Fake News Stance Detection

Victor Wang
University of California, Berkeley
School of Information
victor.wang@ischool.berkeley.edu

Tina Huang
University of California, Berkeley
School of Information
tinahuang@berkeley.edu

Abstract

Misinformation in the media is a widespread issue that poses threats to the public well-being; the repercussions from informational deceit range from annoyances, and can extend to toward the loss of money, time and even health. The goal of this project is to counter this trend by developing a machine learning-based classifier that identifies stances between two bodies of text to mitigate the issue. Working with the Fake News Challenge organization dataset, we developed several neural network-based models organized around recurrent network models as well as the use of advanced transformers models, such as the Bidirectional Encoder Representations from Transformers (BERT). Stance detection is envisioned as a mitigation means for novel data, and seeks to classify the news headline and body texts into stances of “agrees,” “disagrees,” “discusses,” and “unrelated.” We ultimately found that a model that incorporated BERT was superior to a LSTM-based bidirectional model and provided the best performance or F1 score.

1 Introduction

In today’s digital era, the issue of fake news, as defined by the Fake News Challenge (FNC) and the New York Times as “a made-up story with an intention to deceive,” is widespread and pervasive in effect. Fake news poses real threats to virtually all organizations and individuals. No one is exempt from these issues, but it is our contention that with the appropriate tools and safeguard mechanisms, all can seek to protect themselves.

For our W266 final project, our team strives to address the problems posed by solving part of the issue through stance detection between the news article Headline and Body. This approach divides the overall problem as the relationship between the two bodies of text in the following ways.

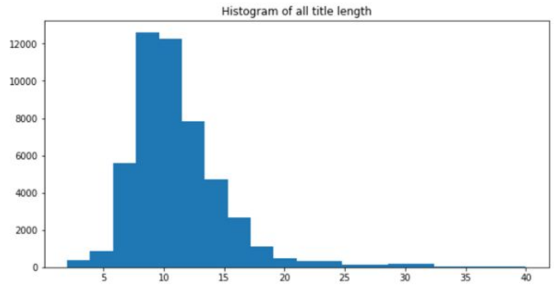
1. The article text agrees with the headline.
2. The article text disagrees with the headline.
3. The article text is a discussion of the headline, without taking a position on it.
4. The article text is unrelated to the headline.

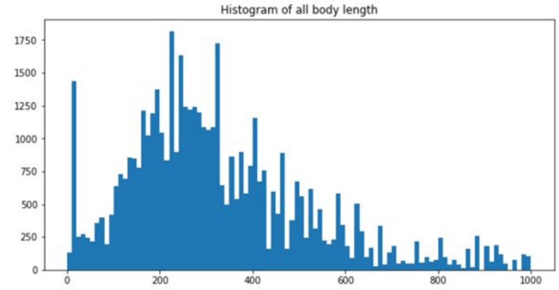
It is our position that classifying stances in between text supplies versatility in novel data and articles and can generalize well for the additional applications structured in sequential statement and responses format. This approach provides strong gains towards formats such as twitter data, product reviews, forums comments, as well as direct question and answer content.

2 Dataset Overview

Our dataset is sourced from the FNC organization and is available on their Github page. The training set consists of 49,972 labeled stances, which build upon two relational databases consisting of an article headline and article body. To begin, the first data processing step is to combine the databases using the relational key of a body identifier and the associated stance. In several cases, a singular headline was matched with multiple article bodies and subsequently labeled with the appropriate stance.

Next, we analyzed the data content for structure, focusing primarily on body length. The following figures capture the word length of the dataset for the article headlines and the article bodies from the FNC. Because BERT base has an input token limit of 512 and larger token size will result in longer training time, we experimented with title length options [10, 15] and article body length options [40, 100,185] based on the histogram below.





We experimented with 3 methods of shortening the text to the target article body length n:

1. Keep the first n tokens.
2. Randomly select n tokens to keep.
3. Use an extractive method to summarize text to n tokens.

We found that randomly selecting n tokens for use in the model to be the best performing method. For the provided stance labels, we converted the stance labels into one hot encoding for processing.

3 Models

Our model development approach was to build a baseline model to experiment with a wide range of deep learning and NLP techniques presented in the w266 class.

