A Benchmark Study on Machine Learning Methods for Fake News Detection

Preprint · May 2019

CITATIONS
0

READS
1,324

4 authors, including:

Junaed Younus Khan
Bangladesh University of Engineering and Technology
9 PUBLICATIONS **1** CITATION
SEE PROFILE

Md. Tawkat Islam Khondaker
Bangladesh University of Engineering and Technology
9 PUBLICATIONS **1** CITATION
SEE PROFILE

Anindya Iqbal
Bangladesh University of Engineering and Technology
41 PUBLICATIONS **196** CITATIONS
SEE PROFILE

A Benchmark Study on Machine Learning Methods for Fake News Detection

Junaed Younus Khan*[1], Md. Tawkat Islam Khondaker*[1], Anindya Iqbal[1] and Sadia Afroz[2]

[1]Department of Computer Science and Engineering, Bangladesh University of Engineering and Technology
[2]International Computer Science Institute
1405051.jyk@ugrad.cse.buet.ac.bd, 1405036.mtik@ugrad.cse.buet.ac.bd, anindya@cse.buet.ac.bd, sadia@icsi.berkeley.edu

May 14, 2019

Abstract

The proliferation of fake news and its propagation on social media have become a major concern due to its ability to create devastating impacts. Different machine learning approaches have been attempted to detect it. However, most of those focused on a special type of news (such as political) and did not apply many advanced techniques. In this research, we conduct a benchmark study to assess the performance of different applicable approaches on three different datasets where the largest and most diversified one was developed by us. We also implemented some advanced deep learning models that have shown promising results.

Introduction

There was a time when if anyone needed any news, he or she would wait for the next-day newspaper. However, with the growth of online newspapers who update news almost instantly, people have found a better and faster way to be informed of the matter of his/her interest. Nowadays social-networking systems, online news portals, and other online media have become the main sources of news through which interesting and breaking news are shared at a rapid pace. However, many news portals serve special interest by feeding with distorted, partially correct, and sometimes imaginary news that is likely to attract the attention of a target group of people. Fake news has become a major concern for being destructive sometimes spreading confusion and deliberate disinformation among the people.

The term fake news has become a buzz word these days. However, an agreed definition of the term "fake news is still to be found. It can be defined as a type of yellow journalism or propaganda that consists of deliberate misinformation or hoaxes spread via traditional print and broadcast news media or online social media [15]. These are published usually with the intent to mislead in order to damage a community or person, create chaos, and gain financially or politically. Since people are often unable to spend enough time to cross-check reference and be sure of the credibility of news, automated detection of fake news is indispensable. Therefore, it is receiving great attention from the research community.

There are many instances where cleverly designed fake news had severe consequence by instigating religious or ethnic groups against innocent victims. On October 17, 2018, United States Congressman Matt Gaetz (R-FL) posted a video to Twitter and suggested, without evidence, that showed a group of people being paid by billionaire George Soros to join a migrant caravan and storm the United States border. The video was miscaptioned and the tweet contained factual inaccuracies.[1] On 23 June 2018, a series of horrifying images and videos began to circulate on Facebook. One showed a mans skull hacked open that was viewed more than 11,000 times. The Facebook users who posted the images claimed they showed a massacre underway in the Gashish district of Plateau State, Nigeria by Fulani Muslims who were killing Christians from the regions Berom ethnic minority. As a consequence, a massacre did happen in Gashish that weekend and somewhere between 86 and 238 Berom people were killed, according to estimates made

*The authors contribute equally to this paper. Names are sorted in alphabetical order.
[1]https://www.snopes.com/fact-check/soros-caravan-refugees/

by the police and by local community leaders. However, some of the most incendiary images and videos were totally irrelevant to the violence in Gashish. The video showing a man's head was cut, was not even happened in Nigeria and it was recorded in Congo, in 2012.[2]

The prior works on fake news detection have applied several traditional machine learning methods and neural networks to detect fake news. However, they have focused on detecting news of particular types (such as political) [19, 24, 26]. Accordingly, they developed their models and designed features for specific datasets that match their topic of interest. It is likely that these approaches would suffer from dataset bias and are likely to perform poorly on news of another topic. Some of the existing studies have also made comparisons among different methods of fake news detection. Wang [26] has built a benchmark dataset namely, Liar and experimented some existing models on that dataset. The comparison result hints us how different models can perform on a structured dataset like Liar. However, the length of this dataset is not sufficient for neural network analysis and some models were found to suffer from overfitting. Gilda has explored some traditional machine learning approaches [10]. However, many advanced machine learning models, e.g., neural network based ones are not applied that have been proved best in many text classification problems.

An important limitation of prior comparative studies is that these are carried out on a specific type of dataset, it is difficult to reach a conclusion about the performance of various models. Moreover, these works have focused on a limited number of features that have resulted in the incomplete exploration of potential characteristics of fake news. In this research, our goal is to present a comparative performance analysis of existing methods by implementing each one on two of the available datasets and another one prepared by us combining news of distributed topics. We also incorporate different features from existing works and investigate the performance of some successful text classification techniques that are yet to be applied for fake news detection to the best of our knowledge.

