2017 13th Int. Conf. Emerg. Technol., IEEE, 2017, pp. 1–6, http://dx.doi.org/10.1109/ICET.2017.8281710.

[30] A.A. Freihat, M. Abbas, G. Bella, F. Giunchiglia, Towards an optimal solution to lemmatization in arabic, Procedia Comput. Sci. 142 (2018) 132–140, http://dx.doi.org/10.1016/j.procs.2018.10.468.

[31] W. López, J. Merlino, P. Rodríguez-Bocca, Learning semantic information from internet domain names using word embeddings, Eng. Appl. Artif. Intell. 94 (2020) 103823, http://dx.doi.org/10.1016/j.engappai.2020.103823.

[32] S. Lai, K. Liu, S. He, J. Zhao, How to generate a good word embedding, IEEE Intell. Syst. 31 (2016) 5–14, http://dx.doi.org/10.1109/MIS.2016.45.

[33] A.B. Soliman, K. Eissa, S.R. El-Beltagy, Aravec: A set of arabic word embedding models for use in arabic NLP, Procedia Comput. Sci. 117 (2017) 256–265, http://dx.doi.org/10.1016/j.procs.2017.10.117.

[34] T. Mikolov, K. Chen, G. Corrado, J. Dean, Efficient estimation of word representations in vector space, 2013, ArXiv Prepr. arXiv:1301.3781, http://arxiv.org/abs/1301.3781.

[35] J. Pennington, R. Socher, C. Manning, Glove: Global vectors for word representation, in: Proc. 2014 Conf. Empir. Methods Nat. Lang. Process., Association for Computational Linguistics, Stroudsburg, PA, USA, 2014, pp. 1532–1543, http://dx.doi.org/10.3115/v1/D14-1162.

[36] S.M. Rezaeinia, A. Ghodsi, R. Rahmani, Improving the accuracy of pre-trained word embeddings for sentiment analysis, 2017, ArXiv Prepr. arXiv:1711.08609, http://arxiv.org/abs/1711.08609.

[37] M. Naili, A.H. Chaibi, H.H. Ben Ghezala, Comparative study of word embedding methods in topic segmentation, Procedia Comput. Sci. 112 (2017) 340–349, http://dx.doi.org/10.1016/j.procs.2017.08.009.

[38] T. Mikolov, E. Grave, P. Bojanowski, C. Puhrsch, A. Joulin, Advances in pre-training distributed word representations, in: Lr. 2018-11th Int. Conf. Lang. Resour. Eval., 2017, pp. 52–55, http://arxiv.org/abs/1712.09405.

[39] R.A. Stein, P.A. Jaques, J.F. Valiati, An analysis of hierarchical text classification using word embeddings, Inf. Sci. (Ny) 471 (2019) 216–232, http://dx.doi.org/10.1016/j.ins.2018.09.001.

[40] P. Bojanowski, E. Grave, A. Joulin, T. Mikolov, Enriching word vectors with subword information, Trans. Assoc. Comput. Linguist. 5 (2017) 135–146, http://dx.doi.org/10.1162/tacl_a_00051.

[41] B. Wang, A. Wang, F. Chen, Y. Wang, C.C.J. Kuo, Evaluating word embedding models: Methods and experimental results, APSIPA Trans. Signal Inf. Process. 8 (2019) http://dx.doi.org/10.1017/ATSIP.2019.12.

[42] A. Chowanda, A.D. Chowanda, Recurrent neural network to deep learn conversation in Indonesian, Procedia Comput. Sci. 116 (2017) 579–586, http://dx.doi.org/10.1016/j.procs.2017.10.078.

[43] Y. Lecun, Y. Bengio, G. Hinton, Deep learning, Nature 521 (2015) 436–444, http://dx.doi.org/10.1038/nature14539.

[44] X. Zhang, L. Wang, Y. Su, Visual place recognition: A survey from deep learning perspective, Pattern Recognit. (2020) 107760, http://dx.doi.org/10.1016/j.patcog.2020.107760.

[45] S. Albawi, T.A. Mohammed, S. Al-Zawi, Understanding of a convolutional neural network, in: 2017 Int. Conf. Eng. Technol., IEEE, 2017, pp. 1–6, http://dx.doi.org/10.1109/ICEngTechnol.2017.8308186.

[46] M.M. Lopez, J. Kalita, Deep learning applied to NLP, 2017, ArXiv Prepr. arXiv:1703.03091, http://arxiv.org/abs/1703.03091.

[47] G. Rao, W. Huang, Z. Feng, Q. Cong, LSTM with sentence representations for document-level sentiment classification, Neurocomputing 308 (2018) 49–57, http://dx.doi.org/10.1016/j.neucom.2018.04.045.

[48] K. He, X. Zhang, S. Ren, J. Sun, Deep residual learning for image recognition, in: 2016 IEEE Conf. Comput. Vis. Pattern Recognit., IEEE, 2016, pp. 770–778, http://dx.doi.org/10.1109/CVPR.2016.90.

[49] A.N. Gomez, M. Ren, R. Urtasun, R.B. Grosse, The reversible residual network: Backpropagation without storing activations, 2017, ArXiv Prepr. arXiv:1707.04585, http://arxiv.org/abs/1707.04585.

[50] I.K. Sastrawan, I.P.A. Bayupati, D.M.S. Arsa, Fake news dataset, Mendeley Data V1 (2021) http://dx.doi.org/10.17632/945z9xkc8d.1.