STEM Fellowship Big Data Challenge 2021

A Hybrid Deep Learning Architecture to Detect Fake COVID-19 News on Social Media

Jia Yi Ong[1]

[1]University of Toronto

June 1, 2021

Abstract

False or inaccurate information that circulates social media platforms can instill false beliefs and encourage ill-informed decisions. This can have adverse effects on the society as a whole, especially during the pandemic period. Thus, it is imperative to utilize the latest techniques in Artificial Intelligence and Analytics to curb the spread of misinformation. I employ a sequential deep learning network with Word Embeddings-based feature extraction to detect false and inaccurate social media content such as Facebook posts and Tweets. My hybrid architecture leverages the strength of Convolutional Neural Nets (CNNs) at feature learning and the insensitivity to gap length of Long Short-Term Memory (LSTM) models. I train my model on a COVID-19 social media data set to learn the semantic structures within various lengths of Tweets or other texts on social media for high-performance detection of fake news. My tuned model yields a test accuracy of 86.88% and a True Positive rate of 88.76%. Further refinement to the hybrid architecture and parameter tuning should yield higher performance closer to that required of a real-time fake news detector.

Keywords
Deep Learning, Hybrid Architecture, Infodemic, LSTM, Word-Embedding

1 Introduction

Being misled by inaccurate or false news during the COVID-19 pandemic can prevent individuals from seeking out and understanding the truth on how to properly protect against or treat the infection [1]. Thus, the spread of misinformation can influence a population's level of compliance with health regulations and may consequently impede efforts to attenuate the effects of the pandemic [2]. The circulation of misinformation about COVID-19 has been shown to be more rampant on social media platforms such as Twitter, than on traditional media [2]. Thus, there is a need to systematically and accurately identify and remove false claims and news on social media platforms before they cause real harm to society.

Deep learning techniques offer an advantage over conventional feature-based machine learning methods in addressing the infodemic due to their ability of automatic feature learning. Thus, the research focus is increasingly directed to applying various network architectures to learn representations from the contents of news and social media articles in order to better detect fake news. Many techniques incorporate embeddings-based feature extraction because pre-trained embeddings are publicly available and are more feasible than training one from scratch. Using a long short-term memory (LSTM) model with Global Vectors for Word Representation (GloVe) word embeddings, a 84.1% fake news classification accuracy can be achieved (Sharma et al., 2020) [3]. Good accuracy of LSTMs can be attributed to their insensitivity to gap length, which means that it can learn language structures present in lengthy sentences or paragraphs making LSTM a good candidate for fake news detection. Ajao et al. (2018) [4] compared three model variants on classifying tweets containing rumours: LSTM, LSTM with dropout regularization, and a LSTM-convolutional neural network (CNN) hybrid; with the plain-vanilla LSTM obtaining the best accuracy of 82%. Kaliyar et al. (2021) [5] used a BERT-based deep learning approach, a bidirectional approach

incorporating parallel 1-dimensional CNN blocks with max-pooling layers, and achieved an accuracy of 98.9% on fake news classification.

I aim to implement a unique deep learning architecture with embeddings-based feature extraction, that includes elements from the mentioned works (bidirectional, CNN-RNN hybrid, drop out regularization etc.). I will train it on a social media COVID-19 Fake News Dataset that is curated by Patwa et al. (2021) [6], and strive to exceed the performance of their benchmark models, which are Logistic Regression, Support Vector Machine (SVM), Decision Tree, and Gradient Boost classifiers (with the SVM achieving best accuracy of 93.32%). Applications of my trained model could range from real-time flagging of suspicious posts to moderators to crime fighting against malignant disinformation. My results will also demonstrate the advantages of using deep learning over classical machine learning methods and showcase the optimal hyperparameters associated with my model architecture for false claims detection on social media.

2 Materials & Methods

2.1 Dataset

The dataset was curated by Patwa et al. [7]. Each row of the dataset contains a COVID-19 related tweet, post, or article that is labelled either 'Real' or 'Fake'. The 'Real' category comprises only of tweets, while data of the 'Fake' category are drawn from various social media platform such as Facebook, Instagram and other popular media sites. Only textual English content is included. Determination of 'Fake' data is done using fact verification sites (e.g. PolitiFact, Snopes, Boomlive). 'Real' data is only drawn from their list of 14 credible sources (including the World Health Organization and Centers for Disease Control) and processed by a human to determine if useful content is present.

I subset the original dataset of 10,700 rows by excluding the content that exceed 54 words (as separated by spaces). This is done to limit the the dimension of input to my model, as I use zero-padding to ensure the sentence inputs are of the same length. The resulting subset has 5576 of the 'Real' category and 5029 of the 'Fake' category. The categories are converted to a binary column (1 for 'Fake' and 0 for 'Real'). I employ a train-test split of 90-10 (subsequent validation data will be split from the training set due to the implementation of the learning packages).

