FN-Net: A Deep Convolutional Neural Network for Fake News Detection

1st Kian Long Tan
Faculty of Information Science & Technology
Multimedia University
Jalan Ayer Keroh Lama, 75450, Melaka, Malaysia.
1181300023@student.mmu.edu.my

2nd Chin Poo Lee
Faculty of Information Science & Technology
Multimedia University
Jalan Ayer Keroh Lama, 75450, Melaka, Malaysia.
cplee@mmu.edu.my

3rd Kian Ming Lim
Faculty of Information Science & Technology
Multimedia University
Jalan Ayer Keroh Lama, 75450, Melaka, Malaysia.
kmlim@mmu.edu.my

2021 9th International Conference on Information and Communication Technology (ICoICT) | 978-1-6654-0447-1/21/$31.00 ©2021 IEEE | DOI: 10.1109/ICoICT52021.2021.9527500

***Abstract*—Information and communication technology has evolved rapidly over the past decades, with a substantial development being the emergence of social media. It is the new norm that people share their information instantly and massively through social media platforms. The downside of this is that fake news also spread more rapidly and diffuse deeper than before. This has caused a devastating impact on people who are misled by fake news. In the interest of mitigating this problem, fake news detection is crucial to help people differentiate the authenticity of the news. In this research, an enhanced convolutional neural network (CNN) model, referred to as Fake News Net (FN-Net) is devised for fake news detection. The FN-Net consists of more pairs of convolution and max pooling layers to better encode the high-level features at different granularities. Besides that, two regularization techniques are incorporated into the FN-Net to address the overfitting problem. The gradient descent process of FN-Net is also accelerated by the Adam optimizer. The empirical studies on four datasets demonstrate that FN-Net outshines the original CNN model.**

Keywords—Fake news, machine learning, natural language processing, CNN

I. Introduction

Since the past decades, social media has gradually become the main source for people to receive news by the virtue of rapidly emerging communication technology. Nowadays, online news is just a click away on social media, such as Facebook, Instagram, Twitter, or any news website. The news is widely spread either real or fake and may bring consequences to the people who are misled by it.

There are several definitions for the term “fake news”. Generally, fake news means deceptive news or propaganda that is used to mislead people or even affect people’s decisions and options [1]. Fake news is created to gain reputation, financial or political benefits. For example, there are some fake reviews intend to influence the consumer to buy their product. If the consumer does not have the basic knowledge of the product, in the end, the consumer may believe the fake review. Hence, fake news detection is needed because most people who do not have the basic knowledge of the information or insufficient time to check the credibility of the news may believe in the news even if it is fake news.

Some organizations have developed the detection tools for fake news, for example, politifact (https://www.politifact.com/) and snopes (https://www.politifact.com/) have their tools to detect the level of fake news. Nevertheless, these tools need manual work so it is costly and time-consuming. In recent years, several challenging tasks are organized to deal with fake news, including Fake News Challenge Stage 1 (http://www.fakenewschallenge.org/), Web Search and Data Mining (WSDM) fake news challenge (https://www.kaggle.com/c/fake-news-pair-classification-challenge), clickbait-challenge (https://webis.de/events/clickbait-challenge/) and etc.

Because of the current trends, this work proposes a FN-Net for fake news detection. Some enhancements are introduced into the original CNN model. FN-Net comprises an additional convolution and max pooling layer to better encode the high-level features at different granularities. In addition, two regularization techniques, namely L2 regularization and dropout are embedded into the model to mitigate the overfitting problem. The training optimization function is also changed from Root Mean Square Propagation (RMSProp) to Adaptive Moment Estimation (Adam) optimizer. Beyond RMSProp, Adam optimizer adapts the learning rate by implementing the momentum as the moving average of the gradient. The Adam optimizer shows faster and smoother convergence than other optimizers. The experimental results and loss plots over the training epoch demonstrate that FN-Net has better stability and less susceptible to overfitting problems.