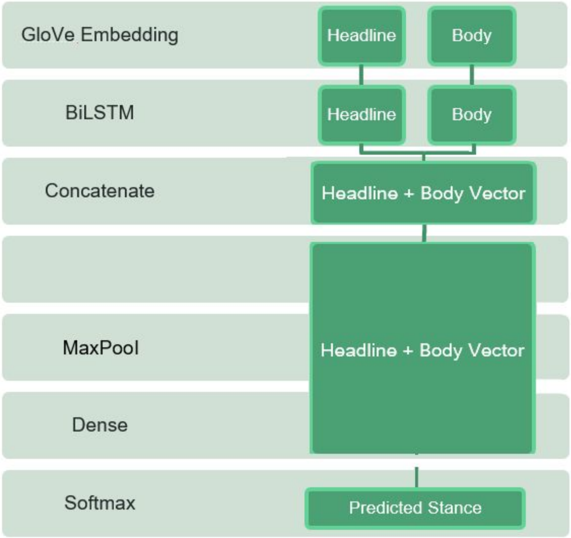
The following concepts were incorporated

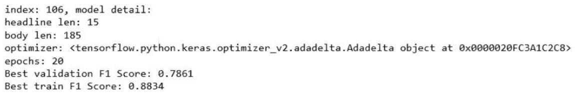
- Word Embeddings, such as GloVe
- Optimizers including momentum, such as ADAM
- Recurrent Models, such as an LSTM layer
- Deep Learning approaches, such as a CNN layer

3.1 Baseline approach

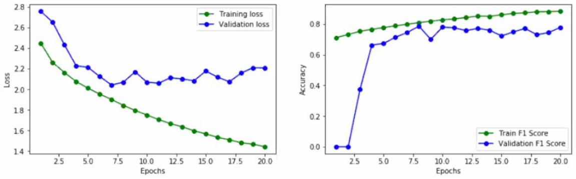
For exploratory data analysis, our baseline model processes the FNC data by forming two concurrent pipelines for the Headline and Body. Both bodies of text go through a character and word level data cleansing, with the elimination of special characters, and select stop words. Next, the text is encoded and padded to 50 for conversion into word embeddings as presented by GloVe.

The embeddings are pushed through a recurrent layer with LSTM and a reduction of dimensions to a 32 output vector. Choice of a single pass, along with a bidirectional LSTM layer was used to compare the differences. The outputs from the Headline and Body from the LSTM are then concatenated and normalization techniques, as well as dropout are used. Finally the data is pushed through a 1 dimensional CNN, as well as a max pooling layer, before presented to a fully connected layer with 4 outputs and a softmax. The loss function used is a categorical cross entropy function to account for the multi-class stance labels that are one-hot encoded. Diagram of the baseline model is presented in figure below





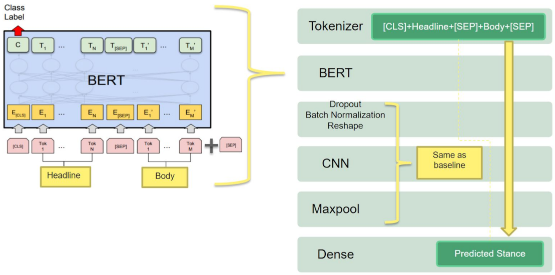
```
index: 106, model detail:
headline len: 15
body len: 185
optimizer: <tensorflow.python.keras.optimizer_v2.adadelta.Adadelta object at 0x0000020FC3A1C2C8>
epochs: 20
Best validation F1 Score: 0.7861
Best train F1 Score: 0.8834
```

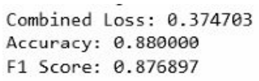


3.2 Transformers

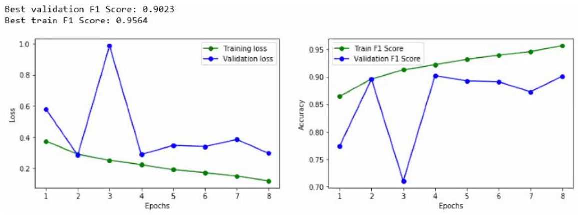
The next exploration step covers transformer architectures using the BERT model. Our scenario selected the BERT base model (L=12, H=768, A=12) as well as the uncased pre-trained model for classification. The BERT model provides a custom tokenizer, that includes separating words into word parts(eg running into [run] + [##ing]), as well as use of the [CLS] and [SEP] tokens necessary for interfacing to the model. To take advantage of the BERT sentence pair classification, our approach tokenized both the headline and body texts, and allocated a fixed number of tokens for each text by padding to the determined token limit. The output of both tokenizers were then concatenated together, along with the [SEP] token as a bridge.

In comparison for our baseline model, the back layer after the BERT layer, we obtained the output vector of 768 and performed the same normalization and CNN layers for the baseline. The model design is presented in figure TBD





```
Combined Loss: 0.374703
Accuracy: 0.880000
F1 Score: 0.876897
```



```
Best validation F1 Score: 0.9023
Best train F1 Score: 0.9564
```

4 Experimental process

For the FNC dataset and model, we selected the following hyperparameters for tuning

- Learning Rate
- Weight Decay
- Token Size
- Token Selection (Text Summarization, Random Drop, and Truncation)
- Optimizer (adagrad, adadelta, sgd, adam, RMSprop)

For the FNC dataset, the 50K data proved to factor into significant training times for fine-tuning the BERT model. The size of the body data set showed the significant word size was present that would exceed the limits for BERT token choice.

