Sherlock: An Ensemble based Deep Learning Framework for Fake News Detection

Sameer Kulkarni
Department of Information Technology
NBN Sinhgad School of Engineering, Pune, India

R. M. Samant
Department of Information Technology
NBN Sinhgad School of Engineering, Pune, India

Atharva Bhusari
Department of Information Technology
NBN Sinhgad School of Engineering, Pune, India

ABSTRACT

Fake news is impacting societal harmony and peace. Considering the magnitude of this harmful impact, there is a need to find a solution to curb the online spread of fake news. Detection of fake news is being tackled with various approaches like manual checks, statistical based classification algorithms and deep learning techniques in recent times. This task however, becomes tricky due to the non-binary (entirely true of false) nature of news reporting. Results reported in existing research work require deeper investigation such as classification on a scale of entirely true to entirely false rather than binary classification of news articles. In this paper, a novel ensemble-based framework – Sherlock, to detect fake news articles using natural language processing (NLP) and deep learning techniques is proposed. Due to unsatisfactory results of using a single approach, this framework consists of three distinct tasks of classification based on the article's semantic structure, source credibility and sentiment of the news. The technique of using pre-trained word vectors as word embeddings for semantic analysis has shown performance boost by 2-4%. Additionally, a scale for measuring fakeness of news is proposed. Sherlock classifies a given news article into one of the four degrees of fakeness- "true", "mostly-fake", "entirely-fake" or "uncertain". A comparison of the performance of text classification task using various statistical based machine learning algorithms and deep neural networks are also reported based on two publicly available benchmark datasets. The best test accuracies of 94% for binary classification and 65.5% for multiclass classification were obtained for a GRU (Gated Recurrent Unit) based deep neural network model which has been incorporated in the proposed framework. Sherlock uses a browser plugin to accept news for detection via web-scraping technique and consequently, the training dataset is updated in order to establish context for current affairs. An indigenous dataset which is frequently updated with Indian news context is introduced for the first time to the best of our knowledge. The overall product experience using Sherlock largely intervenes the impulsive behavior of forwarding news, and thereby provides the solution to curb rampant spread of fake news.

General Terms

Fake News Detection, Machine Learning, Natural Language Processing, Data Mining.

Keywords

Fake News, Natural Language Processing, Deep Learning, Semantic Analysis, Sentiment Analysis, Pre-trained Vectors, Gated Recurrent Unit.

1. INTRODUCTION

Fake news analysis and detection has become an emerging field of research due its effect on the socio-economic and political factors of the world. Impact of fake news ranges from causing political unrest and hampering public administration to spreading social deceptiveness and religious hatred. Existing solutions are less effective in giving precise statistical rating for news articles. One of the reasons is the challenges in natural language processing since natural language can be ambiguous, where a word can have different meaning and interpretations based on the context it is being used in. The proposed framework *SHERLOCK*, comes up with a human-like approach of detecting whether a news article is fake i.e. by considering the semantic style of the news report, checking if the source is reliable and whether the news expresses unrealistically extreme positive or negative sentiments. The basic idea behind the approach is to identify deceptiveness and minimize ambiguity in an intentionally crafted piece of misinformation. Gated recurrent unit is used in the deep neural network architecture which is a variant of Recurrent neural network introduced by Kyunghyun cho et al [1]. The semantic model of the framework has been trained on a dataset of over 45,000 English news articles for multiclass classification which gives an accuracy of over 65%. The source reliability model performs better given its binary nature, and gives an accuracy of 94%. Subsequently the models are retrained with updated data (Headline, news article, source and date), as the semantics and vocabulary changes with time. This also helps with updating a bag-of-words (BOW) model of sources/authors that are linked with producing fake news. This is done by regularly scraping latest articles from news websites.

The paper is organized as follows. Previous related work done on fake news detection is discussed in Section 2. The datasets used are briefly described in Section 3, followed by a detailed methodology of the proposed framework and neural network architecture used for the models in Section 4. Results of experimented algorithms/neural networks along with performance of the models used in the framework are discussed in Section 5 and finally Section 1 concludes the paper.