II. Related Works

From the previous work, traditional machine learning methods and deep neural networks have been applied to many applications [2-7], including fake news detection. The authors in [8] evaluated the performance of three methods on five classification methods. The methods are Term Frequency-Inverse Document Frequency (TF-IDF) bigram, Probabilistic Context-Free Grammar (PCFG) bigram, and the combination of TF-IDF and PCFG bigram. The five classification methods namely Support Vector Machine (SVM), Stochastic Gradient Descent, Gradient Boosting, Bounded Decision Tree, and Random Forests. From the experiments, TF-IDF with Stochastic Gradient Descent achieves the best accuracy of 77.2%.

Three deep learning models were studied in [9], namely Vanilla, gated recurrent units (GRU) and Long Short-Term Memory (LSTM). In their work, text pre-processing such as sentence segmentation and stop words removal were performed. The word embedding representation that captures the word relationships was leveraged as the features. Subsequently, deep learning algorithms were used to classify the news. in the LIAR dataset. The results showed that GRU outperformed Vanilla and LSTM in terms of accuracy, memory optimization and training time.

The authors in [10] built three types of machine learning models, namely naive Bayes with Lidstone smoothing, logistic regression and SVM. They had collected 2136 fake

This research is supported by the Fundamental Research Grant Scheme of the Ministry of Higher Education under the award number FRGS/1/2019/ICT02/MMU/03/7.

news articles and 1872 real news articles. The authors also performed stop words removal on the headline text and body text. The word clouds are used to analyze the frequency and importance of the words. In the experiments, Naive Bayes with Lidstone smoothing used the default setting with no hyperparameter and Lidstone smoothing was used to regularize the model to prevent zero probabilities. SVM selected the normalizing parameter T with value of 12 and set the value of the Lagrange multiplier to 1/64. In logistic regression, the authors set the learning rate to 10 as the learning rate 5 to 12 gave the same results. The experimental results showed that Naive Bayes with Lidstone smoothing model yields the highest accuracy of 83.16%.

In the work by [11], Naive Bayes algorithm was used as the classifier to classify the articles. The author suggested that some phrases or words will always be used in the fake news articles. Hence, finding the probability of the words can help to classify the news. In the implementation, the threshold of the real news in the classifier was set to a value. The news with a probability higher than the threshold value is classified as real news, and vice versa. The research achieved an accuracy of 75.40% using Naïve Bayes algorithm. The paper also discussed some possible improvements, such as using a larger dataset, lengthier articles, removing stop words, using stemming and grouping the words in probability computation.

Paper [12] enhanced the Bidirectional Encoder Representations from Transformers (BERT) to form a BAKE model. The BAKE model mitigates the data imbalance problem by using weighted cross entropy (WCE) to categorize the data. The experiments were separated into two parts, namely pre-training and task-specific fine-tuning. In the pre-training, the BERT model was trained using Wikipedia and Book Corpus datasets. In the model fine-tuning, the Fake News Challenge (FNC-1) dataset was used. The weighted cross entropy (WCE) was used to classify the dataset. The authors further combined extra unlabeled news corpora, that are CNN and Daily Mail news to form an exBAKE model. The exBAKE model achieved a F-1 score of 74.6%.

The authors in [13] used 23 supervised learning methods to classify the fake news datasets collected from the real-world. It included BayesNet, JRip, OneR, Decision Stump, ZeroR, Stochastic Gradient Descent(SGD), CV Parameter Selection (CVPS), Randomizable Filtered Classifier (RFC), Logistic Model Tree (LMT), Locally Weighted Learning (LWL), Classification Via Clustering (CVC), Weighted Instances Handler Wrapper (WIHW), Ridor, Multi-Layer Perceptron (MLP), Ordinal Learning Model (OLM), Simple Cart, Attribute Selected Classifier (ASC), J48, Sequential Minimal Optimization (SMO), Bagging, Decision Tree, IBk, and Kernel Logistic Regression (KLR). The model starts with data pre-processing that includes tokenization, stop words removal, and stemming. Then, they used Computing Term Frequency and created a Document-Term Matrix as the input features to be fed into the classification model. The experimental results showed that Decision Tree, ZeroR, CVPS and WIHW algorithm outshine other methods in comparison.

The authors in [14] proposed a CNN method with TF-IDF vector representation of the properties of the news as the input. Their vector is 300-dimension vector embeddings.

Inspired by the breakthrough of deep learning methods in various domains, this paper proposes an enhanced convolutional neural network for fake news detection.

