A Deep Ensemble Framework for Fake News Detection and Classification

***Fake News Detection using Deep Learning**

Arjun Roy, Kingshuk Basak, Asif Ekbal, Pushpak Bhattacharyya

Department of Computer Science and Engineering
Indian Institute of Technology Patna, India
(arjun.mtmc17, kingshuk.mtcs16, asif, pb) @iitp.ac.in

Abstract. Fake news, rumor, incorrect information, and misinformation detection are nowadays crucial issues as these might have serious consequences for our social fabrics. Such information is increasing rapidly due to the availability of enormous web information sources including social media feeds, news blogs, online newspapers etc. In this paper, we develop various deep learning models for detecting fake news and classifying them into the pre-defined fine-grained categories. At first, we develop individual models based on Convolutional Neural Network (CNN), and Bi-directional Long Short Term Memory (Bi-LSTM) networks. The representations obtained from these two models are fed into a Multi-layer Perceptron Model (MLP) for the final classification. Our experiments on a benchmark dataset show promising results with an overall accuracy of 44.87%, which outperforms the current state of the arts.

1 Introduction

"We live in a time of fake news- things that are made up and manufactured."-by Neil Portnow.
Fake news, rumors, incorrect information, misinformation have grown tremendously due to the phenomenal growth in web information. During the last few years, there has been a year-on-year growth in information emerging from various social media networks, blogs, twitter, facebook etc. Detecting fake news, rumor in proper time is very important as otherwise, it might cause damage to social fabrics. This has gained a lot of interest worldwide due to its impact on recent politics and its negative effects. In fact, Fake News has been named as 2017's word of the year by Collins dictionary[1].

Many recent studies have claimed that US election 2016 was heavily impacted by the spread of Fake News. False news stories have become a part of everyday life, exacerbating weather crises, political violence, intolerance between people of different ethnics and culture, and even affecting matters of public health. All the governments around the world are trying to track and address these problems.

[1] http://www.thehindu.com/books/fake-news-named-word-of-the-year-2017/article19969519.ece

On 1^{st} Jan, 2018, bbc.com published that ”Germany is set to start enforcing a law that demands social media sites move quickly to remove hate speech, fake news, and illegal material.” Thus it is very evident that the development of automated techniques for detection of Fake News is very important and urgent.

1.1 Problem Definition and Motivation

Fake News can be defined as completely misleading or made up information that is being intentionally circulated claiming as true information. In this paper, we develop a deep learning based system for detecting fake news.

Deception detection is a well-studied problem in Natural Language Processing (NLP) and researchers have addressed this problem quite extensively. The problem of detecting fake news in our everyday life, although very much related to deception detection, but in practice is much more challenging and hard, as the news body often contains a very few and short statements. Even for a human reader, it is difficult to accurately distinguish true from false information by just looking at these short pieces of information. Developing suitable hand engineered features (for a classical supervised machine learning model) to identify fakeness of such statements is also a technically challenging task. In contrast to classical feature-based model, deep learning has the advantage in the sense that it does not require any handcrafting of rules and/or features, rather it identifies the best feature set on its own for a specific problem. For a given news statement, our proposed technique classifies the short statement into the following fine-grained classes: *true*, *mostly-true*, *half-true*, *barely-true*, *false* and *pants-fire*. Example of such statements belonging to each class is given in Table 1 and the meta-data related to each of the statements is given in Table 2.

1.2 Contributions

Most of the existing studies on fake news detection are based on classical supervised model. In recent times there has been an interest towards developing deep learning based fake news detection system, but these are mostly concerned with binary classification. In this paper, we attempt to develop an ensemble based architecture for fake news detection. The individual models are based on Convolutional Neural Network (CNN) and Bi-directional Long Short Term Memory (LSTM). The representations obtained from these two models are fed into a Multi-layer Perceptron (MLP) for multi-class classification.

1.3 Related Work

Fake new detection is an emerging topic in Natural Language Processing (NLP). The concept of detecting fake news is often linked with a variety of labels, such as misinformation[1], rumor[2], deception[3], hoax[4], spam[5], unreliable news[6], etc. In literature, it is also observed that social media[7] plays an essential role in

Table 1. Example statement of each class.