Specifically, in this study, we investigate:

- Performance of traditional machine learning and neural network models on three different datasets with the largest dataset developed by us. Unlike other two datasets that contain news on the specific topic, we have covered a wide range of topics and accumulated five times more news compared to others.
- Performance of some advanced models such as convolutional-LSTM, character-level convolutional-LSTM, convolutional-HAN which are not used in fake news detection yet to the best of our knowledge.
- Performance of different models that have shown promising results on other similar problems.
- Topic-based analysis on misclassified news, especially deceptive news that is falsely identified as true.

We observe that the performance of models is not dataset invariant and so it is quite difficult to obtain a unique superior model for all datasets. We have also found that traditional machine learning architecture like Naive Bayes can achieve very high accuracy with proper feature selection. On a small dataset with less than 100k news articles, Naive Bayes(with n-gram) can be a primary choice as it achieves similar performance compared to neural network-based high overhead models. Most importantly, our newly explored neural network based models in this fake news detection achieve as high as 95% accuracy and F1-score and exhibit improvement in performance with the enrichment of dataset. Our codes and dataset are publicly available.[3]

Related Work

Most of the existing researches have been focused on classifying online news and social media posts. Different methods have been proposed by different researches for deception detection.

Fake news can be classified into various types. For instance, Conroy, Rubin, and Chen have mentioned three types of fake news: Serious Fabrications (Type A), Large-Scale Hoaxes (Type B), Humorous Fakes (Type C) [20]. In simpler words, Fake news is a news article that is intentionally and verifiably false and could mislead readers [2]. This narrow definition is useful in the sense that it is able to eliminate the ambiguity between fake news and other related concepts e.g., hoaxes, and satires.

[2] https://www.bbc.co.uk/news/resources/idt-sh/nigeria_fake_news

[3] https://anonymous.4open.science/repository/b7c0d56e-9e4b-434b-87f4-516d9a0f0516/

Shu, Silva, Wang, Jiliang and Liu [22] have proposed to use linguistic-based features such as total words, characters per word, frequencies of large words, frequencies of phrases (i.e., n-grams and bag-of-words approaches [9]), parts-of-speech (POS) tagging.

Conroy, Rubin, and Chen [6] have noted that simple content-related n-grams and part-of-speech (POS) tagging have been proven insufficient for the classification task. Rather, they suggested Deep Syntax analysis using Probabilistic Context-Free Grammars (PCFG) and noted that this method was used by Feng, Banerjee, and Choi [8] to distinguish rule categories (lexicalized, non-lexicalized, parent nodes, etc.) for deception detection with 85-91% accuracy. However, Shlok Gilda has mentioned that while bi-gram TF-IDF yields highly effective models for detecting fake news, the PCFG features do little to add to the models efficacy [10].

Many research works also suggested the use of sentiment analysis for deception detection as some correlation might be found between the sentiment of the news article and its type. Conroy, Rubin, Chen, and Cornwell hypothesized expanding the possibilities of word-level analysis by measuring the utility of features like part of speech frequency, and semantic categories such as generalizing terms, positive and negative polarity (sentiment analysis) [19].

Mathieu Cliche in his sarcasm detection blog has described the detection of sarcasm on twitter through the use of n-grams, words learned from tweets specifically tagged as sarcastic. His work also includes the use of sentiment analysis as well as identification of topics (words that are often grouped together in tweets) to improve prediction accuracy [5]. Wang has compared the performance of SVM, LR, Bi-LSTM, CNN models on their proposed dataset LIAR [26].

Several research works show promising results in detecting fake news through neural network and tracing user propagation. Wang in his [26] has built a hybrid convolutional neural network model that outperform other traditional machine learning models. Hannah Rashkin et al. [18] have performed an extensive analysis of linguistic features and shown the impressive result of LSTM. Singhania et al. [23] have proposed a three-level hierarchical attention network one each for words, sentences and the headline of a news article. Ruchansky et al. [21] have created the CSI model where they have captured text, the response of an article and the source characteristics based on users behavior.

Background

In this section, we discuss some important theoretical concepts which are related to our study.

LIWC and Empath Generated Features

Linguistic Inquiry and Word Count (LIWC) dictionary includes a word classification and count tool. LIWC reads the texts from a given dataset and its text analysis module then compares each word in the text against a user-defined dictionary. The dictionary identifies which words are associated with which psychologically-relevant categories. Then it calculates the percentage of total words that match each of the dictionary categories. LIWC can be used in computational linguistics as a source of features for deception detection [12, 16].

Empath, similar to LIWC, is a tool that can generate new lexical categories from a small set of seed terms. It draws connotations between words and phrases by deep learning a neural embedding across more than 1.8 billion words of modern fiction. Empath's data-driven, human validated categories have been verified to be highly correlated (r=0.906) with similar categories in LIWC [7].