2. RELATED WORK

Fact checking websites such as BOOM, AltNews.in, PolitiFact.com and Factly.com tackle fake news by employing experts and journalists which is an inefficient approach as manual individual checking of news articles will take up non-deterministic amount of time and may still yield inaccurate and biased results. There has been extensive research carried out with regards to detecting fake news from social media using data such as tweets and posts or images ([2], [3], [4], [5]). However, fake news articles spreading through news websites which can be considered to be a more believable source is not much explored. Also, lack of manually labelled fake news dataset is a challenge mentioned in the paper [6] and it introduces a fine-grained dataset containing short statements and their attributes which got an accuracy of 27.4% using hybrid CNN. It also proved that using metadata associated with news can give improved performance for fake news detection. Literatures [6] and [7] report results for multiclass datasets while all other mentioned work focuses on binary classification which might not be an efficient form of detection considering that news articles can partially contain true information along with false, misleading information and hence classifying them as completely true or false can be inconclusive. [8] proposed hand-crafted features to be used in machine learning classifiers. Another literature introduces a form of query matching system using online fact-checkers and trusted news website [9]. This however, may create a dependency on the efficiency of the third-party fact-checkers and availability of news content on those websites.

3. DATASET

Table 1. Summary statistics of datasets used for experimentation and model training

Dataset	Total no. of articles	Attributes	Classes
DoF	45000	Title, Text, Author, Label	4
FNC	20800	Id, Title, Author, Text, Label	2
SR	604	Text, Author, Label	2
SST	10300	Sentence, Label	2

Train, validation and test samples are split as 70%, 10% and 20% respectively except for SST, where standard train test split is given.

- **DOF:** Degree of Fakeness dataset that is an amalgamation of datasets “Getting Real about fake news”, “Liar, Liar pants on fire”, “Fake news challenge dataset” ([6], [10], [11]) and 302 manually scraped true news articles focusing on Indian news. The datasets commonly contain title, text author and label of news articles and statements. The datasets have been relabelled into 4 labels as – 0: True, 0.5: uncertain, 0.75: mostly-fake, 1.0: fake.
- **FNC:** Fake News Challenge dataset [10] contains news articles, attributes and their binary labels as reliable and unreliable. It is used for comparison of accuracies of different algorithms/neural networks with the model architecture used in the framework.
- **SR:** Source reliability dataset that is a subset of the DoF dataset with only a selected number of international authors and labelled as trusted for true and not-trusted for mostly-fake and fake news articles. The labels are as – 0: unreliable, 1: reliable.
- **SST:** Stanford sentiment treebank which includes movie reviews and the fine-grained labels converted to binary as – 0: normal (positive/negative) and 1: extreme (extremely positive/negative) and neutral values discarded [12].

4. PROPOSED FRAMEWORK

The task is to detect the degree of fakeness of news articles by considering the output of three ensembled models discussed in detail in further subsections. To do this, a rule based-decision system is used as shown in Table 2. The basic idea behind designing the rules is a higher priority to the semantic analysis model as it extracts better features from the whole news article and gives a fine-grained output in terms of degrees of fakeness as described in Section 3. Furthermore, the sentiment and source models give leverage to the semantic model by giving information about the degree of sentiment (normal or extreme) of the article and whether the source is reliable, based on a BOW model.

Table 2. Rules for the decision system based on output of three models as semantic-sentiment-score

Condition	Decision
(T-N-R) | (T-N-U) | (T-E-T) | (H-N-T)	TRUE
(H-N-U) | (H-E-U) | (M-N-U) | (M-E-R)	MOSTLY-FAKE
(M-E-U) | (F-N-U) | (F-E-R) | (F-E-U)	ENTIRELY-FAKE
(T-E-U) | (H-E-R) | (M-N-R) | (F-N-R)	UNCERTAIN

Table 2 contains the combination of possible results and the corresponding decision based on those results where, [T: true; H: half-true; M: mostly-fake; F: fake], [N: normal; E: extreme], [R: reliable, U: unreliable]. Fig 1 represents the process flow for a new article to be detected for fakeness. Data regarding news content and its attributes like date of publishing and author is scraped from online news websites using web-scraping technique and preprocessed before feeding respective data to the models and giving result based on the rule-based decision as shown in Table 2. The process ensures total automation from data gathering to displaying output on a user-interface.

