Preliminary Exploration of BERT for Stance Detection

Yu Cheng Li
University of Waterloo
yc22li@uwaterloo.ca

Jackson Zheng
University of Waterloo
j75zheng@uwaterloo.ca

Abstract

The Fake News Challenge Stage 1 (FNC-1) was a stance detection challenge open to the public in 2017, where participants were called to leverage machine learning and natural language processing tools to help distinguish between legitimate and fake news [1]. A dataset containing 49,972 articles were labelled as “unrelated”, “agree”, “disagree”, or “discuss”. For this particular problem, several experiments were conducted using the state-of-the-art BERT model which stands for Bidirectional Encoder Representations from Transformers. These experiments included loss balancing, data augmentation, and an ensemble approach using 2 BERT models. The best performing experiment was the 2 BERT model with both loss balancing and data augmentation. The resulting F1 scores in ascending order were disagree, agree, discuss, and unrelated, which also reflected the data imbalance of the training set.

1 Introduction

The experiments presented in this paper is work done to tackle the Fake News Challenge Stage 1 [1]. Fake news detection can be broken down into different steps, where the first step is stance detection. The goal of this challenge was to explore how machine learning could be used in stance detection, that is, identifying whether an article was unrelated to, agreed with, disagreed with, or discussed a particular headline. The challenge provided a training and a competition test set. Each of the data sets contained headlines, body articles, and their associated stance labels. For this problem, several experiments were conducted using the state-of-the-art BERT model which stands for Bidirectional Encoder Representations from Transformers.

2 Background

The work done in this paper used BERT [2]. BERT was trained on an extremely large linguistic dataset, and is commonly fine-tuned for other natural language processing tasks, like sentiment analysis, question answering, and language inferencing. BERT was selected here as it is a powerful pre-trained model for bidirectional representations of unlabelled text data. Its ability to process longer sequences of text compared to LSTMs and retain contextual information about the corpus provided an advantage for this stance detection problem.

3 Approach

3.1 Neural Network Architecture

The final neural network architecture chosen was an ensemble of two BERT models. The first model was fine-tuned on binary classification between the “unrelated” and “related” stances. The second model was fine-tuned on ternary classification between the related stances “agree”, “disagree”, and “discuss”. The intuition behind this design was that better results would be obtained if one BERT model specialized in just distinguishing between the related stances, of which the “agree” and “disagree” classification problem was particularly nuanced. More details will be discussed in the experiments section. Both the binary and ternary classification models used a softmax output layer with cross entropy loss.

3.2 Preprocessing

The headlines and body articles were tokenized using the BERT tokenizer. When BERT was trained, each token was given a unique token ID. Thus, to use the pre-trained BERT model, the tokenizer first converted each token into their corresponding token IDs. Instead of using the [UNK] token for out-of-vocabulary words, BERT uses a WordPiece algorithm to break down those words into familiar tokens.

For this challenge, there were two pieces of raw data – the headline and the body article. To reconcile the separate data inputs, the headline and body article were concatenated and fed into BERT.

The tokenizer encoding function was called to handle the tokenization of the input, assign token IDs, apply special tokens (i.e., [CLS] at the start of each sequence to indicate that this was a classification problem, [SEP] at the end of each headline and article body, and [PAD] for padding tokens to ensure each input was the same length). The encoding function also returned an attention mask, which was a binary mask that informed the model which tokens within each input sequence encoded actual information (e.g., 1 – word token, 0 – padding token).

The maximum length of the input sequences was determined from inspecting Figure 1 – as a significant portion of the body articles had a token count of approximately 250 tokens, the maximum length of each input sequence was padded or truncated to 250 tokens. This decision did not take into account the length of headlines, as the headlines were placed first in the data concatenation process and their lengths were not deemed significant.

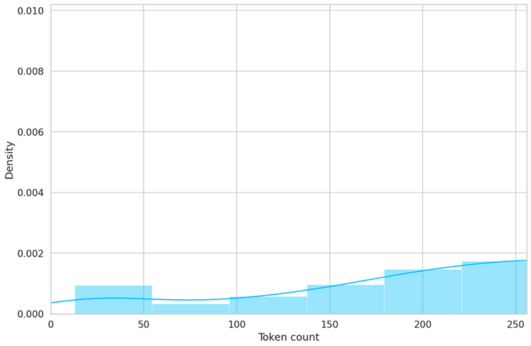

Figure 1: Token length distribution across all body articles in the training set.

3.3 Data Balancing

The training data contained 49972 samples, out of which 36545 were unrelated, 3678 were agree, 840 were disagree, and 8909 were discuss. This is expressed in Figure 2. It is easy to see that the data was extremely imbalanced. To handle the data imbalance, class weighting and data augmentation were applied. These experiments will be discussed in later sections.

3.4 Hyperparameters

The parameters chosen were a batch size of 16, a learning rate of 2e-05, and a dropout rate of 0.3.