C-LSTM

CNN can extract local features of input and RNN (recurrent neural network) can process sequence input and learn the long-term dependencies [27]. C-LSTM utilizes this fact and uses CNN to extract a sequence of higher-level phrase representations, and that are fed into a long short-term memory recurrent neural network (LSTM) to obtain the sentence representation [32]. The convolutional layer applies a matrix-vector operation to each n-gram word of a given sentence and LSTM propagates historical information via neural network chain. First, CNN is constructed on top of the pre-trained word vectors to learn a higher-level representation of n-grams. Then to learn sequential correlations from higher-level sequence representations, the feature maps of CNN are organized as sequential window features to serve as the input of LSTM. In this way, each sentence is transformed into a successive window (n-gram) features to help disentangle factors within sentences.

Hierarchical Attention Layer

Hierarchical Attention Network (HAN) is designed to capture insights about document structure [30]. Since, documents have a hierarchical structure (words form sentences, sentences form a document), we likewise construct a document representation by first building representations of sentences and then aggregating those into document representation.

This model comprises four steps:

- **Word Encoder:** A bidirectional GRU is to get annotations of words by capturing contextual information from both directions for words.
- **Word Attention:** Attention mechanism is introduced to extract words that are important to the meaning of the sentence and aggregate the representation of those informative words to form a sentence vector.
- **Sentence Encoder:** A bidirectional GRU encodes the sentences to build up document vectors.
- **Sentence Attention:** Attention mechanism is introduced again to retrieve a sentence level context vector and use the vector to measure the importance of the sentences. As a final outcome, we get a document vector incorporating all the information of sentences in a document.

Character-level C-LSTM

This architecture accepts a sequence of encoded characters as input [31]. The encoding is done by prescribing a fixed length of alphabets for the input language and then quantizing each character using one hot encoding. The characters are then transformed into fixed-length vectors. The character quantization order is backward so that the latest reading on characters is always placed near the beginning of the output, making it easy for fully connected layers to associate weights with the latest reading.

We combine the word and character-level models by feeding a word-level LSTM [Exploring the Limits of Language Modeling]. This word-level LSTM is fed to a Bidirectional LSTM which receives high-level document representation for the character encoding.

Datasets

We have considered three datasets to measure the performance of different methods. The characteristics of the datasets are presented here.

Liar

Liar[4] is a publicly available dataset that has been used in [26]. It includes 12.8K human labeled short statements from POLITIFACT.COMs API[5]. It comprises six labels of truthfulness ratings: pants-fire, false, barely-true, half-true, mostly-true, and true. In our work, we try to differentiate real news from all types of hoax, propaganda, satire and misleading news. Hence, we mainly focus on classifying news as real and fake. For the binary classification of news, we transform these labels into two labels. Pants-fire, false, barely-true are contemplated as fake and half-true, mostly-true and true are as true. Our converted dataset contains 56% true and 44% fake statements. This dataset mostly deals with political issues that include statements of democrats and republicans, as well as a significant amount of posts from online social media. The dataset provides some additional meta-data like subject, speaker, job, state, party, context, history, but in the real life scenario, we may not have this meta-data always available. Therefore, we experiment on the texts of the dataset using textual features.

[4] https://www.cs.ucsb.edu/~william/data/liar_dataset.zip

[5] http://static.politifact.com.s3.amazonaws.com/api/v2apidoc.html

Fake or Real News

Fake or real news dataset[6] is developed by George McIntire. The fake news portion of this dataset was collected from Kaggle fake news dataset[7] comprising news of 2016 USA election cycle. The real news portion was collected from media organizations such as the New York Times, WSJ, Bloomberg, NPR, and the Guardian for the duration of 2015 or 2016. The GitHub repository of the dataset includes around 7.8k news with equal allocation of fake and real news and half of the corpus comes from political news.

Combined Corpus

Apart from the other two datasets, we have built a text corpus with plain text news. We have collected news from several sources of same time domain mostly from 2016 and partially from 2015 and 2017. Multiple types of fake news like a hoax, satire, propaganda have come from The Onion, Borowitz Report, Clickhole, American News, DC Gazette, Natural News and Activist Report. We have collected the real news from the trusted sources like the New York Times, Breitbart, CNN, Business Insider, the Atlantic, Fox News, Talking Points Memo, Buzzfeed News, National Review, New York Post, the Guardian, NPR, Gigaword News, Reuters, Vox, and the Washington Post. One important property of this corpus is that it incorporates topic variance (Figure 1) including national and international politics, economy, investigation, health-care, sports, entertainment and others. This corpus contains around 80k news of which 51% are true and 49% are deemed as fake.

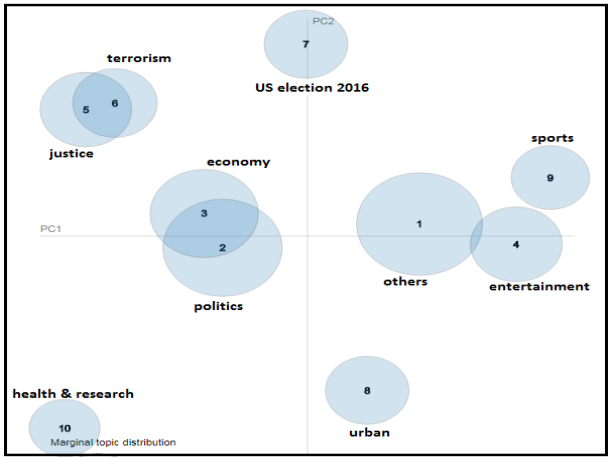

Figure 1: Intertopic Distance Map (considering 10 topics via multidimensional scaling).