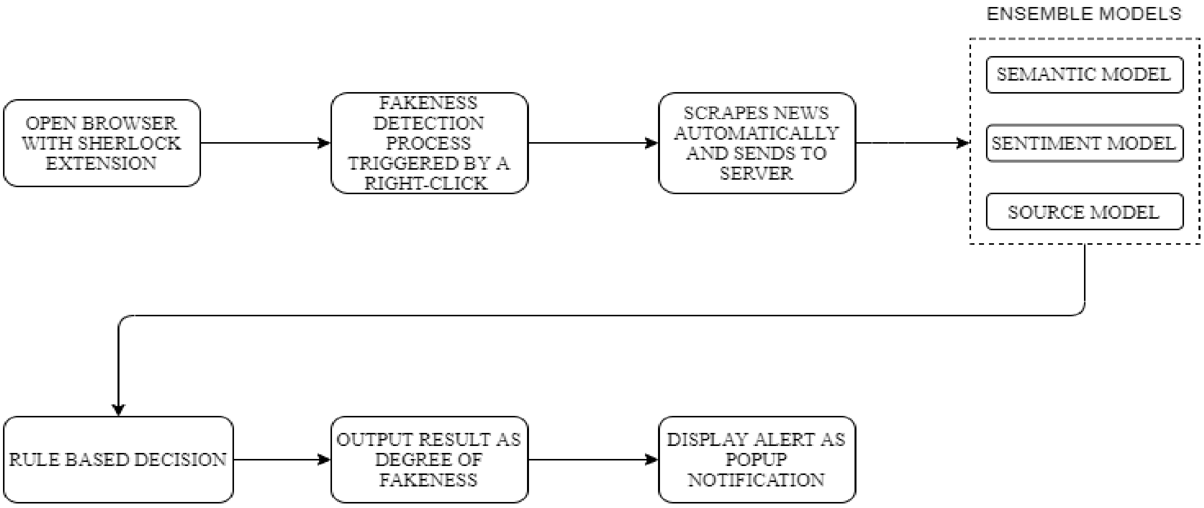

ENSEMBLE MODELS
OPEN BROWSER WITH SHERLOCK EXTENSION
FAKENESS DETECTION PROCESS TRIGGERED BY A RIGHT-CLICK
SCRAPES NEWS AUTOMATICALLY AND SENDS TO SERVER
SEMANTIC MODEL
SENTIMENT MODEL
SOURCE MODEL
RULE BASED DECISION
OUTPUT RESULT AS DEGREE OF FAKENESS
DISPLAY ALERT AS POPUP NOTIFICATION