Ex	Statement (St)	Label
1	McCain opposed a requirement that the government buy American-made motorcycles. And he said all buy-American provisions were quote 'disgraceful.'	True
2	Almost 100,000 people left Puerto Rico last year.	Mostly-true
3	Rick Perry has never lost an election and remains the only person to have won the Texas governorship three times in landslide elections.	Half-true
4	Mitt Romney wants to get rid of Planned Parenthood.	barely-true
5	I dont know who (Jonathan Gruber) is.	FALSE
6	Transgender individuals in the U.S. have a 1-in-12 chance of being murdered.	pants-fire

the rapid spread of fake contents. This rapid spread is often greatly influenced by social bots[8]. It has been some time now since AI,ML, and NLP researchers have been trying to develop a robust automated system to detect Fake/ Deceptive/ Misleading/ Rumour news articles on various online daily access media platforms. There have been efforts to built automated machine learning algorithm based on the linguistic properties of the articles to categorize Fake News. Carlos Castillo et al.[9] in their work on social media (twitter) data showed that information from user profiles can be useful feature in determining veracity of news. These features were later also used by Aditi et al. to build a real-time system[10] to access credibility of tweets using SVM-rank. Researchers have also attempted to use Rule-Based and knowledge driven techniques to track the problem. Zhou et al. in their work [11] showed that deceptive senders have certain linguistic cues in their text. The cues are higher quantity, complexity, non-immediacy, expressiveness, informality, and affect; and less diversity, and specificity of language in their messages. Methods based on Information Retrieval from web were also proposed to verify authenticity of news articles. Etzioni et al. in their work[12] extracted claims from web to match with that of a given document to find inconsistencies. To deal with the problem further, researchers have also tried to seek deep learning strategies in their work. Bajaj[13] in his work applied various deep learning strategies on dataset composed of fake news articles available in Kaggle[2] and authentic news articles extracted from Signal Media News[3] dataset

[2] https://www.kaggle.com/mrisdal/fake-news

[3] http://research.signalmedia.co/newsir16/signal-dataset.html

Table 2. Meta-data related to each example. P, F, B, H, M is speaker's previous count of Pants-fire, False, Barely-true, Half-true, Mostly-true respectively.

Ex	St Type	Spk	Spk's Job	State	Party	P	F	B	H	M	Context
1	federal-budget	barack-obama	President	Illinois	democrat	70	71	160	163	9	a radio ad
2	bankruptcy, economy, population	jack-lew	Treasury secretary	Washington, D.C.	democrat	0	1	0	1	0	an interview with Bloomberg News
3	candidates-biography	ted-nugent	musician	Texas	republican	0	0	2	0	2	an oped column.
4	abortion, federal-budget, health-care	planned-parenthood -action-fund	Advocacy group	Washington, D.C.	none	1	0	0	0	0	a radio ad
5	health-care	nancy-pelosi	House Minority Leader	California	democrat	3	7	11	2	3	a news conference
6	corrections-and-updates, crime, criminal -justice, sexuality	garnet-coleman	president, ceo of Apartments for America, Inc.	Texas	democrat	1	0	1	0	1	a committee hearing

and observed that classifiers based on Gated Recurrent Unit (GRU), Long Short Term Memory (LSTM), Bi-directional Long Short Term Memory (Bi-LSTM) performed better than the classifiers based on CNN. Jing Ma et al. in their work[14], focused on developing a system to detect Rumor at EVENT level rather than at individual post level. The approach was to look at a set of relevant posts to a event at a given time interval to predict veracity of the event. They showed that use of recurrent networks are particularly useful in this task. Dataset from two different social media platform, Twitter, and Weibo were used. Chen et al. further built on the work of [2] for early detection Rumors at Event level, using the same dataset. They showed that the use of attention mechanism in recurrent network improves the performance in terms of precision, and recall measure, outperforming every other existing model for detecting rumor at an early stage. Natali Ruchansky et al.[15] used social media dataset (which is also used in [14] for Rumor Detection) and developed a hybrid deep learning model which showed promising performance on both Twitter data and Weibo data. They showed that both, capturing the temporal behavior of the articles as well as learning source characteristics about the behavior of the users, are essential for fake news detection. Further integrating these two elements improves the performance of the classifier.