III. FN-NET: A DEEP CNN FOR FAKE NEWS DETECTION

This section describes the process flow of the fake news detection. Firstly, pre-processing is performed on the raw text to remove noise from the data. Subsequently, the text is tokenized and represented as a word vector. Sequence padding is performed to ensure all texts have the same word vector length. The basic architecture and enhancements of FN-Net are then presented.

A. Preprocessing

The text pre-processing is essential to enable the classifiers focus on the most discriminative features. The noise in the dataset is minimized by removing the unnecessary columns such as ID and date, incomplete rows, punctuations and stop words. The text is thereafter standardized into lowercase to inject case insensitivity into the model.

B. Tokenization and Sequence Padding

Tokenization breaks the text into tokens or words to ease the processing in the later phase. After that, the words are used to create the word index that forms a vocabulary corpus. The length of the word index is set to 1000 to ease computation. Therefore, zero padding is performed when required. Fig. 1 shows the tokenized words in an article and a portion of word index. The output of this step is a vector representing the word index with length 1000. Every index in the vector represents a word.

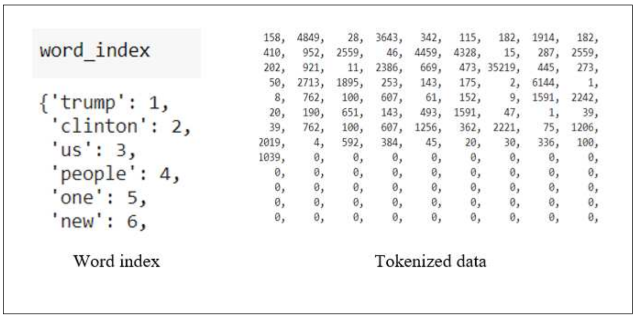

Fig. 1. The tokenized words (left) and the vector representation (right).

C. Word embeddings

Word embeddings encode the semantic relations between words as geometric relationships between vectors, as illustrated in Fig. 2. Hence, words of similar contexts are close to each other, while words of heterogenous contexts are far away from each other. In the word embeddings, every word is represented as a unique vector. Due to the size limitations of the dataset used in this work, the pre-trained weights of Global Vectors for Word Representation (GloVe) [15] with dimension 100 is adopted. There are 400000 of word vectors contained in the Glove 6B vector 100d file.

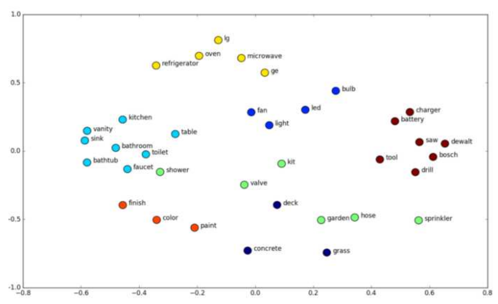

Fig. 2. Example of word embedding in geometric space.

D. *FN-Net*

Inspired by the success of CNN models in computer vision, the model is also adopted for natural language processing tasks. The underlying concepts of CNN model is to extract the high-level features from the data using the convolutional layer. The proposed FN-Net contains an embedding layer, two convolutional and max pooling layers, one global max pooling layer and a series of dense layers. A few enhancements are also introduced into the CNN model to avoid overfitting problem and optimize the training process. Fig. 3 depicts the architecture of FN-Net.

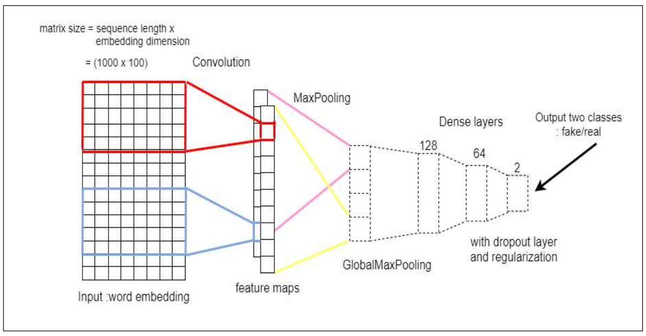

Fig. 3. The architecture of FN-Net.