Fig 1: System flow diagram

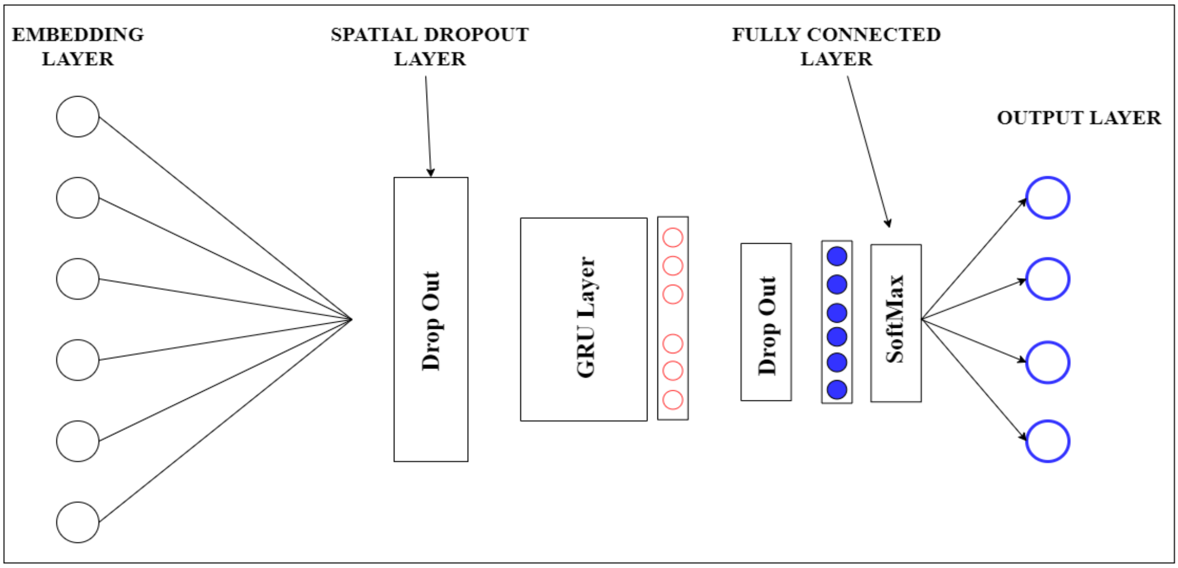


Fig 2: Model Architecture

4.1 Model Architecture

Recurrent Neural Networks (RNN) are specially used for text-based classification problems and in theory, can backpropagate through time to calculate the gradients. Gated recurrent unit is a variation of recurrent neural networks that solves the vanishing or exploding gradient problem typically found in vanilla RNNs. [13] shows that, as the gap grows between an input x_t and another output h_{t+n}, RNN does not tend to learn or connect to the information. This led to the development of Long short-term memory [14] and GRU which are similar to RNNs but with gates to improve long term dependencies. GRU is similar to LSTM network in controlling information flow mechanism, without using the memory unit. Hence clearly, without memory unit, GRU has better computational efficiency. The GRU model has two gates as update gate and the reset gate which control the flow of amount of information from one unit to the next. The mechanism of an update gate z_t at a time step t for an input x_t can be represented mathematically as:

$$z_t = \sigma(W^{(z)}x_t + U^{(z)}h_{t-1}) \qquad (1)$$

Where, h_{t-1} is output of a previous unit and $W^{(z)}$, $U^{(z)}$ are the weights for update gate z_t associated with the current input x_t and previous unit output respectively. The sum of these results is regulated using an activation function σ (Gate activation), typically Tanh or ReLU. Thus, it controls the amount of information to be passed ahead from the previous time steps. Similarly, a reset gate r_t can be represented as:

$$r_t = \sigma(W^{(r)}x_t + U^{(r)}h_{t-1}) \qquad (2)$$

The difference from update gate lies in the value of wights. Specifically, the reset gate controls the amount of information to forget from the previous time steps. The content of current unit stored in h'_t is calculated as sum of current input with its weight and Hadamard product of reset gate and previous information with its weight and finally, this result is multiplied by an activation function (Unit activation) to regulate the output as shown in (3) The collective final information h_t to be passed on to the next unit calculates the sum of Hadamard product of update gate with previous input and (1-z_t) with content of current unit as shown in (4).

$$h'_t = \sigma(Wx_t + r_t \ \Theta \ Uh_{t-1}) \qquad (3)$$

$$h_t = z_t \ \Theta \ h_{t-1} + (1 - z_t) \ \Theta \ h'_t \qquad (4)$$

A simplified model architecture as shown in Fig 2 is common for all the three models used in the framework but with different characteristics such as hyperparameter values and training rate suitable for the specific task of each model as described in Table 1with detailed discussion in Section 5.2.

4.2 Semantic Model

The semantic model uses news articles to extract semantic features using static pre-trained word embeddings. A popular method to improve performance of network models is to initialize word vectors with those obtained from an unsupervised neural language model [15]. Word2vec is a predictive modeling algorithm by Tomas Mikolov used to predict the context words based on target words or vice-versa using either skip-gram or a continuous bag-of-words model (CBOW) [16]. However, skip-gram has proved to perform better with long length corpuses by producing 2 vectors for each word; one vector considering the word as center word and another considering it as a context word to predict another center word. Hence, skip-gram is better for the scope of this framework as data is in the form of long-length news articles.

