Bias Detection in News Articles

Akshett Rai Jindal, Suyash Vardhan Mathur
{akshett.jindal, suyash.mathur}@research.iiit.ac.in

PROBLEM STATEMENT

Hyper-partisan bias detection in news articles

INTRODUCTION

Sentiment Analysis is a well-known problem in NLP, with various facets like bias detection, fake news detection, etc. Of these, bias detection is a very popular area in contemporary NLP. Such bias can be based on religion, race, gender, a political organization etc.
The goal of this project is to detect hyperpartisan bias in news articles. This kind of bias is one where news is reported in such a way that it strongly favours a particular position(mostly political), and would be in fierce disagreement with the opponents. Such kind of news often involves either sketching the truth or combining it with fake news in an attempt to create sensational content.

The problem taken in the project is a **binary classification problem**, where we would be classifying whether a particular article has a hyperpartisan bias or not.

EXAMPLES

- The article *The New York Times abandoned its integrity just to bash Donald Trump*[1] has a clear bias in favour of Donald Trump's right-wing politics and bashes left-favouring news articles.
- On the other hand, the article *CAPLAN: Will Bannonism—Not Trumpism—Shape The Republican Party's Future?*[2] is clearly biased against Trump, and has a hyperpartisan bias towards the left-wing.
- Another class of articles are the ones with no bias, like *Jennifer Aniston And Justin Theroux's Marriage In Trouble Because Of Chelsea Handler?*[3], which clearly doesn't have any hyperpartisan bias and is simply celebrity news.

RELATED WORK

The SemEval 2019 task 4 was *Hyperpartisan News Detection task*. This task received 42 submissions from various teams. Various approaches like using *n-gram* models like bag-of-words, word embeddings, stylometric features, HTML features like hyperlink targets and metadata features in the form of publication dates were used.

Among the usage of word embeddings, many used GloVE, fastText and Word2Vec embeddings. The winning submission was team Bertha-von-suttner's model that used sentence representations from average ELMo word embeddings coupled with Convolutional Neural Networks and Batch-Normalization for predicting hyper-partisan bias.

DATA

The dataset that used is the one used in the International Workshop on Semantic Evaluation (SemEval-2019)[4]. Specifically, we used the dataset for Task 4 which is **Hyperpartisan News Detection**. The dataset is tagged article-wise whether it follows a

[1] https://nypost.com/2016/10/11/the-new-york-times-abandoned-its-integrity-just-to-bash-donald-trump/

[2] https://www.thegatewaypundit.com/2017/10/caplan-will-bannonism-not-trumpism-shape-republican-partys-future/
[3] https://www.kdramastars.com/articles/99477/20151002/jennifer-aniston-justin-theroux.htm
[4] https://alt.qcri.org/semeval2019/index.php?id=tasks

hyperpartisan argumentation or not i.e. if it exhibits bind, prejudice, or unreasoning allegiance to one party, faction, cause or person.

For training and testing, the dataset was split in the ratio 8:2 into training and testing sets. An example from the dataset:

```
<article hyperpartisan="true" id="0000012"
labeled-by="article"
url="https://www.circa.com/story/2017/09/13/
action-sports/jemele-hill-trump-tweets-espn-
distances-itself-from-sportscenter-anchors-c
omments"/>
```

In addition to the above dataset, we also made use of the **IMDB Subjectivity Dataset**[5] for the subjective/objective sentence detection task. This dataset contains 5000 subjective and 5000 objective processed sentences, which were then combined and split in the training-testing ratio of 8:2 to be used for training of the sentence subjectivity classifier .

EXPERIMENT

Baseline Model

For our baseline model, we tried to replicate the results from Bertha-von-suttner's paper by using Convolutional Neural Networks.

The model loads those articles which were manually annotated for hyper-partisan manually and not the ones which were semi-automated by the publisher. This was because the semi-automated annotation dataset worsened the performance of the model.

Now, for converting the articles into inputs for CNN layers, we converted the article into a tensor of sentence embeddings. These sentence embeddings were obtained by averaging the ELMo embeddings of all the words of the sentence. The word embeddings were generated by using Allen NLP's pretrained ELMo model which was trained on 1 billion words. ELMo embeddings are contextual in nature, and so, are superior to other embeddings like GLoVE, FastText, etc.