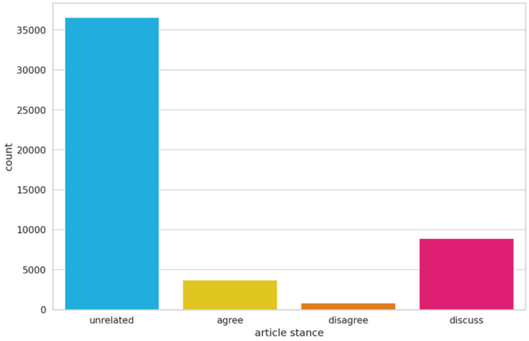

Figure 2: Class distribution of the original training set.

The parameters were chosen as recommended by the BERT authors [2] and as demonstrated in the provided MSCI 598 BERT tutorial [3].

3.5 Optimizers

Adam optimization and a linear scheduler with no warmup was used.

4 Experiments

Several experiments were conducted with the goal of achieving the best possible performance.

The sections describe each experiment and their results in more detail. Note that most of the experiments only underwent 2-5 training epochs as per the recommendation for fine-tuning by the original BERT authors [2]. It was often observed that after the fifth epoch, the training accuracy would reach stagnation.

4.1 Experiment 1 - Baseline BERT

A baseline model was trained for 2 epochs using a pre-trained BERT model. From this initial experiment, the low recall score for the disagree class informed that the model was incorrectly predicting disagreeing stances as other stances. This motivated more training strategies that were later implemented to help the model better identify disagreeing stances.

	Precision	Recall	f1-score
Agree	0.56	0.68	0.61
Disagree	0.29	0.11	0.16
Discuss	0.82	0.78	0.80
Unrelated	0.98	0.99	0.98

Table 1: Classification report for the baseline BERT model.

4.2 Experiment 2 - Loss Balancing

The same model in experiment 1 was used but with the addition of loss balancing or class weights. From experiment 1, it was observed that the performance for unrelated samples was very high, while the model did poorly on classifying between agree and disagree. This is expected due to the data imbalance. The class weights applied for the unrelated, agree, disagree, and discuss stances were 0.357, 3.125, 15.625, and 1.388 respectively. The class weights were calculated using the following equation.

$$weight = \frac{n}{n_c \times n_j} \tag{1}$$

where n was the total number of data points, n_c was the number of classes, and n_j was the number of data points per class. The rationale was that by applying class weights, the model would focus more attention on the agree and disagree samples. Looking at Table 2, the resulting scores with loss balancing were actually worse than those without from the baseline model.

	Precision	Recall	f1-score
Agree	0.48	0.66	0.56
Disagree	0.31	0.08	0.12
Discuss	0.75	0.74	0.74
Unrelated	0.98	0.97	0.98

Table 2: Classification report for the baseline BERT model with loss balancing.

4.3 Experiment 3 - Downsampling Unrelated Samples

Another strategy to handle the data imbalance problem was to train BERT using only half of the unrelated samples. Table 3 shows that the results from downsampling were significantly worsened compared to the baseline experiment. While the initial intuition for downsampling was to make the dataset less biased towards the unrelated class, the end effect on the model was not a performance improvement in predicting the other classes, but just an overall performance deterioration in predicting all classes.

4.4 Experiment 4 - Ensemble Approach using 2 BERTs

	Precision	Recall	f1-score
Agree	0.43	0.67	0.52
Disagree	0.26	0.09	0.13
Discuss	0.50	0.84	0.63
Unrelated	1.00	0.80	0.89

Table 3: Classification report for the baseline BERT model with downsampling of the unrelated class.

As another experiment to handle data imbalance, 2 BERT models were used in an ensemble approach. The idea was that the first model would classify an article as either unrelated or related, and if it was related, then the second model further classified it as agree, disagree or discuss. The rationale was that classification generally becomes more difficult with more classes, and therefore splitting the model into two binary and ternary classification problems could result in better performance.

In addition, the ensemble approach also helped with the data imbalance problem. While the second model still had an imbalance problem due to the small amounts of disagrees, the large imbalance due to unrelated samples was removed.

Another reason to try the ensemble approach was to improve training efficiency. The baseline BERT model took a substantially long time to train (over 1 hour per epoch) due to having 49972 data points. Training the two models separately avoided having to repeatedly train on such a large dataset. As the binary classification model performed well on the first try (Table 4), further training was not necessary. The second model, which had the more challenging classification task, also only had 13427 data points, so it trained a lot faster throughout various other experiments.

For the first model, all the samples related to agree, disagree and discuss were combined into a single class to represent related. Class 0 represented unrelated articles and class 1 represented related articles. For the second model, classes 0, 1, 2 represented the stances agree, disagree, and discuss. To use both models for prediction, the following procedure was used: first, the input was passed into both models, then, the output of the second model was offset by 1 to match the original class indices of 1, 2, 3 representing agree, disagree, and discuss. The output of the first model was used as a binary mask, and the final prediction was an element-wise product between the binary mask and the output of the second model.