The skip-gram model will define a probability distribution to predict context words $\mathbf{w_{-t}}$ (words around target word w_t). It will adjust the vectors such that to maximize the probability of predictions such as:

$$P(O|C) = \frac{\exp(U_0 + V_c)}{\sum_{w=1}^{v} \exp(U_w + V_c)} \tag{5}$$

Where, O is the output word index; C is the center word index; V_c and V_o are center and outside vectors associated with indices c and o. Sherlock uses the transfer-learning to create word-embeddings and transfer these embeddings to the framework model architecture to extract semantic-based features.

The word2vec vectors are trained on the same dataset used to train the model and this dataset is continuously updated to include latest vocabulary used in news articles. This approach performs exceptionally better when compared to training the vectors at embedding layer in the network architecture.

Table 3. Network Model characteristics

Model	Units	Gate Activation	Activation	Loss	Dropout	Optimizer	Batch Size	Epochs
Semantic	200	Tanh	ReLU	Cross Entropy	0.4	Adam	128	5
Sentiment	100	Tanh	Sigmoid	Cross Entropy	0.2	Adam	128	5
Source	50	Tanh	Sigmoid	Cross Entropy	0.2	Adagrad	256	5

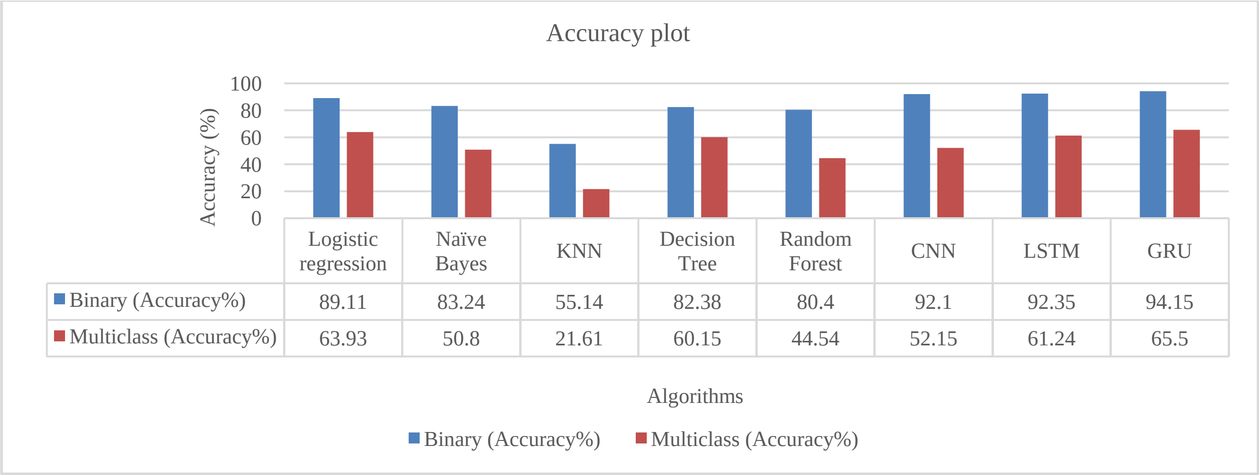
4.3 Source and Semantic Model

While analyzing fakeness in a news articles, information regarding the source reliability is identified using the frequently updated BOW model. Each time a common source associated with fake news, their frequency count in the unreliable class increases. The model is built on the idea that news articles from reliable sources increases the probability of the article being true and in contrast, unreliable sources increase the probability of the news being fake. For example, the results can be observed to be highly manipulated by the output of source model when a news article is classified as "uncertain" (0.5) by the semantic model. In cases where the author is anonymous, the model would not be part of the decision-making process as clearly, it's prediction would not be correct regardless of the result. Similar action takes place for cases where author is not a part of the current BOW. However, in this case, the article along with the author and its result predicted by the framework will be saved into the dataset for inclusion in future training process. Thus, the sentiment model needs to be trained frequently with updated dataset as reliability of sources depend upon the number of true/fake news articles published by them. The sentiment model detects whether the positive and negative sentiments in the article are normal or extreme. It should be noted the labels of dataset [SST] are divided such that the values from 0 to 0.2 and from 0.81 to 1.0 are considered as "extreme" and the values from 0.21 to 0.4 and from 0.61 to 0.8 as "normal" unlike for example in [15], [17] and [18] where values are divided as "positive" and "negative" on either side of the neutral values (from 0.41 to 0.6). The basic hypothesis is that the probability of a news being fake increases when news text is carrying extremely negative or positive sentiment coupled with an unreliable source/author. For example, by adding extremely negative sentiments in a religion-based news, an author can create a sense of hatred among readers. Significant features like adjectives, adverbs, etc. are selected as features for sentiment classifier to be classified into normal or extreme sentiment.

