Fine-Grained Fake News Detection System

Harsh Goyal
harsh20562@iiitd.ac.in

Ayush Sharma
ayush20042@iiitd.ac.in

Anupam Narayan
anupam20030@iiitd.ac.in

Shivam Jindal
shivam20125@iiitd.ac.in

Anas Ahmad
anas20023@iiitd.ac.in

March 27, 2023

1 Motivation

False or misleading information presented as news is known as fake news. Most of the time, fake news aims to hurt someone or something's reputation or make money from advertising. Of late, fake news has been in the spotlight of mainstream journalism and the general public because it can affect a country's political scenario. In this project, we attempt to detect the authenticity of specifically political news. Social media is the primary channel for disseminating such content, though it occasionally becomes mainstream media. Because fake news can significantly impact an election's political outcome, it is becoming increasingly important to identify and classify it as such. The loose definition of ”fake news” presents the primary obstacle to resolving the problem. For instance, fake news can be broken down into subcategories: a statement known to be completely false, a speech that presents statistics as facts that have not been thoroughly investigated, or satirical writing. Our main aim of the project is to detect and classify fake news.

2 Problem Statement

Fake news is false or misleading information presented as news. Multiple strategies for fighting fake news are currently being actively researched for various types of fake news. Our task is determining whether a news statement/speech is fake or not given an input statement using Natural Language Processing and associated language models. We aim to design novel and hybrid models using pre-trained large language models to integrate the original news statement/speech with the metadata. Fake News is a text-classification task. Since the news statements are very short in length and the text from the speech is noisy and contains grammatical errors, the task becomes more complex and exciting.

3 Literature review

3.1 Liar, Liar Pants on Fire

The author proposed a hybrid Convolution Neural Networks Framework for integrating text and metadata. Hybrid CNN consists of two parallel Convolution Layers in which the input to the first Convolution Layer is the word embeddings for the given statement, followed by the Max-Pooling. In contrast, the input to the second convolution layer is the metadata like speaker name, subjects of the speech,/statement, speaker's party, etc., followed by a Bidirectional-LSTM layer. The output of both layers is then concatenated and fed into a fully-connected layer with softmax on the output layer.

3.2 A Retrospective Analysis of the Fake News Challenge Stance Detection Task

The author uses a fake news challenge dataset which is a 4-class classification task. The author used two stacked LSTMs with 50-dimensional GloVe word embeddings as input and a three-layered neural network with 600 neurons each. The output from a three-layered neural network is fed into the output layer consisting of 4 neurons to estimate the class-wise probabilities to estimate to which class it belongs.

3.3 Exploring Text-transformers in AAAI 2021 Shared Task: COVID-19 Fake News Detection in English

The author proposed two solutions to the problem. The first solution uses RNNs (also called Bidirectional-LSTMs) as the LSTMs are based on previous text information. In contrast, the RNNs are based on both previous and later text information, which helps get a better sentence context. The second solution uses 3 different techniques - a Five-Fold Single-model ensemble, a Five-Fold Five-model ensemble, and a Pseudo Label algorithm. In the Five-Fold Single-model ensemble, the author used the same transformer-based pre-trained models (like BERT, Roberta, etc.) on all the five-folds of the dataset. The author used different pre-trained models for fine-tuning each fold in the Five-Fold Five-model ensemble. In the Pseudo Label algorithm, the author proposed that if the test data sample is classified to some class with a probability greater than 0.95, then the author proposed to use that test data sample as the training data sample for future test samples.

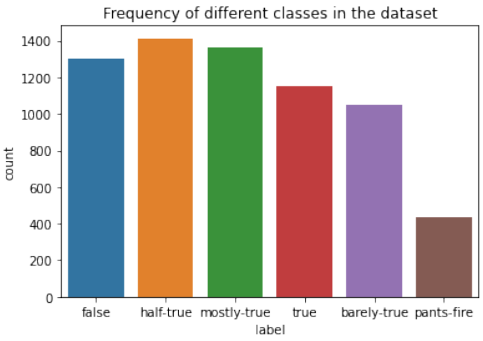
3.4 On the Benefit of Combining Neural, Statistical and External Features for Fake News Identification