To reduce noise in the data, each tweet/post is removed of any hyperlinks and English stop words (as given in the Gensim stopword library). The 50-dimensional GloVe pre-trained word embedding for Twitter [8] is used for feature extraction [1].

2.2 Deep Learning Architecture

I implemented a stacked bi-directional LSTM architecture that incorporates CNN and drop-out regularization. The details are shown below.

Table 1: Hybrid Architecture

Layer	Output Size	Parameters
Embedding	(?, 54, 50)	59675700
Conv1D	(?, 52, 32)	4832
Maxpool	(?, 26, 32)	0
Bi-LSTM	(?, 26, 64)	16640
Bi-LSTM	(?, 26, 64)	24832
Dropout	(?, 26, 64)	0
Bi-LSTM	(?, 26, 64)	24832
Bi-LSTM	(?, 64)	24832
Dropout	(?, 64)	0
Dense	(?, 512)	33280
Dense	(?, 320)	164160
Dense	(?, 1)	321
Total Params:		59,969,429
Trainable Params:		293,729
Non-Trainable Params:		59,675,700

I first initialize a sequential model with Keras. Then, a function is implemented to extract from the Twitter GloVe embedding to form the embedding matrix: its rows store the 50-dimensional vector representations of every word in the GloVe vocabulary. A Keras Embedding layer is initialized and its weights are set equal to the derived embedding matrix. The embedding layer is set fixed throughout training. The other layers are then added in their respective order as seen in Figure 1.

The embedding layer feeds into a 1-dimensional Convolutional layer to enhance the learning of semantic relationships between words

[1]The GloVe pre-trained word vectors are made available under the Public Domain Dedication and License v1.0 whose full text can be found at: http://opendatacommons.org/licenses/pddl/1.0/

of different syntactic forms. This layer's ReLU activation then feeds into a Max Pooling layer to lower the number of weights in the model. After the next few bidirectional LSTM layers and drop-out layers are the fully-connected dense layers. The final dense layer outputs one unit that is fed into the sigmoid activation for the final classification. The model minimizes binary cross-entropy loss.

2.3 Parameter Optimization

With reference to Table 1, the hyperparameters are the drop out rates, the dense layer unit sizes, and the learning rate for the Adam optimizer. Hyperparameter tuning is done with the random search tuner of the Keras Tuner library, with the learning rate sampled on a log scale. A total of 20 trials were run, with each trial corresponding to one randomly sampled point in the hyperparamter space. Each trial executes once with 10 epochs, with early stopping implemented. Thereafter, the set of parameters with the best tuning objective metric, validation accuracy, is selected. Validation accuracy is computed on 10% of the training set. The parameters are used to build a new model that is trained on 50 epochs to determine the optimal number of epochs. The performance statistics are then generated for the final model.

3 Results

The model tuning has yielded the following set of hyper-parameters that maximizes validation accuracy at 91.27%.

Table 2: Optimal Hyperparameters

Hyper-parameter	Tuned
Conv1D Activation	ReLU
drop-out rate 1	0.25
drop-out rate 1	0.10
Dense Layer 1 Units	512
Dense Layer 2 Units	320
Adam Learning Rate	0.006718

Using these sets of hyper-parameters, the model was trained again on 50 epochs to determine the optimal epoch number. In Figure 1, there is a large divergence between training and validation accuracy. Training accuracy converges towards near 100% (with the exception

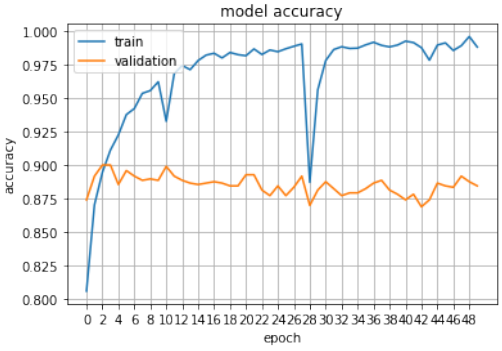

Figure 1: Train-Validation Accuracy

of abnormal behaviour at around 26-30 epoch). On the other hand, validation accuracy seems to be steadily decreasing. The optimal epoch is deemed to be 3.

The final model is trained with the optimal hyper-parameters and 3 epochs. The performance of the final model is shown below.