5. RESULTS

5.1 Comparison Report

Initial task in the research was to compare statistical machine learning algorithms and deep neural networks. Fig 1 represents comparison in terms of accuracy metrics when trained and tested on binary and multiclass datasets. All the algorithms/models were trained using the same random split for both datasets to maintain fairness in comparison. It was observed that while the machine learning algorithms performed fairly good in terms of binary classification, they overfit the training samples and gave poor test results for multiclass classification. This was not the case with neural networks where only a minor difference was observed in training and testing results. The GRU-based neural network outperformed all the other models. The performance was further also compared in terms of precision, recall and F1 score. Results in similar proportions were observed with a few exceptions, more precisely in multiclass classification but the GRU based network was fairly ahead in those metrics as well.



	Logistic regression	Naïve Bayes	KNN	Decision Tree	Random Forest	CNN	LSTM	GRU
🟦 Binary (Accuracy%)	89.11	83.24	55.14	82.38	80.4	92.1	92.35	94.15
🟥 Multiclass (Accuracy%)	63.93	50.8	21.61	60.15	44.54	52.15	61.24	65.5

Fig 1: Results (Test Accuracy) for Binary and Multiclass classification

5.2 Performance of Framework Models

Table 3 gives information about the model characteristics. There are different hyperparameters and batch settings for each model. The best settings have been selected via a grid search on respective models. However, it was observed that change in dropout layer values for regularization proved to be ineffective for the source model. It is set at a standard value of 20% and can be adjusted in the future when an increased dataset is available through frequent web scraping of news content as mentioned above. Semantic model uses categorical cross-entropy as loss function for multiclass classification as opposed to binary cross-entropy for the other 2 models as they are assigned for binary classification. Learning rate has not been explicitly set for the models as Adam optimizer is used which is an adaptive algorithm and would compute individual learning rates for various parameters [19]. The results for all the ensemble models are reported in Table 3. The source model performed fairly good despite being trained on a small dataset. It is to be seen whether the model performs consistently with a continuously increasing training data size. Semantic model is the same as used for comparison in Fig 3 and has been reported in terms of the other parameters as well in Table 4. The sentiment model gives a lesser accuracy than expected for a binary classification but it is understood that the reason is due to there being less distinction between extremely positive/negative and normal positive/negative sentiments as compared to the higher distinction between positive (extreme and normal) and negative (extreme and normal) sent

Table 3. Model metrics of Proposed Framework

Models	Class	Precision	Recall	F1 Score	Accuracy
Semantic Model	TRUE	0.88	0.68	0.77	0.655
	Mostly-fake	0.59	0.5	0.54	
	Fake	0.63	0.66	0.65	
	Uncertain	0.51	0.84	0.63	
Sentiment model	Normal	0.62	0.6	0.61	0.628
	Extreme	0.58	0.68	0.66	
Source Model	Reliable	0.86	0.98	0.92	0.917
	unreliable	0.9	0.82	0.9	

6. CONCLUSION

The proposed framework performs fairly well on the parameters of response time and classification accuracy. The reported accuracy of 94% for binary and 65.5% for multiclass classification is satisfactorily higher than the current reported ones. The features of news like semantic, sentiment and source play a major part in boosting the overall performance of the framework. The GRU-based deep learning model performs much better than the earlier reported techniques. Ensemble based framework approach widens the scope for detecting fakeness from different dimensions of news content. Sherlock is implemented as a tool for detection and intervention of fake news articles on the internet in order to curb its spreading and consequently create a culture of fact checking and control on impulsive forwarding of such articles through social media platforms. Future scope is to incorporate a model in the framework to predict reliability based on images available with the news.