Methodology

In this section, we describe preprocessing and feature extraction from raw texts of our datasets.

Dataset Preprocessing

Before feeding into the models, raw texts of news required some preprocessing. We first eliminated unnecessary IP and URL addresses from our texts. Next step was to remove stop words. After that, we cleaned our corpus by correcting the spelling of words. We split every text by white-space and remove suffices from words by stemming them with Snowball Stemmer from NLTK library.[8] Finally, we rejoined the word tokens by white-space to present our clean text corpus which had been tokenized later for feeding our models.

[6] https://github.com/GeorgeMcIntire/fake_real_news_dataset, accessed 20 October 2018
[7] https://www.kaggle.com/mrisdal/fake-news
[8] https://www.nltk.org/_modules/nltk/stem/snowball.html

Feature Extraction

The performance of machine learning models depends on a great deal on features design. Hence we have extracted a wide range of features for beyond those need in other relevant works.

Lexical and Sentiment Features Extraction:

Some existing works suggested the potential of lexical and sentiment features for fake news detection [19, 22]. We used word count, average word length, article length, count of numbers, count of parts of speech(adjective), count of exclamation mark as lexical features. For sentiment features, we measured the positive, negative and neutral sentiment of every article. To extract sentiment features, we used *SentimentIntensityAnalyzer* function of python NLTK library.

n-gram Feature Extraction:

Word-based n-gram was used to represent the context of the document and generate features to classify the document as fake and real [1, 3, 11, 18, 29]. Many existing works used unigram (n=1) and bigram (n=2) approaches for fake news detection [19, 25]. We also evaluated these two approaches in this benchmark. We used *TfidfVectorizer* function of python sklearn.feature_extraction library to generate TF-IDF n-gram features.

Feature Extraction Using Empath Tool:

We used default Empath's library function which generates categories for topic variance such as violence, crime, pride, sympathy, deception, war and so on. We used these categories as features to identify key information in a news article.

Pre-trained Word Embedding:

For neural network models, word embeddings were initialized with 100-dimensional pre-trained embeddings from GloVe [17]. GloVe is an unsupervised learning algorithm for obtaining vector representations for words. It was trained on a dataset of one billion tokens (words) with a vocabulary of 400 thousand words.

Classification Approaches

In this section, we describe the experimental setup of different models. We also provide implementation details of our newly explored approaches in fake news detection.

Traditional Machine Learning Models

Here we will describe the traditional machine learning models that we have explored.

SVM, LR and Decision Tree Models:

We built our first three models using SVM, LR and Decision Tree with the lexical and sentiment features. Among the four main variants of the SVM kernel, which we discussed earlier, we used the linear one. For LR and Decision Tree, we used default library functions of python.

Naive Bayes Model:

We used Multinomial Naive Bayes classifier as our next model. We fed the n-gram features into it. We used the python library function named MultinomialNB for this.

k-NN Model:

We fed the features generated by empath into *k*-NN classifier. For *k*-NN, we used python library function named KNeighborsClassifier with K=5.

Neural Network-Based and Deep Learning Models

Now we will describe the neural network based and deep learning models used in our experiment.

CNN Model:

The one-dimensional convolutional model was initialized with 100-dimensional pre-trained GloVe embeddings. It contained 128 filters of filter size 3 and a max pooling layer of pool size 2 is selected. Dropout probability of 0.8 was preserved which was expunged for Combined Corpus. The model was compiled with ADAM optimizer with learning rate 0.001 to minimize binary cross entropy loss. A sigmoid activation function was used for the final output layer. A batch size of 64 and 512 was used for training the datasets over 10 epochs.

LSTM:

Our LSTM model was pre-trained with 100-dimensional GloVe embeddings. Output dimension and time steps were set to 300. ADAM optimizer with learning rate 0.001 was applied to minimize binary cross entropy loss and sigmoid was the activation function for the final output layer. Finally, this model was trained over 10 epochs with batch size 64 and 512.

Bi-LSTM:

Usually, news which is deemed as fake is not fully comprised with false information, rather it is blended with true information. To detect the anomaly in a certain part of the news, we need to examine it both with previous and next events of action. We constructed Bi-LSTM model to perform this task. Bi-LSTM was initialized with 100-dimensional pre-trained GloVe embeddings. Output dimension of 100 and time steps of 300 was applied. ADAM optimizer with learning rate 0.001 was used to minimize binary cross entropy loss. Training batch size was set to 128 and loss over each epoch was observed with a callback. Learning rate was reduced by a factor of 0.1 and patience was set at 10. We also used an early stop to monitor validation accuracy to check whether the accuracy was deteriorating for 5 epochs. Loss of the binary cross-entropy of the ensemble model was minimized by ADAM with learning rate 0.0001.

C-LSTM:

The C-LSTM based model contained one convolutional layer and one LSTM layer. We used 128 filters with filter size 3 on top of which a max pooling layer of pool size 2 was set. We fed it to our LSTM architecture with 100 output dimensions and dropout 0.2. Finally, we used sigmoid as the activation function of our output layer.