The author proposed how using three different word embeddings in parallel and concatenating them can benefit fake news detection. The author combined neural embeddings, statistical features, and external features to a similar type of model (with a little difference in activation functions) in parallel and concatenated them to get the combined features. The author defined neural embeddings as the skip-thought vectors which encode the given sentences to vector embedding of length 4800. The statistical features are defined as the vectors obtained from the text using Bag-of-words(BOW), TF-IDF, and n-grams techniques. The external features include heuristics such as the similarity between headline-body pairs, the number of similar words in the headline and the body, etc. All these three types of features are combined using pre-trained models and then fed into dense layers to predict the correct label for the given input sentence.

4 Dataset Details

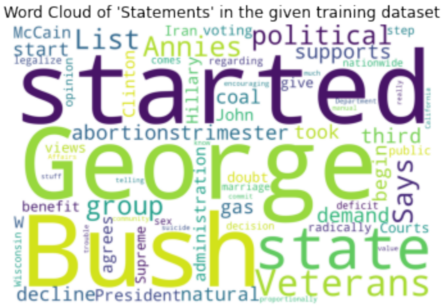
Dataset Statistics	
Training set size	10,269
Validation set size	1,284
Testing set size	1,283
Avg. statement length(tokens)	17.9

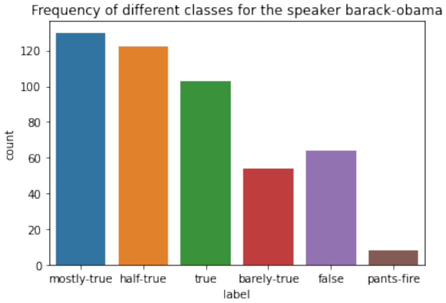
The 6-classes of the dataset are defined as:-
(These definitions were taken from PolitiFact's "truth-o-meter" methodology page).
1. **true** – The statement is accurate, and there's nothing significant missing.
2. **mostly-true** – The statement is accurate but needs clarification or additional information.
3. **half-true** – The statement is partially accurate but leaves out important details or takes things out of context.
4. **barely-true** – The statement contains an element of truth but ignores critical facts that would give a different impression.
5. **false** – The statement is not accurate.
6. **pants-fire** – The statement is inaccurate and makes a ridiculous claim. a.k.a. "Liar, Liar, Pants on Fire!"



4.1 Updated Dataset

As the training dataset is minimal (after removing NaN rows from the dataset, the amount of data decreased further), therefore to generalize our models better, we merged the validation data in the training data and tested our models on the testing data.





5 Models and Methodology

5.1 TFIDF Vectorizer on text statements and ML Models

We computed the TFIDF matrix for the text statements in the training data. For training purposes, we gave TFIDF matrix and the corresponding labels as input to the classification model and then tested the model for the testing data. We tried this technique on various ML models like Logistic Regression, Naive Bayes, Random Forest, etc. We also tried hyperparameter tuning for all these ML models.

5.2 TFIDF Vectorizer on POS taggings of text statements and ML models

In this technique, we computed the TFIDF matrix for the POS taggings for the tokens in the text statements in the training data. For training the classifier, we gave the TFIDF matrix and the corresponding labels as input to the classifier and tested the model for the testing data.

5.3 Non-Contextual Word Embeddings as input

There are various Non-Contextual word embeddings like Word2vec, GloVe, FastText, etc. We used GloVe embeddings for our models because, unlike Word2vec, GloVe does not rely on local context information of words but incorporates global statistics (word co-occurrence) to obtain word vectors. In this technique, we gave the glove embeddings for the text statements in the training data as input to the classifier and tested the model for the testing data. We tried different ML models in this technique.

5.4 Contextual Word Embeddings as input

Contextual word embeddings assign each word a vector representation based on its context in the sentence. They can be used to learn sequence-level semantics by considering the sequence of all words in the document. Non-contextual word embeddings form the exact vector representation for the polysemous words as it does not consider the context of the sentence. In contrast, contextual word embeddings can assign different vector representations for the same word depending on the context of that word in the sentence. In this technique, we used a sentence-transformer-based model **all-MiniLM-L6-v2** to generate the contextual word embeddings of the text sentences in the given data. We gave the embeddings as input to various ML models.