7. REFERENCES

[1] K. Cho, B. v. Merrienboer, C. Gulcehre, D. Bahdanau, F. Bougares, H. Schwenk and Y. Bengio, "Learning Phrase Representations using RNN Encoder–Decoder," arXiv:1406.1078, 2014.

[2] S. Helmstetter and H. Paulheim, "Weakly Supervised Learning for Fake News Detection on Twitter," IEEE/ACM International Conference on Advances in Social Networks Analysis and Mining (ASONAM) , 2018, pp. 274-277.

[3] S. Krishnan and M. Chen, "Identifying Tweets with Fake

News," IEEE International Conference on Information Reuse and Integration for Data Science, 2018, pp. 460-464.

[4] A. Sen, K. Rudra and S. Ghosh, "Extracting Situational Awareness from Microblogs during Disaster Events," Social Networking Workshop, COMSNETS 2015 , 2015.

[5] T. Traylor , J. Straub, Gurmeet and N. Snell, "Classifying Fake News Articles Using Natural Language Processing to Identify In-Article Attribution as a Supervised Learning Estimator," IEEE 13th International Conference on Semantic Computing (ICSC), 2019, pp. 445-449.

[6] W. Y. Wang, "Liar, liar pants on fire: A new benchmark dataset for fake news detection," arXiv:1705.00648v1 [cs.CL], 2017, p. 422–426.

[7] S. Saad, W. Nicholas, S. Mei-Ling and F. Daniel, "High Dimensional Latent Space Variational Auto Encoders for Fake News Detection," IEEE Conference on Multimedia Information Processing and Retrieval (MIPR), 2019, pp. 437-442.

[8] J. Reis, A. Correia, F. Murai, A. Veloso and F. Benevenuto, "Supervised Learning for Fake News Detection," IEEE Computer Society, 2019, pp. 76-81.

[9] S. Sudarshan, S. Seth, K. Chebrolu, S. Chakrabarti, M. Agarwal, A. Pale and A. Bagade, "The Kauwa-Kaate Fake News Detection System: Demo," ACM India Joint International Conference on Data Science and Management of Data (CoDS-COMAD) , 2020.

[10] "Fake News Challenge," [Online]. Available: http://www.fakenewschallenge.org/.

[11] M. Risdal, "Kaggle.com," [Online]. Available: https://www.kaggle.com/mrisdal/fake-news.

[12] R. Socher, A. Perelygin, J. Y. Wu, J. Chuang, C. D. Manning, A. Y. Ng and C. Potts, "Recursive Deep Models for Semantic Compositionality Over a Sentiment Treebank," In Proceedings of EMNLP, 2013.

[13] Y. Bengio, P. Simrad and P. Franconi, "Learning Long-Term Dependencies with Gradient Descent is Difficult," IEEE Transactions on Neural Networks , Vol 5, No. 2, 1994, pp. 157-166.

[14] S. Hochreiter and J. ¨. Schmidhuber, "Long Short-Term Memory," in *Neural Computation Vol. 9,8*, https://doi.org/10.1162/neco.1997.9.8.1735, 1997, pp. 1735-1780.

[15] Y. Kim, "Convolutional Neural Networks for Sentence Classification," Conference on Empirical Methods in Natural Language Processing (EMNLP), 2014, pp. 1746-1751.

[16] T. Mikolov, I. Sutskever, K. Chen, G. Corrado and J. Dean, "Distributed Represnetations of Words and Phrases and their Compositionality," 2013, pp. 1-9.

[17] R. Socher, J. Pennington, E. H. Huang, A. Y. Ng and C. D. Manning, "Semi-Supervised Recursive Autoencoders," EMNLP '11: Proceedings of the Conference on Empirical Methods in Natural Language Processing, 2011, pp. 151-161.

[18] N. Kalchbrenner, E. Grefenstette and P. Blunsom, "A Convolutional Neural Network for Modelling Sentences," arXiv:1404.2188v1 [cs.CL] , 2014.

[19] D. P. Kingma and J. L. Ba, "ADAM: A METHOD FOR STOCHASTIC OPTIMIZATION," 3rd International Conference for Learning Representations, San Diego arXiv:1412.6980, 2015.