HAN:

We used Keras library with Tensorflow backend to implement attention mechanism. Hierarchical attention network consisted of two attention mechanisms for word-level and sentence-level encoding. Prior to training, we set the maximum number of sentences in a news article as 20 and the maximum number of words in a sentence as 100. In both level encoding, a bidirectional GRU with output dimension 100 was fed to our customized attention layer. We used word encoder as input to our sentence encoder time-distributed layer. We optimized our model with ADAM that learned at a rate of 0.001.

Convolutional HAN:

In order to extract high-level features of input, we incorporated a one-dimensional convolutional layer before each bidirectional GRU layer in HAN. This layer selected features of each tri-gram from the news article before feeding it to the attention layer.

Character-level C-LSTM:

First, we created character level embedding by retrieving character set from our news corpus. On the top of the embedding layer, we created two convolutional layers of 128 and 256 filters with filter size 3 and 5, respectively. We used max-pooling of pool size 2 and dropout of 0.2. Then, we built bidirectional LSTM with recurrent dropout 0.2

which was fed to the input of word-level encoding. In word-level encoding, we built a fully connected layer with ReLU activation function on the top of a bidirectional LSTM. Finally, we built a sigmoid activation layer to generate the binary result of our model and compiled this architecture with ADAM optimizer.

Results

In this section, we describe in-depth performance analysis of our traditional machine learning and neural network based deep learning models. We present the best performance for each dataset and each matrix in bold. We calculate accuracy, precision, recall, and f1-score for fake and real class, and find their average, weighted by support (the number of true instances for each class) and report an average score of these metrics.

Result Applying Traditional Machine Learning Models

Table 1: Performance of Traditional Machine Learning Models

		Datasets											
		Liar				***Fake or Real News***				***Combined Corpus***			
Model	**Feature**	Acc	Pre	Rec	F1-Score	Acc	Pre	Rec	F1-Score	Acc	Pre	Rec	F1-Score
SVM	Lexical	.56	.56	.56	.48	.67	.67	.67	.67	.71	.78	.71	.72
SVM	Lexical +Sentiment	.56	.57	.56	.48	.66	.66	.66	.66	.71	.77	.71	.72
LR	Lexical +Sentiment	0.56	.56	.56	.51	.67	.67	.67	.67	.76	.79	.76	.77
Decision Tree	Lexical +Sentiment	.51	.51	.51	.51	.65	.65	.65	.65	.67	.71	.69	.7
Adaboost	Lexical +Sentiment	.56	.56	.56	.54	.72	.72	.72	.72	.73	.74	.73	.74
Naive Bayes	Unigram (TF-IDF)	.60	.60	.60	.57	.86	.88	.86	.86	**.95**	**.95**	**.95**	**.95**
Naive Bayes	Bigram (TF-IDF)	**.60**	**.59**	**.60**	**.59**	**.90**	**.91**	**.90**	**.90**	.93	.93	.93	.93
k-NN	Empath Features	.53	.53	.53	.53	.71	.72	.71	.71	.70	.70	.70	.70

In Table 1, we have reported the performance of various traditional machine learning models in detecting fake news. We observe that among the traditional machine learning models, Naive Bayes, with n-gram (bigram TF-IDF) features, has performed the best. In fact, it has achieved almost 94% accuracy on our combined corpus. We also find that addition of sentiment features along with lexical features does not improve the performance significantly. For lexical and sentiment features, SVM and LR models have performed better than other traditional machine learning models as suggested by most of the prior studies [4, 19, 24, 26, 28]. On the other hand, though features generated using Empath have been used for understanding deception in a review system [7], they have not shown promising performance for fake news detection.

Result Applying Neural Network Based Models

Table 2: Performance of Neural Network Based Models

		Datasets											
		Liar				***Fake or Real News***				***Combined Corpus***			
Model	**Feature**	Acc	Pre	Rec	F1-Score	Acc	Pre	Rec	F1-Score	Acc	Pre	Rec	F1-Score
CNN	Glove Em-bedding	.58	.58	.58	.58	.86	.86	.86	.86	.93	.93	.93	.93
LSTM		.54	.29	.54	.38	.76	.78	.76	.76	.93	.94	.93	.93
Bi-LSTM		.58	.58	.58	.57	.85	.86	.85	.85	**.95**	**.95**	**.95**	**.95**
C-LSTM		.54	.29	.54	.38	.86	.87	.86	.86	**.95**	**.95**	**.95**	**.95**
HAN		.57	.57	.57	.56	.87	.87	.87	.87	.92	.92	.92	.92
Conv-HAN		**.59**	**.59**	**.59**	**.59**	.86	.86	.86	.86	.92	.92	.92	.92
Char-level C-LSTM	Character Em-bedding	.56	.56	.56	.54	**.95**	**.95**	**.95**	**.89**	.89	.89	.89	.89

In Table 2, we have reported performance of different neural models. Assessing performance on Liar, Fake or real news and combined corpus datasets, we observe that no model is uniquely superior to others.