Problems related to these topics have mostly been viewed concerning binary classification. Likewise, most of the published works also has viewed fake news detection as a binary classification problem (i.e., fake or true). But by observing very closely it can be seen that fake news articles can be classified into multiple classes depending on the fakeness of the news. For instance, there can be certain exaggerated or misleading information attached to a true statement or news. Thus, the entire news or statement can neither be accepted as completely true nor can be discarded as entirely false. This problem was addressed by William Y Yang in his paper [16] where he introduced **Liar** dataset comprising of a substantial volume of short political statements having six different class annotations determining the amount of fake content of each statement. In his work, he showed comparative studies of several statistical and deep learning based models for the classification task and found that the CNN model performed best. Long et al. [17] in their work used the **Liar** [17] dataset, and proposed a hybrid attention-based LSTM model for this task, which outperformed W.Yang's hybrid CNN model, establishing a new state-of-the-art.

In our current work we propose an ensemble architecture based on CNN [18] and Bi-LSTM [19], and this has been evaluated on **Liar**[16] dataset. Our proposed model tries to capture the pattern of information from the short statements and learn the characteristic behavior of the source speaker from the different attributes provided in the dataset, and finally integrate all the knowledge learned to produce fine-grained multi-class classification.

2 Methodology

We propose a deep multi-label classifier for classifying a statement into six fine-grained classes of fake news. Our approach is based on an ensemble model that makes use of Convolutional Neural Network (CNN) [18] and Bi-directional Long Short Term Memory (Bi-LSTM) [19]. The information presented in a statement is essentially sequential in nature. In order to capture such sequential information we use Bi-LSTM architecture. Bi-LSTM is known to capture information in both the directions: forward and backward. Identifying good features manually to separate true from fake even for binary classification, is itself, a technically complex task as human expert even finds it difficult to differentiate true from the fake news. Convolutional Neural Network (CNN) is known to capture the hidden features efficiently. We hypothesize that CNN will be able to detect hidden features of the given statement and the information related to the statements to eventually judge the authenticity of each statement. We make an intuition that both- capturing temporal sequence and identifying hidden features, will be necessary to solve the problem. As described in data section, each short statement is associated with 11 attributes that depict different information regarding the speaker and the statement. After our thorough study we identify the following relations among the various attributes which contribute towards labeling of the given statements.

1. Relation between **Statement** and **Statement type**
2. Relation between **Statement** and **Context**
3. Relation between **Speaker** and **Party**.
4. Relation between **Party** and **Speaker's job**.
5. Relation between **Statement type** and **Context**.
6. Relation between **Statement** and **State**.
7. Relation between **Statement** and **Party**.
8. Relation between **State** and **Party**.
9. Relation between **Context** and **Party**.
10. Relation between **Context** and **Speaker**.

To ensure that deep networks understand these relations we propose to feed each of these relations into separate network layers and eventually after extracting all the relations we group them together along with the five-column attributes containing information regarding speaker's total credit history count. In addition to that, we also feed in a special feature vector that is proposed by us and is to be formed using the count history information. This vector is a five-digit number signifying the five count history columns, with only one of the digit being set to '1' (depending on which column has the highest count) and the rest of the four digits are set to '0'.

2.1 Bi-LSTM

Bidirectional LSTMs are the networks with LSTM units that process word sequences in both the directions (i.e. from left to right as well as from right to left). In our model we consider the maximum input length of each statement to be 50 (average length of statements is 17 and the maximum length is 66, and only 15 instances of the training data of length greater than 50) with post padding by zeros. For attributes like statement type, speaker's job, context we consider the maximum length of the input sequence to be 5, 20, 25, respectively. Each input sequence is embedded into 300-dimensional vectors using pre-trained Google News vectors[20] (Google News Vectors 300dim is also used in [16] for embedding). Each of the embedded inputs are then fed into separate Bi-LSTM networks, each having 50 neural units at each direction. The output of each of these Bi-LSTM network is then passed into a dense network of 128 neurons with activation function as 'ReLU'.

2.2 CNN

Over the last few years many experimenters has shown that the convolution and pooling functions of CNN can be successfully used to find out hidden features of not only images but also texts. A convolution layer of $n \times m$ kernel size will be used (where m-size of word embedding) to look at n-grams of words at a time and then a MaxPooling layer will select the largest from the convoluted inputs.The attributes, namely speaker, party, state are embedded using pre-trained 300-dimensional Google News Vectors[20] and then the embedded inputs are fed into separate Conv layers.The different credit history counts the fake statements of a speaker and a feature proposed by us formed using the credit history counts are directly passed into separate Conv layers.