1) *Embedding layer*

The embedding layer transforms words into their corresponding word embeddings matrix. The vector of word index is fed as the input into the embedding layer. Given the vector of word index with length = 1000 where every index represents a word, the weights of each word are taken from the GloVe pretrained model. The output of this step is the word embeddings matrix with the size 1000×100 where each row represents a word, and the columns encode the weights. Fig. 4 shows sample word embeddings matrix with dimension = 100 and sequence length = 1000.

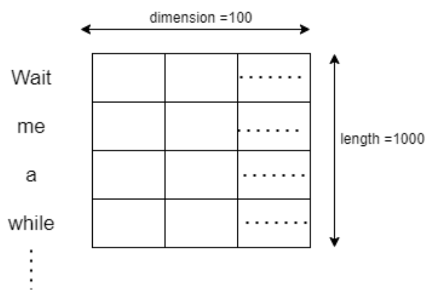

Fig. 4. Word embeddings with sequence length=1000 and dimension=100.

2) *1D Convolutional layer*

The core building block of CNN is convolutional layers. The convolutional layers aim to extract high level meaningful features from the input. A convolution is a linear operation that multiplies the input with a set of weights, known as a filter or a kernel. The filter is applied systematically to each filter-sized patch of the input data. The output from this operation is a feature map. Subsequently, each value in the feature map is passed through a nonlinearity activation function. The convolutional layer in this model involves a set of 128 filters with the size of 5×5.

In this model, a Rectified Linear Unit (ReLU) activation function is used. The main benefit of using the ReLU function is that only the neurons with value ≥ 0 are activated, therefore it is more computationally efficient compared to other activation functions.

3) *Max pooling layer*

The main purpose of max pooling layer is to reduce the dimension of the input for computational efficiency in the subsequent layers. The max pooling is a convolution process where the filter extracts the maximum value in each patch of the feature map it convolves. This model leverages the max pooling filter of size 5×5. The output of this layer is the set of feature maps that are reduced to 20% of the original size.

4) *Global max pooling layer*

In the global max pooling layer, the maximum values are computed in the depth dimension. The purpose of this layer is to flatten the 3D feature map into 2D feature map.

5) *Dense layer*

The dense layers are also known as fully connected layers. These layers connect all the inputs from one layer to every activation unit of the next layer. The last few dense layers in the model compile the data extracted by previous layers to form the final output. The output layer applies the “sigmoid” activation function due to the binary labelling of the data, which is fake or real news.

6) *Enhancements*

Two regularization techniques, namely L2 regularization and dropout are also applied in the dense layers to prevent overfitting. The overfitting problem is normally caused by the over complicated network model. It causes the model to perform very well during training but performs poorly when given unseen data during testing.

The L2 regularization makes the current weights smaller in every update to reduce the impact of the hidden neurons. By doing so, the hidden neurons become negligible thus reducing the overall complexity of the network. In addition to L2 regularization, the dropout regularization is also applied. FN-Net uses a dropout with probability 0.3, where 30% of the neuron will be dropped at every update.

FN-Net also optimizes the gradient descent process by leveraging the Adam optimizer. Adam optimizer adaptively tunes the learning rate for each weight in the network by considering the momentum. The advantage of the momentum is that it uses the moving average of the gradients hence avoid stuck in the local minima. The Adam optimizer accelerates and smooths the process of gradient descent.

As part of the optimization algorithm, a loss function is required to estimate the loss in every training epoch. In the fake news detection task, the binary cross entropy is chosen as the loss function. The binary cross entropy is calculated as below:

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

where y is the label (1 for fake news and 0 for real news) and $p(y)$ is the predicted probability of the text being fake news for all N samples.