The baseline CNN model is considered as the best model for Liar in [26], but we find it as the second best among

all the models. LSTM-based models are most vulnerable to overfitting for this dataset which is reflected by its performance. Although Bi-LSTM is also a victim of overfitting on Liar dataset as mentioned in [26], we find it third best neural network based model according to its performance on the dataset. The models successfully used for text classification like C-LSTM, HAN, Char-level C-LSTM hardly surmount overfitting problem for Liar dataset. Our hybrid Conv-HAN model exhibits the best performance among the neural models for the Liar dataset with 0.59 accuracy and 0.59 F1-score. Char-level C-LSTM shows an excellent performance in Fake or real news dataset and defeats others by a clear margin with its 0.95 accuracy and 0.95 F1-score. Other LSTM-based models show an improvement whereas CNN and Conv-HAN continue their impressive performance on this dataset. LSTM-based models exhibit their best performance on our Combined Corpus where both Bi-LSTM and C-LSTM achieve 0.95 accuracy and 0.95 F1-score. Char-level C-LSTM is an exception here that fails to achieve 0.90 accuracy and F1-score. CNN and all hierarchical attention models including Conv-HAN secure more than 0.90 accuracy and F1-score maintain a decent performance on this dataset. This result indicates that, although neural network-based models may suffer from overfitting for a small dataset(LIAR), they show high accuracy and f1-score on a moderately large dataset(Combined Corpus).

Discussion

In this section, we discuss the overall performance of our models and a topic-based analysis of false positive news.

Overall Model Performance

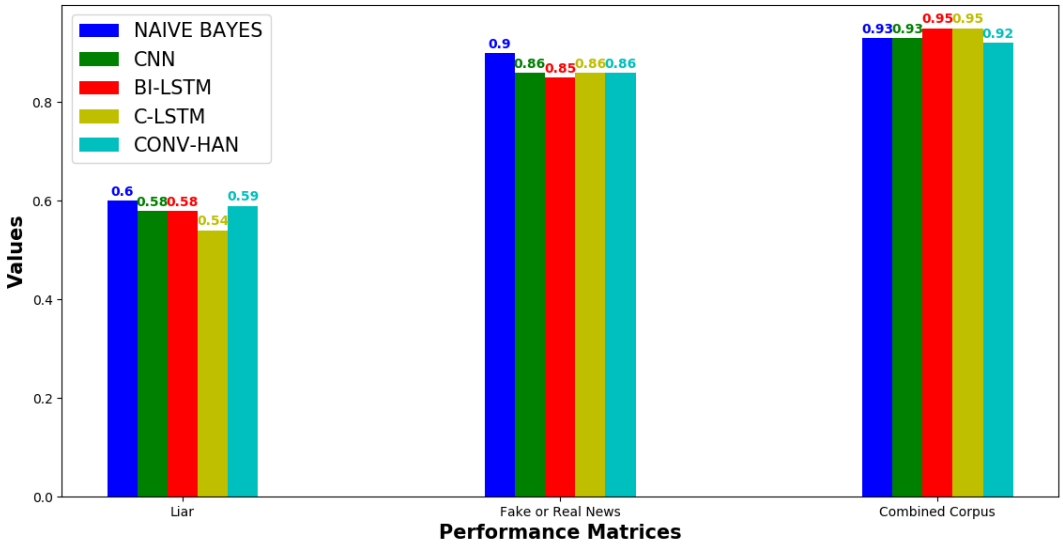

Figure 2: Accuracy comparison among models for three datasets.

Naive Bayes (with n-gram) model has shown the best performance among the traditional machine learning models while CNN, Bi-LSTM, C-LSTM, and Conv-HAN are the most promising ones among the NN based models (Figure 2). It is found that n-gram features show great promise in spam detection [13]. Hence, the outstanding performance of n-gram features in fake news detection is not surprising. We find that the performance of Naive Bayes (with n-gram) model is almost equivalent to the performances of these NN based models. Hence, Naive Bayes with n-gram is our recommended model for a small dataset. The one dimensional CNN model maintains a moderate performance on three datasets. On the other hand, LSTM based models show gradual improvement when the dataset length increases from LIAR to Combined Corpus. The more an article contains information, the less these models will be vulnerable to overfitting and the better they will perform. Hence, neural network-based models may show high performance on a larger dataset over 100k samples [14], but to avoid computational overhead and time complexity, Naive Bayes is a

good choice for a smaller dataset. Finally, our proposed hybrid model Conv-HAN exhibits high performance on all three datasets which definitely claims attention for future exploration with a larger dataset.

Performance Analysis on Article Length

We have investigated the relation between models' performance and length of news. Hence, we have observed the performance of Naive Bayes model with n-gram features (n=2) on 5000 data randomly selected from each of our three datasets. We find that when dataset size is fixed, the accuracy of the model is proportional to the average article length of news (Figure 3). This indicates that, with the increase of news article length, the model can extract more information to correctly detect the news.