2.3 Combined CNN and Bi-LSTM Model

The representations obtained from CNN and Bi-LSTM are combined together to obtain better performance.

The individual dense networks following the Bi-LSTM networks carrying information about the statement, the speaker's job, context are reshaped and then passed into different Conv layers. Each convolution layer is followed by a Max-pooling layer, which is then flattened and passed into separate dense layers. Each of the dense layers of different networks carrying different attribute information are merged, two at a time-to capture the relations among the various attributes as mentioned at the beginning of 2. Finally, all the individual networks are merged together and are passed through a dense layer of six neurons with softmax as activation function as depicted in. The classifier is optimized using Adadelta as optimization technique with categorical cross-entropy as the loss function.

3 Data

We use the dataset, named **LIAR** (Wang 2017), for our experiments. The dataset is annotated with six fine-grained classes and comprises of about 12.8K annotated short statements along with various information about the speaker. The statements which were mostly reported during the time interval [2007 to 2016], are considered for labeling by the editors of **Politifact.com**. Each row of the data contains a short statement, a label of the statement and 11 other columns correspond to various information about the speaker of the statement. Descriptions of these attributes are given below:

1. **Label**: Each row of data is classified into six different types, namely
   (a) **Pants-fire:** Means the speaker has delivered a blatant lie .
   (b) **False:** Means the speaker has given totally false information.
   (c) **Barely-true:** Chances of the statement depending on the context is hardly true. Most of the contents in the statements are false.
   (d) **Half-true:** Chances of the content in the statement is approximately half.
   (e) **Mostly-true:** Most of the contents in the statement are true.
   (f) **True:** Content is true.
2. **Statement by the politician:** This statement is a short statement.
3. **Subjects:** This corresponds to the content of the text. For examples, foreign policy, education, elections etc.

4. **Speaker:** This contains the name of the speaker of the statement.
5. **Speaker's job title:** This specifies the position of the speaker in the party.
6. **State information:** This specifies in which state the statement was delivered.
7. **Party affiliation:** This denotes the name of the party of the speaker belongs to.
8. The next five columns are the counts of the speaker's statement history. They are:
   (a) **Pants fire count;**
   (b) **False count;**
   (c) **Barely true count;**
   (d) **Half false count;**
   (e) **Mostly true count.**
9. **Context:** This corresponds to the venue or location of the speech or statement.

The dataset consists of three sets, namely a training set of 10,269 statements, a validation set of 1,284 statements and a test set of 1,266 statements.

4 Experiments and Results

In this section, we report on the experimental setup, evaluation results, and the necessary analysis.

4.1 Experimental Setup

All the experiments are conducted in a python environment. The libraries of python are required for carrying out the experiments are **Keras**, **NLTK**, **Numpy**, **Pandas**, **Sklearn**. We evaluate the performance of the system in terms of accuracy, precision, recall, and F-score metrics.

4.2 Results and Analysis

We report the evaluation results in Table 3 that also show the comparison with the system as proposed in [16] and [17].

Table 3. Overall evaluation results

Model	Network	Attributes taken	Accuracy
William Yang Wang[16]	Hybrid CNN	All	0.274
Y. Long et al.[17]	Hybrid LSTM	All	0.415
Bi-LSTM Model	Bi-LSTM	All	0.4265
CNN Model	CNN	All	0.4289
Our Proposed Model	RNN-CNN combined	All	**0.4487**

We depict the overall evaluation results in Table 3 along with the other existing models. This shows that our model performs better than the existing state-of-the-art model as proposed in [17]. This state-of-the-art model was a hybrid LSTM, with an accuracy of 0.415. On the other hand, our proposed model shows 0.4265, 0.4289 and 0.4487 accuracies for Bi-LSTM, CNN and the combined CNN+Bi-LSTM model, respectively. This clearly supports our assumption that capturing temporal patterns using Bi-LSTM and hidden features using CNN are useful, channelizing each profile attribute through a different neural layer is important, and the meaningful combination of these separate attribute layers to capture relations between attributes, is effective.