The CNN model consisted of 5 convolutional layers, each followed by non-linear RELU activation function. Batch Normalization was used for reducing the internal co-variant shift in the neural networks. This process involved normalizing the input distribution by subtracting the batch mean and dividing by the batch standard deviation, so that the ranges of input distribution between each layer stay the same. This allows the model to have a higher learning rate, so that the training speed is accelerated. It also reduces overfitting by decreasing the dependence of weight initialization between each layer.

Then max-pooling was applied on the output of the batch-normalization layers, the outputs of which are combined to form the input to a fully connected layer which maps the input to one value. Sigmoid activation function is then applied to use the output for Binary Classification. Since the task is that of classifying articles as 0 or 1, we make use of Binary Cross Entropy loss to update the gradients. Adam optimizer is used (for being compact and fast in nature) to speed-up the training process.

For computing storage and speed reasons, we make use of the first 200 sentences from each article, taking 66 words from each sentence, which was the maximum word count in a sentence for the whole dataset.

Baseline+ Model: Subjectivity Classifier

The sentences in the dataset can be either **objective** - which are facts, or **subjective**, which are opinions. Sentiment Analysis tasks are only dependent upon the Subjective sentences, because these are the ones that express the intent of the content. Therefore, detection of bias is supposed to come only from the subjective sentences, and thus, we give preference to subjective sentences over objective sentences in choosing the 200 sentences for each article.

[5] https://www.cs.cornell.edu/people/pabo/movie-review-data/

The model uses ELMo embeddings to represent each sentence as a tensor of all its words' ELMo representations. These ELMo representations are passed as input to a Convolutional Network.

The Convolutional Network consists of 3 convolutional layers, each followed by the application of the RELU activation function. After this, Maxpooling was applied to the outputs of each RELU layer, and these were concatenated to give the outputs. Similar to the previous Neural Network, we made use of Binary Cross Entropy loss to train the model. The dataset was split in the ratio 8:2

Now, we improve the baseline model by finding out all the subjective sentences from the article. Then, for picking the 200 representative sentences for an article, we make use of the subjective sentences first, and take any remaining from the objective sentences. This would increase the chances of subjective sentences coming in the 200 sentences limit, and thus we expected it to improve the accuracy of our model.

Baseline+ Model: BERT

The BERT model aims to do a "masked language modeling" instead of traditional "language modelling" i.e. it randomly replaces certain words in a sentence with a special "masked" token (with a low probability) and then uses a Transformer to generate prediction for the masked word based on the remaining unmasked words surrounding it from both the sides.

Also, it is a more deep bidirectional model whereas ELMo uses a concatenation of right-to-left and left-to-right LSTM's. The embeddings generated from BERT have shown to perform better than any other model till now.

So, we made another Baseline+ model in which we replaced the ELMo embeddings with those obtained from BERT. For getting these embeddings, we used S-BERT which directly gave us embeddings for a sentence which we were obtaining earlier by taking the mean of the word embeddings in our Baseline model. These embeddings were of length 384 which were different from ELMo which gave embeddings of size 1024.

After obtaining the embeddings, these are then again passed into a CNN model which will learn to classify whether the article contains hyper-partisan bias or not. The only difference from the previous models was to incorporate the change in the size of embeddings from ELMo to BERT.

Thus, with the usage of Transformer-based embeddings in place of the simple bidirectional LSTM embeddings of ELMo, we expected an improvement in accuracy of the outputs.

Baseline++: Combined Baseline+

Now, we combined the two improvements to create the Baseline++ model of our project, which combines the CNN network along with the prioritized Subjectivity classification and the usage of deeper embeddings using the BERT embeddings in place of ELMo.

For the implementation, we had to store the sentences which were subjective in a given article separately in a JSON file, since the CUDA memory wasn't able to handle loading both the embeddings at the same time, as well as version conflicts existed between dependencies of the two models.

Thus, for the Baseline++, we iterated over the articles and found out the subjective sentences in the article from the JSON file. Then, we take the 200 sentences while giving priority to the subjective sentences and convert them into BERT embeddings and feed them to the CNN for classification.