Table 3: Final Tuned Model Performance

Metric	Value
Validation Accuracy	90.12%
Test Accuracy	86.88%
AUC Score	86.93%

Below is the Confusion Matrix of the classifier on the test set of 1067 data points:

		Truth		
		Fake	Real	Total
Predict	Fake	474	80	554
	Real	60	453	513
	Total	534	533	1067

The following measures are derived from the Confusion Matrix (keeping in mind 'Fake' is the positive class):

Table 4: Performance Measures

Metric	Value
True Positive Rate	$474/534 = 88.76\%$
True Negative Rate	$453/533 = 84.99\%$
False Positive Rate	$80/533 = 15.00\%$
False Negative Rate	$60/534 = 11.24\%$

4 Discussion

4.1 Strengths and Limitations

My model's performance as seen in Table 3 is adequate, with an AUC score of 86.93% indicating a good ability to differentiate between fake and real news. This demonstrates that this architecture is somewhat an improvement to the plain vanilla LSTM benchmark performance as shown by Ajao et al. (2018) with an accuracy of 82% [4]. However, it still has vast room for refinements before qualifying for deployment in real-time fake news detection. In Table 4, a false negative rate (mislabelling fake news as real) at 11.24% is only moderately acceptable when it comes to processing vast quantities of real-time tweets. From Table 3, my model's test accuracy at 86.88% also fails to exceed the benchmark performance (at 93.32%) that is set by the SVM algorithm of the dataset's curators Patwa et al. (2021) [6]. Moreover, Patwa et al. (2021) achieved that higher performance using the Term Frequency - Inverse Document Frequency method for feature extraction, which does not capture semantic relationships in language as effectively as word embeddings do. This may indicate that my model's discrepancy in performance is arising from other sources. Combined with the observation in Figure 1 that the validation accuracy clearly plateaued within a few epochs, it suggests that adjustments are needed for my deep learning architecture or for my tuning procedure.

The divergence of the training accuracy from the validation accuracy is a clear sign of over-fitting to the training data. Although I have employed drop-out regularization and early stopping, these devices does not seem to have curbed over-fitting. It could indicate that four-layers of bidirectional LSTM adds too much complexity to the architecture and that my training data size is too small for such complexity. These insights shed light on many potential future improvements.

Conclusions

The unique hybrid deep learning architecture that I have implemented does moderately well in detecting fake news on social media and exceeds some of the bench-mark performance in the literature. More data can be collected to train my model, and more work can be done to ensure similarity in the distribution of data in the train-validation and test sets. This may be done by controlling for the variation in lengths of tweets for example. The number of trials during hyperparameter tuning can be increased so that more combination of parameters are tested. With enough computational resource, a grid search rather than a random search may also provide a more comprehensive tuning. I still have a lot of room to experiment with new hybrid architectures by adding or removing layers and other elements. Lastly, there are still a range of deep learning models that have not been bench-marked on COVID-19 social media data. More bench-marking work would shed light on the tractability of fake news using deep learning.

Acknowledgements

I would like to extend my gratitude towards the amazing team at STEM Fellowship, for their effort in hosting the workshops and for organizing such a great opportunity for us to develop our analytics skills.

References

[1] Hye Kyung Kim, Jisoo Ahn, Lucy Atkinson, and Lee Ann Kahlor. Effects of covid-19 misinformation on information seeking, avoidance, and processing: A multicountry comparative study. *Science Communication*, 22(5):586–615, 2020.

[2] Aengus Bridgman, Eric Merkley, Peter John Loewen, Taylor Own, Derek Ruths, Lisa Teichmann, and Oleg Zhilin. The causes and consequences of covid-19 misperceptions: Understanding the role of news and social media. *The Harvard Kennedy School Misinformation Review*, 1, 2020.

[3] Rishibha Sharma, Vidhi Agarwal, Sushma Sharma, and Meenakshi S. Arya. An lstm-based fake news detection system using word embeddings-based feature extraction. *ICT Analysis and Applications. Lecture Notes in Networks and Systems*, 154:247–255, 2021.

[4] Oluwaseun Ajao, Deepayan Bhowmik, and Shahrzad Zargari. Fake news identification on twitter with hybrid cnn and rnn models. *SMSociety '18 Proceedings of the 9th International Conference on Social Media and Society. ACM International Conference Proceeding Series*, pages 226–230, 2018.

[5] Rohit Kumar Kaliyar, Anurag Goswami, and Pratik Narang. Fakebert: Fake news detection in social media with a bert-based deep

learning approach. *Multimed Tools Appl*, 80:11765–11788, 2021.

[6] Parth Patwa, Shivam Sharma, Srinivas Pykl, Vineeth Guptha, Gitanjali Kumari, Md Shad Akhtar, Asif Ekbal, Amitava Das, and Tanmoy Chakraborty. Fighting an infodemic: Covid-19 fake news dataset. pages 21–29, 2021.

[7] Parth Patwa, Shivam Sharma, Srinivas PYKL, Vineeth Guptha, Gitanjali Kumari, Md Shad Akhtar, Asif Ekbal, Amitava Das, and Tanmoy Chakraborty. Fighting an infodemic: Covid-19 fake news dataset, 2020.

[8] Jeffrey Pennington, Richard Socher, and Christopher D. Manning. Glove: Global vectors for word representation. pages 1532–1543, 2014.