Table 4. Evaluation of Bi-LSTM model: precision, recall, and F1 score

	precision	**recall**	**F1-score**	**No. of instances**
PANTS-FIRE	0.73	0.35	0.47	92
FALSE	0.47	0.53	0.50	249
BARELY-TRUE	0.58	0.32	0.41	212
HALF-TRUE	0.39	0.46	0.42	265
MOSTLY-TRUE	0.33	0.66	0.44	241
TRUE	0.88	0.14	0.23	207
Avg/Total	0.53	0.43	0.41	1266

Table 5. Evaluation of CNN model: precision, recall, F1 score

	precision	**recall**	**F1-score**	**No. of instances**
PANTS-FIRE	0.67	0.39	0.49	92
FALSE	0.36	0.63	0.46	249
BARELY-TRUE	0.50	0.36	0.42	212
HALF-TRUE	0.42	0.46	0.44	265
MOSTLY-TRUE	0.41	0.49	0.45	241
TRUE	0.70	0.16	0.26	207
Avg/Total	0.48	0.43	0.42	1266

We also report the precision, recall and F-score measures for all the models. Table4, Table 5 and Table 6 depict the evaluation results of CNN, Bi-LSTM and the combined model of CNN and Bi-LSTM, respectively. The evaluation shows that on the precision measure the combined model performs best with an average precision of **0.55** while that of Bi-LSTM model is 0.53 and CNN model is 0.48. The combined model of CNN and Bi-LSTM even performs better with respect to recall and F1-Score measures. The combined model yields the average recall of **0.45** and average F1-score of **0.43** while that of Bi-LSTM model is 0.43

Table 6. Evaluation of Bi-LSTM, CNN combined model: precision, recall, F1 score

	precision	recall	F1-score	No. of instances
PANTS-FIRE	0.70	0.43	0.54	92
FALSE	0.45	0.61	0.52	249
BARELY-TRUE	0.61	0.32	0.42	212
HALF-TRUE	0.35	0.73	0.47	265
MOSTLY-TRUE	0.50	0.36	0.42	241
TRUE	0.85	0.14	0.24	207
Avg/Total	**0.55**	**0.45**	**0.43**	1266

Table 7. Sample text with wrongly predicted label and original label. Spk is speaker, and P, F, B, H, M is speaker's previous count of Pants-fire, False, Barely-true, Half-true, Mostly-true respectively.

Label	Statement	St Type	Spk	Spk's Job	State	Party	Context	P	F	B	H	M	Predicted Label
barely-true	We know there are more Democrats in Georgia than Republicans. We know that for a fact.	elections	mike-berlon	none	Georgia	democrat	an article	1	0	0	0	0	False

and 0.41, respectively and of the CNN model is 0.43 and 0.42, respectively. On further analysis, we observe that although the performance (based on precision, recall, and F1-score) of each of the models for every individual class is close to the average performance, but in case of the prediction of the class label **TRUE** the performance of each model varies a lot from the respective average value. The precisions of TRUE is promising (Bi-LSTM model:0.88, CNN model: 0.7, Combined model:**0.85**), but the recall (Bi-LSTM model:0.14, CNN model: 0.16, Combined model:0.14) and the F1-score (Bi-LSTM model:0.23, CNN model: 0.26, Combined model:0.24) are very poor. This entails the fact that our proposed model predicts comparatively less number of instances as TRUE, but when it does the prediction is very accurate. Thus it can be claimed that if a statement is predicted as **True** by our proposed model then one can rely on that with high confidence. Although our model performs superior compared to the existing state-of-the-art, still the results were not error free. We closely analyze the models' outputs to understand their behavior and perform both quantitative as well as qualitative error analysis. For quantitative analysis, we create the confusion matrix for each of our models. Confusion matrix corresponding to the experiment 1 i.e with Bi-LSTM model is given in Table8, corresponding to experiment 2 i.e with CNN model is given in Table9 and corresponding to our

final experiment i.e with RNN-CNN combined model is given in Table 10.

From these quantitative analysis it is seen that in majority of the cases the test data statements originally labeled with **Pants-Fire** class gets confused with the **False** class, statements originally labeled as **False** gets confused with **Barely true** and **half true** classes, statements originally labeled as **Half true** gets confused with **Mostly True** and **False** class, statements originally labeled as **Mostly true** gets confused with **Half True**, statements originally labeled with **True** gets confused with **Mostly True** class.