6 Evaluation Metric

For the baseline results, earlier, we used weighted F1-score as our evaluation metric. But, after doing some research and based on our understanding of the concepts, we thought that the weighted F1-score is not a good metric for the given dataset. According to our understanding, the weighted F1-score is generally used when the number of samples corresponding to each class in the dataset is similar, as the weighted F1-score gives more weightage to more regular classes as compared to less frequent classes while computing the weighted F1-score. So, we decided to change our evaluation metric from weighted F1-score to **macro F1-score** as the macro F1-score gives equal weightage to all the classes, unlike the weighted F1-score.

7 Experimental Results

Models	Macro F1-Score
CV + MNB	0.17
TF-IDF + LR	0.21
CV + LR	0.22
TF-IDF + SVM	0.22
CV + DT	0.22
TF-IDF + RF	0.21
TF-IDF(POS) + LR	0.21
TF-IDF(POS) + SVM	0.18
TF-IDF(POS) + DT	0.18
TF-IDF(POS) + MLP	0.19
Contextual Embedding + MLP	0.20
Contextual Embedding + LR	0.14
Non-contextual embedding + MLP	0.17

The results presented show the performance of various machine learning models on the given dataset. The evaluation metric used is the macro F1 score.

Looking at the results, the highest-performing model is the CV (Count Vectorizer) + LR (logistic regression), TF-IDF + SVM, and CV + DT (Decision Trees) models with a macro F1 score of 0.22. (TF-IDF stands for Term Frequency-Inverse Document Frequency). This indicates that this model achieved the best balance between precision and recall for both the positive and negative classes.

The CV + MNB (Multinomial naive Bayes) and Non-contextual + MLP models have the lowest performance with a macro F1 score of 0.17. Since Non-contextual word embeddings do not consider the context of the word while assigning vector representation to the word and consider the global statistics of the word, they are less effective than contextual word embedding. This indicates that the model's accuracy in predicting both classes is low.

The other models, including TFIDF + LR (logistic regression), Contextual Embeddings + MLP (Multi-layer Perceptron), etc., have relatively similar performances with macro F1 scores ranging from 0.18 to 0.21

8 Future Work

We planned to dive deep into deep learning-based models like CNNs, LSTMs, and some pre-trained large language models (LLMs) like BERT for future work. We planned to develop some hybrid CNN and LSTM models, which along with word embeddings, take metadata as input as in recent research in various domains, it can be seen that the metadata helped in improving the performance of the various models. We have also thought of implementing some novel architectures using the pre-trained LLMs. Using our novel architecture, we also aim to beat state-of-the-art for the given dataset.

9 References

[1] William Yang Wang. 2017. "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), pages 422–426, Vancouver, Canada. Association for Computational Linguistics.

[2] Andreas Hanselowski, Avinesh PVS, Benjamin Schiller, Felix Caspelherr, Debanjan Chaudhuri, Christian M. Meyer, and Iryna Gurevych. 2018. A Retrospective Analysis of the Fake News Challenge Stance-Detection Task. In Proceedings of the 27th International Conference on Computational Linguistics, pages 1859–1874, Santa Fe, New Mexico, USA. Association for Computational Linguistics.

[3] Li, X., Xia, Y., Long, X., Li, Z., and Li, S. (2021). Exploring Text-transformers in AAAI 2021 Shared Task: COVID-19 Fake News Detection in English. ArXiv. https://doi.org/10.48550/arXiv.2101.02359

[4] Bhatt, G., Sharma, A., Sharma, S., Nagpal, A., Raman, B., and Mittal, A. (2017). On the Benefit of Combining Neural, Statistical and External Features for Fake News Identification. ArXiv. https://doi.org/10.48550/arXiv.1712.03935

10 Contributions

Motivation - Anas Ahmad, Shivam Jindal
Problem Statement - Anupam Narayan, Ayush Sharma
Literature Review - Harsh Goyal
Models and Evaluations - Harsh Goyal, Ayush Sharma, Shivam Jindal, Anupam Narayan