IV. Dataset

The experiments in this work involve four publicly available fake news detection dataset.

a) Fake or real news dataset

The dataset was compiled by George McIntire. The sources of the news articles include Kaggle challenge, New York Times, WSJ, Bloomberg, NPR and The Guardian that were published in 2015 and 2016. The dataset contains 6335 articles with 3171 real news articles and 3164 fake news articles.

b) Kaggle – Fake and real news dataset

The dataset [1, 16] contains 21417 true news articles in the True.csv file and 23481 fake news articles in the Fake.csv file. The true news was collected from Reuter.com whereas fake news articles were from the unreliable websites that were flagged by Politifact and Wikipedia. PolitiFact is a fact-checking organization in the USA. The true news is categorized into two topics whereas the fake news is classified into six topics.

c) Kaggle – Getting real about fake news dataset

The dataset [17] contains 20015 news articles in which 11941 are fake news and 8074 are real news. The fake news articles were compiled from over 240 unreliable websites. On the other hand, the real news articles were crawled from the reliable news websites like New York News, Washington Post, etc.

d) Combined news dataset

The dataset was aggregated from multiple Kaggle fake news dataset by Amol Mavuduru (https://github.com/AmolMavuduru/FakeNewsClassification). The dataset contains 74012 with 36969 real news articles and 37043 fake news articles.

V. Results and Analysis

For a fair comparison, some existing fake news detection methods are also included in the experiments. The methods include SVM, Random Forest, logistic regression, decision tree, passive aggressive, Naïve Bayes and CNN model. All datasets are partitioned into 80% training set, 10% validation set and 10% testing set.

The performance of the models is measured in accuracy, precision, recall and F1-score, computed as below:

$$Accuracy = \frac{\mathrm{TP}+\mathrm{TN}}{\mathrm{TP}+\mathrm{FP}+\mathrm{TN}+\mathrm{FN}} \tag{2}$$

$$Precision = \frac{\mathrm{TP}}{\mathrm{TP}+\mathrm{FP}} \tag{3}$$

$$Recall = \frac{TP}{TP+FN} \tag{4}$$

$$F1-score = \frac{2 \times Precision \times Recall}{Precision + Recall} \tag{5}$$

where TP, TN, FP, FN are true positive, true negative, false positive and false negative, respectively.

Table I, Table II, Table III and Table IV present the results of the fake news detection methods on the fake news datasets. The experimental results show that CNN model and FN-Net outperform the other methods in all datasets. The CNN model, however, has some limitations. The FN-Net achieves an accuracy of 92.25%, precision of 92%, recall of 92% and F1-score of 92% in Fake or Real News Dataset. On the Kaggle – Fake and Real News Dataset, the FN-Net method yields an accuracy of 99.39%, precision, recall and F1-score of 99%. The FN-Net method records an accuracy of 95.07%, precision of 95%, recall and F1-score of 96% on the Kaggle – Getting Real about Fake News Dataset. On the Combined News Dataset, the FN-Net method obtains slightly lower accuracy and prevision than the CNN method with an accuracy of 95.07% and precision of 95%. The FN-Net however records the highest recall and F1-score of 96% on the dataset.

Fig. 5 compares the loss at every epoch of the CNN model and FN-Net. The loss plots demonstrate that the CNN model suffers from the overfitting problems and instable gradient descent process. In contrary, the FN-Net shows a steadier loss decrement in every epoch of the training and validation. The observation implies that the FN-Net has better generalization ability and training stability. Incorporating L2 regularization and dropout regularization makes the FN-Net less susceptible to the overfitting problem. Not only that, Adam optimizer has also accelerated and smoothed the training process of FN-Net.

TABLE I. The Results of the Fake News Detection Methods on Fake or Real News Dataset

Model	Accuracy	Precision	Recall	F1-score
SVM	88.13%	88%	88%	88%
Random Forest	89.04%	89%	89%	89%
Logistic regression	88.05%	88%	88%	88%
Decision tree	78.33%	78%	78%	78%
Passive Aggressive	82.94%	84%	83%	83%
Naive Bayes [11]	83.03%	84%	83%	83%
CNN [14]	91.31%	91	91	91
FN-Net	**92.25**	**92**	**92**	**92**

TABLE II. THE RESULTS OF THE FAKE NEWS DETECTION METHODS ON KAGGLE - FAKE AND REAL NEWS DATASET

Model	Accuracy	Precision	Recall	F1-score
SVM	91.89	92	92	92
Random Forest	94.65%	95	95	95
Logistic regression	91.85	92	92	92
Decision tree	87.19	87	87	87
Passive Aggressive	89.11	90	89	89
Naive Bayes [11]	88.58%	89	89	89
CNN [14]	97.9	98	98	98
FN-Net	**99.39**	**99**	**99**	**99**