It is quite clear that errors were mostly concerned with the classes, overlapping in nature. Confusion is caused as the contents of the statements belonging to these classes are quite similar. For example, the difference between 'Pants-Fire' and 'False' class is that only the former class corresponds to the false information with more intensity. Likewise 'Half True' has high similarity to 'False', and 'True' with 'Mostly True'. The difference between 'True' and 'Mostly True' is that the later class has some marginal amount of false information, while the former does not.

For qualitative analysis, we closely look at the actual statements and try to understand the causes of misclassifications. We come up with some interesting facts. There are some speakers whose statements are not present in the training set, but are present in the test set. For few of these statements, our model tends to produce wrong answers. Let us consider the example given in Table7. For this speaker, there is no training data available and also the count history of the speaker is very less. So our models assign an incorrect class. But it is to be noted that even if there is no information about the speaker in the training data and the count history of the speaker is almost empty, still we are able to generate a prediction of a class that is close to the original class in terms of meaning.

It is also true that classifiers often make mistakes in making the fine distinction between the classes due to the insufficient number of training instances. Thus, classifiers tend to misclassify the instances into one of the nearby (and overlapped) classes.

Table 8. Confusion matrix of the Bi-LSTM model

Actual\Predicted	**Pants-Fire**	**False**	**Barely-True**	**Half-True**	**Mostly-True**	**True**
Pants-Fire	32	35	3	8	14	0
False	4	131	16	36	59	3
Barely-True	5	31	68	48	60	0
Half-True	0	38	8	123	95	1
Mostly-True	1	20	8	54	158	0
True	2	25	15	47	90	28

Table 9. Confusion matrix of the CNN model

Actual\Predicted	**Pants-Fire**	**False**	**Barely-True**	**Half-True**	**Mostly-True**	**True**
Pants-Fire	36	35	6	11	2	2
False	7	156	21	30	28	7
Barely-True	5	66	76	34	29	2
Half-True	2	75	14	123	48	3
Mostly-True	1	53	17	51	119	0
True	3	44	18	44	65	33

Table 10. Confusion matrix of the Bi-LSTM+CNN combined model

Actual\Predicted	**Pants-Fire**	**False**	**Barely-True**	**Half-True**	**Mostly-True**	**True**
Pants-Fire	40	34	4	10	4	0
False	7	152	10	67	11	2
Barely-True	4	48	68	83	9	0
Half-True	0	43	7	193	20	2
Mostly-True	2	31	9	112	86	1
True	4	31	13	89	41	29

5 Conclusion and Future Works

In this paper, we have tried to address the problem of fake News detection by looking into short political statements made by the speakers in different types of daily access media. The task was to classify any statement into one of the fine-grained classes of fakeness. We have built several deep learning models, based on CNN, Bi-LSTM and the combined CNN and Bi-LSTM model. Our proposed approaches mainly differ from previously mentioned models in system architecture, and each model performs better than the state of the art as proposed in [17], where the statements were passed through one LSTM and all the other details about speaker's profile through another LSTM. On the other hand, we have passed every different attribute of speaker's profile through a different layer, captured the relations between the different pairs of attributes by concatenating them. Thus, producing a meaningful vector representation of relations between speaker's attributes, with the help of which we obtain the overall accuracy of 44.87%. By further exploring the confusion matrices we found out that classes which are closely related in terms of meaning are getting overlapped during prediction. We have made a thorough analysis of the actual statements, and derive some interesting facts. There are some speakers whose statements are not present in the training set but present in the test set. For some of those statements, our model tends to produce the wrong answers. This shows the importance of speakers' profile information for the task. Also as the classes and the meaning of the classes are very near, they tend to overlap due to less number of examples in training data.

We would like like to highlight some of the possible solutions to solve the problems that we encountered while attempted to solve fake news detection problem in a more fine-grained way.

- More labeled data sets are needed to train the model more accurately. Some semi-supervised or active learning models might be useful for this task.
- Along with the information of a speaker's count history of lies, the actual statements are also needed in order to get a better understanding of the patterns of the speaker's behavior while making a statement.

Fake news detection into finely grained classes that too from short statements is a challenging but interesting and practical problem. Hypothetically the problem can be related to **Sarcasm detection**[21] problem. Thus it will also be interesting to see the effect of implementing the existing methods that are effective in sarcasm detection domain in Fake News detection domain.