5 Results

Model	F1 Score	Time per sample
GloVe + BiLSTM + CNN	0.7480	3ms
GloVe + BiLSTM + Drop + CNN	0.7630	3ms
BERT (15H, 100B*) + CNN	0.8769	10ms
BERT (30H, 30B) + CNN	0.7291	6ms
BERT (20H, 80B) + CNN	0.7314	9ms
BERT (20H, 180B) + CNN	0.7340	6ms
BERT (20H, 280B) + CNN	0.7340	31ms

* H is token length for title text, B is token length for article body text.

For scoring metrics, we made use of the F1 Score, which is a harmonic mean of precision and recall. The optimal F1 Score for our models was the BERT at 0.8679.

6 Conclusion

For our stance detection models, we found the strongest F1 scoring results included the BERT architecture which performed notably better over the baseline models using GloVe embeddings and recurrent LSTM layers. As our intent was to examine the effects of Transformer architectures, we nevertheless designed commonality between our baseline models through the back layers and the approaches for token truncation and text summarization. In a sense, the BERT architecture with its bidirectional encoder decoder sequencing layers and conditional attention vectors replaced and enhanced the performance for GloVe and bidirectional CNNs by approximately 10% improvement in accuracy and F1 score.

Furthermore, for the sentence pair classification from BERT, our team felt there was still opportunity and ample space to continue to fine tune and identify additional hyperparameter optimizations, such as incorporating the BERT large model and formulate more optimal token selection along with larger token limits. During the training processes, we observed notable increases in training times and computing requirements for utilizing BERT, especially for larger token sizes and for larger datasets. While none of these limits are insurmountable, they nevertheless hinder some of the exploration space and tuning direction.

Going forward, we expect that a good portion of the research workflow, in order to appropriately harness the power of BERT and transformer models, will be spent on data analysis and optimization and efficiency strategies. This means the selection of efficient tokenizer approaches, such as use of a data analysis model to pre-select an optimal subset of words from a body text, the use of abstractive or extractive text summarization, or even incorporating syntactical structure parsing for prioritizing text selection would achieve better results. This is to deal with the two limits within the BERT model, first an explicit token limit, and a second implicit limit based on prolonged training times.

Overall, we posit the direction for state of the art NLP techniques will be built around transformer architectures such as BERT, or OpenAI GPT2. BERT has performed well for general tasking and the pre-trained models provide an advanced initial generalized foundation that allows for high accuracy fine-tuning to various tasks. Our team next steps will be to continue to update our model parameters as well as apply transformer based models for stance detection in sequential statement and responses formatted datasets.

References

Pennington J, Socher R, Manning C. (2014). GloVe: Global Vectors for Word Representation, Association for Computational Linguistics, Oct, 1532–1543. doi:10.3115/v1/D14-1162. Retrieved from *https://www.aclweb.org/anthology/D14-1162/*

Devlin J, Chang M.W, Lee K, Toutanova K. (2019) BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding, Association for Computational Linguistics, May,4171–4186, doi: 10.18653/v1/N19-1423. Retrieved from

https://www.aclweb.org/anthology/N19-1423/

Thorne J, Chen M, Myrianthous G, Pu J, Wang X, Vlachos A. (2017). Fake news stance detection using stacked ensemble of classifiers. Association for Computational Linguistics, Sept, 80-83. doi:10.18653/v1/W17-4214.

Retrieved from https://www.aclweb.org/anthology/W17-4214/

Vaibhav V, Mandyam R, Hovy E. (2019). Do Sentence Interactions Matter? Leveraging Sentence Level Representations for Fake News Classification. *Association for Computational Linguistics, Nov*, 134-139. doi:10.18653/v1/D19-5316. Retrieved from https://www.aclweb.org/anthology/D19-5316/

Vlad G-A, Tanase M-A, Onose C, Cercl D-C. (2019). Sentence-Level Propaganda Detection in News Articles with Transfer Learning and BERT-BiLSTM-Capsule Model. *Association for Computational Linguistics, Nov*, 148-154. doi:10.18653/v1/D19-5022. Retrieved from https://www.aclweb.org/anthology/D19-5022/

Kotonya N, Toni F. (2019). Gradual Argumentation Evaluation for Stance Aggregation in Automated Fake News Detection. Association for Computational Linguistics, Aug, 156-166. doi:10.18653/v1/W19-4518. Retrieved from https://www.aclweb.org/anthology/W19-4518/

Yuan J, Zhao Y, Xu J, Qin B,(2019) Exploring Answer Stance Detection with Recurrent Conditional Attention, Proceedings of the AAAI Conference on Artificial Intelligence ,Vol 33 No 01: AAAI-19, IAAI-19, EAAI-20 , DOI: https://doi.org/10.1609/aaai.v33i01.33017426, Retrieved from https://www.aaai.org/ojs/index.php/AAAI/article/view/4732