TABLE III. THE RESULTS OF THE FAKE NEWS DETECTION METHODS ON KAGGLE – GETTING REAL ABOUT FAKE NEWS DATASET

Model	Accuracy	Precision	Recall	F1-score
SVM	88.41	88	87	88
Random Forest	89.39	89	89	89
Logistic regression	88.41	88	87	88
Decision tree	82.87	82	82	82
Passive Aggressive	86.58	88	86	86
Naive Bayes [11]	81.02	82	78	79
CNN [14]	94.5	**95**	94	94
FN-Net	**95.07**	**95**	**96**	**96**

TABLE IV. THE RESULTS OF THE FAKE NEWS DETECTION METHODS ON COMBINED NEWS DATASET

Model	Accuracy	Precision	Recall	F1-score
SVM	86.79	87	87	87
Random Forest	89.73	90	90	90
Logistic regression	86.52	86	87	87
Decision tree	82.21	82	82	82
Passive Aggressive	78.73	82	79	78
Naive Bayes [11]	81.70	82	82	82
CNN [14]	**96.37**	**96**	**96**	**96**
FN-Net	95.07	95	**96**	**96**

VI. CONCLUSION

Information technology has greatly reshaped our daily life. On one hand, these advancements in information technology have made communication easier than ever. On the other hand, the misuse of information technology has also made the fake news diffuse deeper and more rapidly. In view of this, an enhanced CNN model, known as the FN-Net, is devised for fake news detection. The FN-Net improves the original CNN model by integrating more convolution and max pooling layers. Besides that, L2 regularization and dropout are integrated to address the overfitting problem. The FN-Net is further optimized the Adam optimizer for smoother and faster training process. The experimental results on four fake news datasets corroborate the FN-Net performs better than the original CNN model.

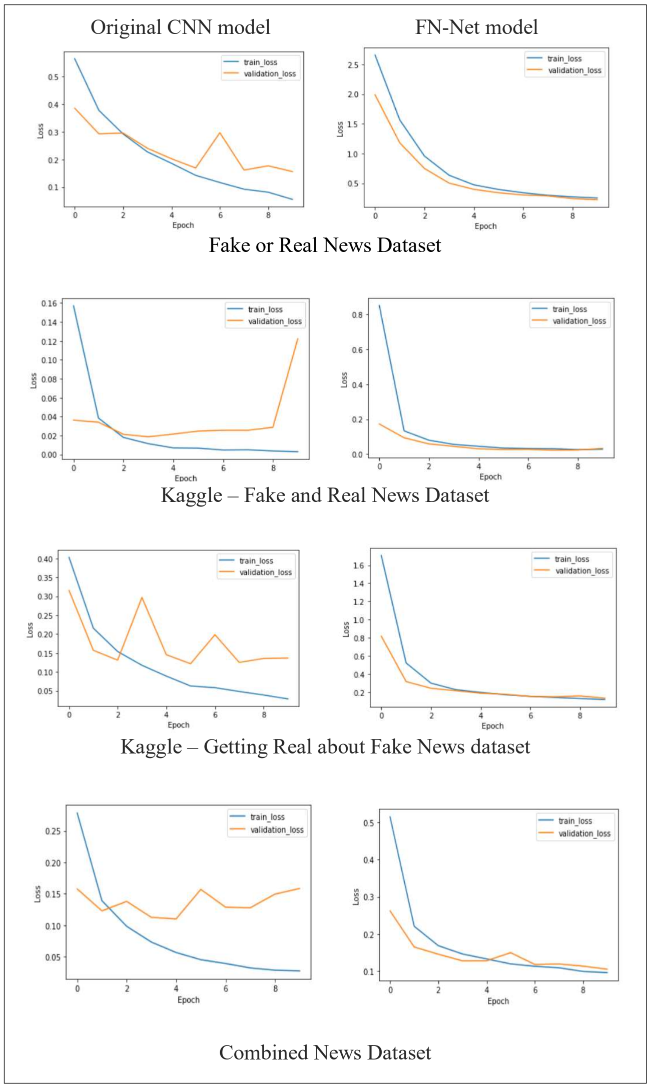

Fig. 5. The loss at every epoch of the original CNN and FN-Net.