The trained models / weights were uploaded on Google Drive[6] due to size issues

[6] https://drive.google.com/drive/folders/1i5qDhrat7Gs5jMVtRywcPQBKpD15zYMO?usp=sharing

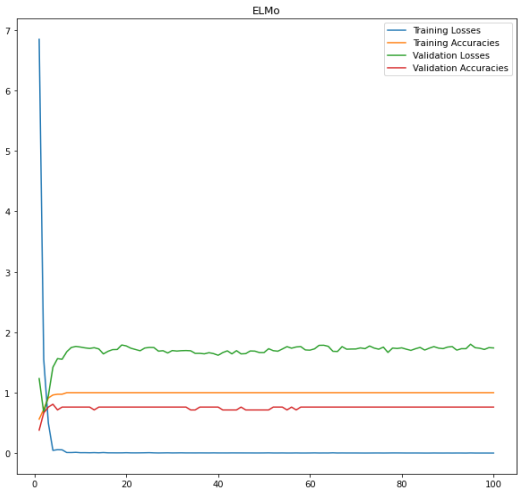
OBSERVATIONS
RESULTS/ANALYSIS

MODEL	ACCURACY	PRECISION		RECALL		F-1 SCORE	
		F	T	F	T	F	T
ELMo	76.19%	0.67	0.83	0.75	0.77	0.71	0.80
ELMo + Subjectivity	78.29%	0.86	0.69	0.82	0.76	0.84	0.72
BERT	79.07%	0.81	0.65	0.79	0.68	0.80	0.67
BERT + Subjectivity	80.62%	0.82	0.73	0.85	0.69	0.84	0.71

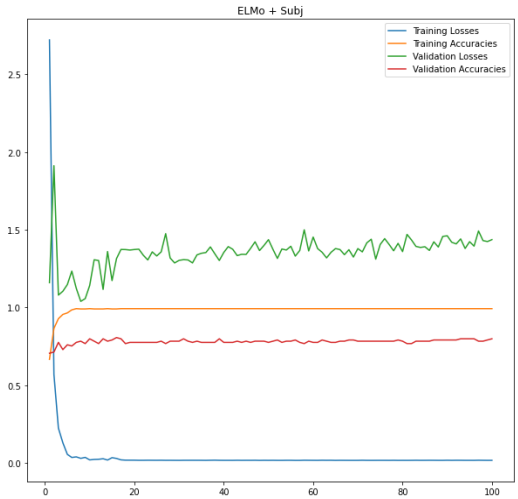
- Here, we can see that the False precision is much lower for the ELMo model as compared to that of the True class. This indicates that the model is more biased towards False class, and it treats a lot of positive sentences as False class too. This might be because the bias sentences might be further ahead in the dataset, and did not come within the 200 sentences limit. However, when we give preference to subjective sentences, we can see that the precision of False sentences increases drastically, i.e. the True sentences labelled as False drastically decreases. Similar observations are made for the models using BERT as well.
- We notice BERT improves the performance of the baseline significantly. However, the improvements through subjectivity over the BERT model are little, which shows that the deeper trained embeddings are able to track sentiments to such a good extent that subjectivity priority does minor improvements.
- Also, we expected BERT to give more boost to the performance, being much more powerful than ELMo. It is possible that the comparatively lesser increase is because we are using Convolutional Neural Networks on the embeddings, which bring down the complexities represented by embeddings as deep as BERT. Thus, the CNN is not able to fully utilize the information given by BERT, and RNNs or LSTMs might have proven to give bigger boosts to performance in the BERT models.

NOTE: The metrics Precision, Recall and F1-score are calculated after the epoch number 100 for all the models. The accuracies mentioned are the most frequent ones among all the epochs of a model.

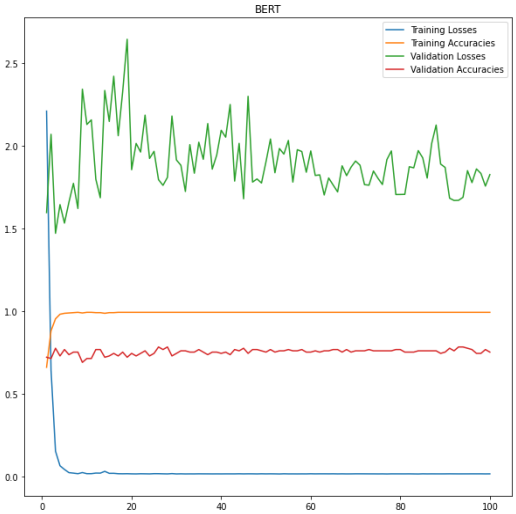
Below are the graphs for all the models showing their losses and accuracies on training and validation:



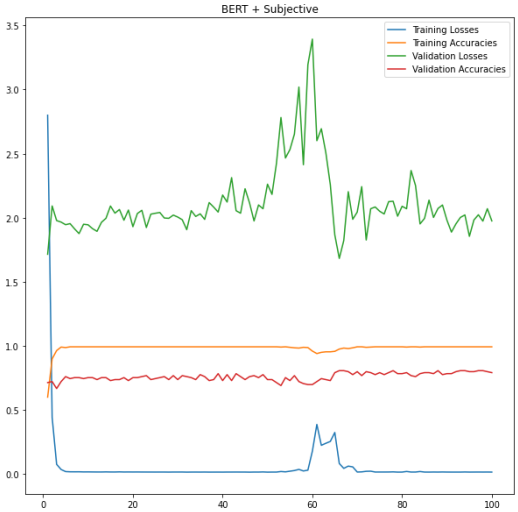
ELMo Model Accuracies and Losses



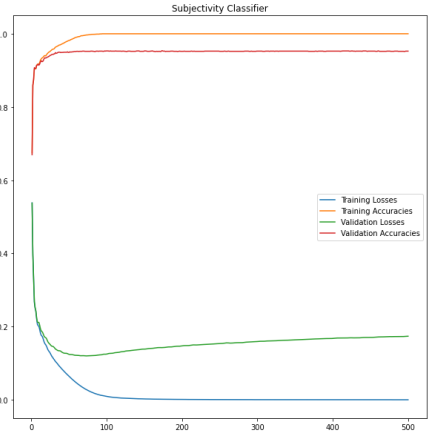
ELMO+Subjectivity Accuracies and Losses



BERT Model Accuracies and Losses



BERT + Subjectivity Accuracies and Losses



Subjectivity Classifier Accuracies and Losses

CHALLENGES FACED

- We had trained our Sentence Subjectivity classification model on ELMo embeddings. For this reason, while attempting to check the subjectivity of the sentences in Baseline++ model, we were also finding out the ELMo embeddings for the sentences for classifying them in our classifier model along with finding out their BERT embeddings using S-BERT. Using so many embeddings was using a lot of memory and our machine on ADA with 4 RTX 2080Ti GPUs and 40 CPUs was also not able to fit them. So, we wrote another script to first determine all the subjective sentences in each article and store their indices in a JSON file which was then loaded in Baseline++ model to determine the subjectivity of sentences.
- The larger dataset for articles, which was created semi-automatically by using publisher bias was not an accurate dataset in comparison to the manually annotated dataset. The accuracy of our model dipped by a huge margin, despite the large size of the dataset with **7,54,000 articles**. This shows the lack of properly generated dataset.
- The manually annotated dataset for articles was very small and consisted of only 1,200 articles. So, the model has seen very limited data and cannot perform well in the real world.
- The sentence subjectivity classification dataset was also small in nature, containing only 10,000 sentences. Thus, the model was able to see limited data in order to correctly detect subjectivity in articles.
- Some articles contained unreadable characters, and so had to be removed from being considered for classification.

CONCLUSION

The simple model with ELMo embeddings and Convolutional Neural Networks gave good results and formed a high-benchmark baseline, achieving an accuracy of **76.19%**. Significant boost was added through prioritizing subjective sentences over objective sentences, as the ELMo+Subjectivity model gained an accuracy of **78.29%**.

In addition to these, the use of much deeper embeddings - BERT seemed to improve the accuracy even higher, with the model having accuracy of **79.07%**. Coupling both the improvements gave the BERT+Subjectivity model, which increased the accuracy to **80.62%**, giving the best accuracy out of all the models taken.

REFERENCES

[1] Jiang, Ye, et al. “Team Bertha von Suttner at SemEval-2019 Task 4: Hyperpartisan News Detection Using ELMo Sentence Representation Convolutional Network.” Proceedings of the 13th International Workshop on Semantic Evaluation, Association for Computational Linguistics, 2019, pp. 840–44. ACLWeb, https://doi.org/10.18653/v1/S19-2146.

[2] Kiesel, Johannes, et al. “SemEval-2019 Task 4: Hyperpartisan News Detection.” Proceedings of the 13th International Workshop on Semantic Evaluation, Association for Computational Linguistics, 2019, pp. 829–39. ACLWeb, https://doi.org/10.18653/v1/S19-2145.

[3] Reimers, Nils, and Iryna Gurevych. “Sentence-BERT: Sentence Embeddings Using Siamese BERT-Networks.” ArXiv:1908.10084 [Cs], Aug. 2019. arXiv.org, http://arxiv.org/abs/1908.10084.