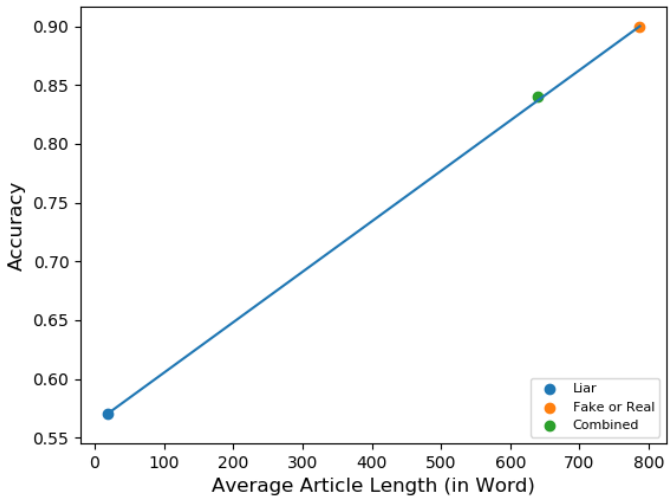

Figure 3: Relation between models' performance and article length (Trained on 5000 articles).

Topic Analysis of False Positive News

Figure 4: : Inter-topic Distance Map for false positive news.

The more a deceptive article is misconstrued as a real one, the more it can be hazardous. Hence, we concentrate our topic-based analysis on especially those fake news articles, which our model misclassifies as real (Figure 4). We

find that the most frequent words in these articles are 'said', 'study', and 'research'. The profuse use of the word 'said' indicates how fake news sources misconstrue quotes to make these as believable as possible and carry out their own agendas.

Table 3: Topic-wise percentage of false positive news

Topic	**News on Combined Corpus (%)**	**False Positive News (%)**
Politics	23.4	27.6
Health and Research	5.5	49.6
Miscellaneous	17.5	22.8

The topic-wise analysis shows that almost 50% of news articles are related to health and research-based topics although, we have trained our models on 5.5% of health and research-based and 23.4% political news. An equal proportion of false positive news on politics is justified by the fact that political news is easy to verify by other sources. But the high false positive rate of health and research related news is evidence that clickbait news on health and research can be produced convincingly. A slight change in the actual research article will still keep the fake news in the close proximity of the actual article which will be very difficult to judge due to unavailability of sufficient related resources. In this way, it is quite easy for clickbait news sources to attract people by publishing news on claiming invention of a vaccine for incurable diseases like terminal cancer. Hence, although in recent times, the media has focused mostly on combating against unauthentic political news, it should also pay attention to stop the proliferation of false health and research related news for public safety.

Conclusion and Future Work

In this study, we present an overall performance analysis of different approaches on three different datasets. We show that Naive Bayes with n-gram can attain analogous result to neural network-based models on a dataset with less than 100k news articles. The performance of LSTM-based models greatly depends on the length of the dataset as well as information given in a news article. With adequate information provided in a news article, LSTM-based models have a higher probability to overcome overfitting. Moreover, advanced models like C-LSTM, Conv-HAN and character level C-LSTM have shown high promise that demands further attention on these models in fake news detection. Finally, we perform a topic-based analysis that exposes the difficulty to correctly detect political, health and research related deceptive news. Our future plan is to experiment on a larger dataset to find how the traditional model like Naive Bayes competes against highly computational neural network-based models to detect fake news.

References

[1] Hadeer Ahmed, Issa Traore, and Sherif Saad. Detection of online fake news using n-gram analysis and machine learning techniques. In *International Conference on Intelligent, Secure, and Dependable Systems in Distributed and Cloud Environments*, pages 127–138. Springer, 2017.

[2] Hunt Allcott and Matthew Gentzkow. Social media and fake news in the 2016 election. *Journal of Economic Perspectives*, 31(2):211–36, 2017.

[3] Peter Bourgonje, Julian Moreno Schneider, and Georg Rehm. From clickbait to fake news detection: an approach based on detecting the stance of headlines to articles. In *Proceedings of the 2017 EMNLP Workshop: Natural Language Processing meets Journalism*, pages 84–89, 2017.

[4] Yimin Chen, Niall J Conroy, and Victoria L Rubin. Misleading online content: Recognizing clickbait as false news. In *Proceedings of the 2015 ACM on Workshop on Multimodal Deception Detection*, pages 15–19. ACM, 2015.

[5] Mathieu Cliche. The sarcasm detector, 2014.

[6] Niall J Conroy, Victoria L Rubin, and Yimin Chen. Automatic deception detection: Methods for finding fake news. In *Proceedings of the 78th ASIS&T Annual Meeting: Information Science with Impact: Research in and for the Community*, page 82. American Society for Information Science, 2015.

[7] Ethan Fast, Binbin Chen, and Michael S Bernstein. Empath: Understanding topic signals in large-scale text. In *Proceedings of the 2016 CHI Conference on Human Factors in Computing Systems*, pages 4647–4657. ACM, 2016.

[8] Song Feng, Ritwik Banerjee, and Yejin Choi. Syntactic stylometry for deception detection. In *Proceedings of the 50th Annual Meeting of the Association for Computational Linguistics: Short Papers-Volume 2*, pages 171–175. Association for Computational Linguistics, 2012.

[9] Johannes Fürnkranz. A study using n-gram features for text categorization. *Austrian Research Institute for Artifical Intelligence*, 3(1998):1–10, 1998.