References

1. Fernandez, M., Alani, H.: Online misinformation: Challenges and future directions. In: Companion Proceedings of the The Web Conference 2018. WWW '18, Republic and Canton of Geneva, Switzerland, International World Wide Web Conferences Steering Committee (2018) 595–602
2. Chen, T., Wu, L., Li, X., Zhang, J., Yin, H., Wang, Y.: Call attention to rumors: Deep attention based recurrent neural networks for early rumor detection. CoRR **abs/1704.05973** (2017)
3. Rubin, V., Conroy, N., Chen, Y.: Towards news verification: Deception detection methods for news discourse. (2015)
4. Tacchini, E., Ballarin, G., Vedova, M.L.D., Moret, S., de Alfaro, L.: Some like it hoax: Automated fake news detection in social networks. CoRR **abs/1704.07506** (2017)
5. Eshraqi, N., Jalali, M., Moattar, M.H.: Spam detection in social networks: A review. In: 2015 International Congress on Technology, Communication and Knowledge (ICTCK). (2015) 148–152
6. Duppada, V.: "attention" for detecting unreliable news in the information age. In: AAAI Workshops. (2018)
7. Shu, K., Sliva, A., Wang, S., Tang, J., Liu, H.: Fake news detection on social media: A data mining perspective. SIGKDD Explor. Newsl. **19** (2017) 22–36
8. Bessi, A., Ferrara, E.: Social bots distort the 2016 u.s. presidential election online discussion. First Monday **21** (2016)
9. Castillo, C., Mendoza, M., Poblete, B.: Information credibility on twitter. In: Proceedings of the 20th International Conference on World Wide Web. WWW '11, New York, NY, USA, ACM (2011) 675–684
10. Gupta, A., Kumaraguru, P., Castillo, C., Meier, P. In: TweetCred: Real-Time Credibility Assessment of Content on Twitter. Springer International Publishing, Cham (2014) 228–243
11. Zhou, L., Twitchell, D.P., Qin, T., Burgoon, J.K., Nunamaker, J.F.: An exploratory study into deception detection in text-based computer-mediated communication. In: 36th Annual Hawaii International Conference on System Sciences, 2003. Proceedings of the. (2003) 10 pp.–
12. Banko, M., Cafarella, M.J., Soderland, S., Broadhead, M., Etzioni, O.: Open information extraction from the web. In: Proceedings of the 20th International Joint Conference on Artifical Intelligence. IJCAI'07, San Francisco, CA, USA, Morgan Kaufmann Publishers Inc. (2007) 2670–2676

13. Bajaj, S.: the pope has a new baby ! fake news detection using deep learning. (2017)
14. Ma, J., Gao, W., Mitra, P., Kwon, S., Jansen, B.J., Wong, K.F., Cha, M.: Detecting rumors from microblogs with recurrent neural networks. In: Proceedings of the Twenty-Fifth International Joint Conference on Artificial Intelligence. IJCAI'16, AAAI Press (2016) 3818–3824
15. Ruchansky, N., Seo, S., Liu, Y.: Csi: A hybrid deep model for fake news detection. In: Proceedings of the 2017 ACM on Conference on Information and Knowledge Management. CIKM '17, New York, NY, USA, ACM (2017) 797–806
16. Wang, W.Y.: "liar, liar pants on fire": A new benchmark dataset for fake news detection. In: Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers), Association for Computational Linguistics (2017) 422–426
17. Long, Y., Lu, Q., Xiang, R., Li, M., Huang, C.R.: Fake news detection through multi-perspective speaker profiles. In: Proceedings of the Eighth International Joint Conference on Natural Language Processing (Volume 2: Short Papers), Asian Federation of Natural Language Processing (2017) 252–256
18. Kim, Y.: Convolutional neural networks for sentence classification. In: Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP), Association for Computational Linguistics (2014) 1746–1751
19. Hochreiter, S., Schmidhuber, J.: Long short-term memory. Neural computation **9** (1997) 1735–80
20. Mikolov, T., Sutskever, I., Chen, K., Corrado, G., Dean, J.: Distributed representations of words and phrases and their compositionality. In: Proceedings of the 26th International Conference on Neural Information Processing Systems - Volume 2. NIPS'13, USA, Curran Associates Inc. (2013) 3111–3119
21. Joshi, A., Bhattacharyya, P., Carman, M.J.: Automatic sarcasm detection: A survey. ACM Comput. Surv. **50** (2017) 73:1–73:22