REFERENCES

[1] Ahmed H, Traore I, Saad S. (2017) "Detection of Online Fake News Using N-Gram Analysis and Machine Learning Techniques. In: Traore I., Woungang I., Awad A. (eds) Intelligent, Secure, and Dependable Systems in Distributed and Cloud Environments. ISDDC 2017. Lecture Notes in Computer Science, vol 10618. Springer, Cham (pp. 127-138).

[2] Lee, C. P., Lim, K. M., & Woon, W. L. (2010, July). Statistical and entropy based multi purpose human motion analysis. In *2010 2nd International Conference on Signal Processing Systems* (Vol. 1, pp. V1-734). IEEE.

[3] Mogan, J. N., Lee, C. P., Lim, K. M., & Tan, A. W. (2017, November). Gait recognition using binarized statistical image features and histograms of oriented gradients. In *2017 International Conference on Robotics, Automation and Sciences (ICORAS)* (pp. 1-6). IEEE.

[4] Yu, T. J., Lee, C. P., Lim, K. M., & Razak, S. F. A. (2019). AI-Based Targeted Advertising System. *Indonesian Journal of Electrical Engineering and Computer Science*, *13*(2), 787-793.

[5] Lee, C. P., Woon, W. L., & Lim, K. M. (2010). Statistical and entropy based human motion analysis. *KSII Transactions on Internet and Information Systems (TIIS)*, *4*(6), 1194-1208.

[6] Mogan, J. N., Lee, C. P., & Tan, A. W. (2017, May). Gait recognition using temporal gradient patterns. In *2017 5th International Conference on Information and Communication Technology (ICoIC7)* (pp. 1-4). IEEE.

[7] Tan, Y. S., Lim, K. M., Tee, C., Lee, C. P., & Low, C. Y. (2020). Convolutional neural network with spatial pyramid pooling for hand gesture recognition. *Neural Computing and Applications*, 1-13.

[8] Gilda, S. (2017, December). Notice of Violation of IEEE Publication Principles: Evaluating machine learning algorithms for fake news detection. In *2017 IEEE 15th student conference on research and development (SCOReD)* (pp. 110-115). IEEE.

[9] Girgis, S., Amer, E., & Gadallah, M. (2018, December). Deep learning algorithms for detecting fake news in online text. In *2018 13th International Conference on Computer Engineering and Systems (ICCES)* (pp. 93-97). IEEE.

[10] Agarwalla, K., Nandan, S., Nair, V. A., & Hema, D. D. (2019). Fake news detection using machine learning and Natural Language Processing. *International Journal of Recent Technology and Engineering (IJRTE)*, *7*(6).

[11] Granik, M., & Mesyura, V. (2017, May). Fake news detection using naive Bayes classifier. In *2017 IEEE First Ukraine Conference on Electrical and Computer Engineering (UKRCON)* (pp. 900-903). IEEE.

[12] Jwa, H., Oh, D., Park, K., Kang, J. M., & Lim, H. (2019). exBAKE: automatic fake news detection model based on bidirectional encoder representations from transformers (bert). *Applied Sciences*, *9*(19), 4062.

[13] Ozbay, F. A., & Alatas, B. (2020). Fake news detection within online social media using supervised artificial intelligence algorithms. *Physica A: Statistical Mechanics and its Applications*, *540*, 123174.

[14] Thota, A., Tilak, P., Ahluwalia, S., & Lohia, N. (2018). Fake news detection: A deep learning approach. *SMU Data Science Review*, *1*(3), 10.

[15] Pennington, J., Socher, R., & Manning, C. D. (2014, October). Glove: Global vectors for word representation. In *Proceedings of the 2014 conference on empirical methods in natural language processing (EMNLP)* (pp. 1532-1543).

[16] Ahmed, H., Traore, I., & Saad, S. (2018). Detecting opinion spams and fake news using text classification. *Security and Privacy*, *1*(1), e9.

[17] Yang, Y., Zheng, L., Zhang, J., Cui, Q., Li, Z., & Yu, P. S. (2018). *TI-CNN: Convolutional Neural Networks for Fake News Detection*. http://arxiv.org/abs/1806.00749