[10] Shlok Gilda. Evaluating machine learning algorithms for fake news detection. In *Research and Development (SCOReD), 2017 IEEE 15th Student Conference on*, pages 110–115. IEEE, 2017.

[11] Mykhailo Granik and Volodymyr Mesyura. Fake news detection using naive bayes classifier. In *Electrical and Computer Engineering (UKRCON), 2017 IEEE First Ukraine Conference on*, pages 900–903. IEEE, 2017.

[12] Ángel Hernández-Castañeda and Hiram Calvo. Deceptive text detection using continuous semantic space models. *Intelligent Data Analysis*, 21(3):679–695, 2017.

[13] Johan Hovold. Naive bayes spam filtering using word-position-based attributes. In *CEAS*, pages 41–48, 2005.

[14] Armand Joulin, Edouard Grave, Piotr Bojanowski, and Tomas Mikolov. Bag of tricks for efficient text classification. *arXiv preprint arXiv:1607.01759*, 2016.

[15] David Leonhardt and Stuart A Thompson. Trumps lies. *New York Times*, 21, 2017.

[16] Rada Mihalcea and Carlo Strapparava. The lie detector: Explorations in the automatic recognition of deceptive language. In *Proceedings of the ACL-IJCNLP 2009 Conference Short Papers*, pages 309–312. Association for Computational Linguistics, 2009.

[17] Jeffrey Pennington, Richard Socher, and Christopher Manning. Glove: Global vectors for word representation. In *Proceedings of the 2014 conference on empirical methods in natural language processing (EMNLP)*, pages 1532–1543, 2014.

[18] Hannah Rashkin, Eunsol Choi, Jin Yea Jang, Svitlana Volkova, and Yejin Choi. Truth of varying shades: Analyzing language in fake news and political fact-checking. In *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing*, pages 2931–2937, 2017.

[19] Victoria Rubin, Niall Conroy, Yimin Chen, and Sarah Cornwell. Fake news or truth? using satirical cues to detect potentially misleading news. In *Proceedings of the Second Workshop on Computational Approaches to Deception Detection*, pages 7–17, 2016.

[20] Victoria L Rubin, Yimin Chen, and Niall J Conroy. Deception detection for news: three types of fakes. In *Proceedings of the 78th ASIS&T Annual Meeting: Information Science with Impact: Research in and for the Community*, page 83. American Society for Information Science, 2015.

[21] Natali Ruchansky, Sungyong Seo, and Yan Liu. Csi: A hybrid deep model for fake news detection. In *Proceedings of the 2017 ACM on Conference on Information and Knowledge Management*, pages 797–806. ACM, 2017.

[22] Kai Shu, Amy Sliva, Suhang Wang, Jiliang Tang, and Huan Liu. Fake news detection on social media: A data mining perspective. *ACM SIGKDD Explorations Newsletter*, 19(1):22–36, 2017.

[23] Sneha Singhania, Nigel Fernandez, and Shrisha Rao. 3han: A deep neural network for fake news detection. In *International Conference on Neural Information Processing*, pages 572–581. Springer, 2017.

[24] Eugenio Tacchini, Gabriele Ballarin, Marco L Della Vedova, Stefano Moret, and Luca de Alfaro. Some like it hoax: Automated fake news detection in social networks. *arXiv preprint arXiv:1704.07506*, 2017.

[25] James Thorne, Mingjie Chen, Giorgos Myrianthous, Jiashu Pu, Xiaoxuan Wang, and Andreas Vlachos. Fake news stance detection using stacked ensemble of classifiers. In *Proceedings of the 2017 EMNLP Workshop: Natural Language Processing meets Journalism*, pages 80–83, 2017.

[26] William Yang Wang. " liar, liar pants on fire": A new benchmark dataset for fake news detection. *arXiv preprint arXiv:1705.00648*, 2017.

[27] Xingyou Wang, Weijie Jiang, and Zhiyong Luo. Combination of convolutional and recurrent neural network for sentiment analysis of short texts. In *Proceedings of COLING 2016, the 26th International Conference on Computational Linguistics: Technical Papers*, pages 2428–2437, 2016.

[28] Liang Wu, Jundong Li, Xia Hu, and Huan Liu. Gleaning wisdom from the past: Early detection of emerging rumors in social media. In *Proceedings of the 2017 SIAM International Conference on Data Mining*, pages 99–107. SIAM, 2017.

[29] Liang Wu and Huan Liu. Tracing fake-news footprints: Characterizing social media messages by how they propagate. In *Proceedings of the Eleventh ACM International Conference on Web Search and Data Mining*, pages 637–645. ACM, 2018.

[30] Zichao Yang, Diyi Yang, Chris Dyer, Xiaodong He, Alex Smola, and Eduard Hovy. Hierarchical attention networks for document classification. In *Proceedings of the 2016 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 1480–1489, 2016.

[31] Xiang Zhang, Junbo Zhao, and Yann LeCun. Character-level convolutional networks for text classification. In *Advances in neural information processing systems*, pages 649–657, 2015.

[32] Chunting Zhou, Chonglin Sun, Zhiyuan Liu, and Francis Lau. A c-lstm neural network for text classification. *arXiv preprint arXiv:1511.08630*, 2015.