Experiments 5 - 8 were conducted on the 2 BERT model.

	Precision	Recall	f1-score
Unrelated	0.98	0.99	0.98
Related	0.96	0.94	0.95

Table 4: Classification report for the binary model in the 2 BERT ensemble.

	Precision	Recall	f1-score
Agree	0.37	0.65	0.47
Disagree	0.15	0.15	0.15
Discuss	0.85	0.58	0.69

Table 5: Classification report for the ternary model in the 2 BERT ensemble.

4.5 Experiment 5 - Flipping the Headline and Body Order

This experiment differed from the baseline BERT experiment by flipping the order of the headline and body articles during the input data concatenation stage. The body article was first in the sequence followed by the headline. As a result, the maximum sequence length was increased from 250 tokens to 270 tokens to prevent the truncation of entire headlines from the inputs to the model. The training underwent 5 epochs.

	Precision	Recall	f1-score
Agree	0.53	0.74	0.62
Disagree	0.43	0.06	0.11
Discuss	0.85	0.81	0.83

Table 6: Classification report for the ternary model with headline and body article order flipped.

The original intuition was that it might be easier for the model to learn the general nature of the body text before seeing the headline; however, given that BERT is bidirectional, and the maximum token length was also increased, it was unclear to which change the performance improvements to the agree and discuss classes should have been attributed. Looking at Table 6, the main objective to improve the disagree scores was not met, as recall halved, which meant that almost all the disagreeing stances were still being predicted as agree or discuss. Thus, this change was not carried forward into the subsequent experiments.

4.6 Experiment 6 - Data augmentation of Headlines (Synonyms)

An additional approach that was used to compensate for the data imbalance problem was applying data augmentation to the disagreeing headlines. The methodology here was to apply synonym text transformation to each of the original headlines, whereby each headline had approximately 50% of its tokens replaced with synonyms. The tokens that were replaced were chosen at random, so long as they were not stop words, to prevent the removal of stop words that may have indicated stance, like "not". Each synonym was randomly chosen from a synonyms list at random, which was also filtered for stop words. For instance, the headline:

> "Weather Reporter Caught Writing His Name In The Snow Was NOT Ready To Go On Camera (UPDATED)"

was transformed to:

> "atmospheric condition newsperson enchant drop a line His identify In The Snow comprise NOT cook To go bad On Camera (UPDATED)"

The original headlines were transformed four times, producing an overall disagree stance count of $840 \times 5 = 4200$, minus any possible duplicates (which infrequently occurred if the transformed headlines were unchanged from the original headlines). The resulting stance data distribution is shown in Figure 3.

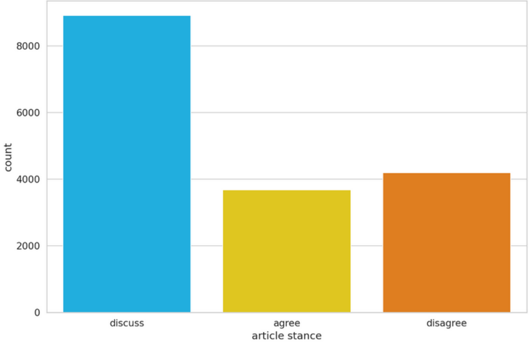

Figure 3: Class distribution for ternary BERT model after data augmentation of disagreeing headlines.

	Precision	Recall	f1-score
Agree	0.56	0.74	0.64
Disagree	0.47	0.08	0.13
Discuss	0.86	0.85	0.85

Table 7: Classification report for the ternary model with data augmentation on the disagreeing headlines.

Table 7 shows that even with 5 fold more disagreeing data points, the model was still confusing this class with other classes. Note that this model was trained with 10 epochs, but stagnation occurred after the third epoch. Efforts were also made to prevent stagnation (it was thought that the model was getting stuck at a local minimum) by increasing the learning rate to 2e-4 and 0.01. However, these attempts only got the model to stagnate at even lower training accuracies.

4.7 Experiment 7 - Data Augmentation of Headlines and Body Articles (Synonyms + Machine Translation)

Since a headline-body article pair received most of its token contribution from the body article, it made sense to also perform data augmentation on the body articles that corresponded to disagreeing headlines. In this experiment, the headlines were transformed using synonyms as in the previous case, but the body articles were augmented with a combination of two methods: synonym replacement and machine translation using Google Translate. First, each body article had approximately 50% of its tokens replaced, and then the text would undergo two layers of translation until it was converted back to English. For each article, the target languages for translation were selected at random from a list of languages that were selected to be significantly different from English, such as synthetic languages (e.g., Russian or Finnish). An example of this process is given for the original article:

> “There’s not being ready to go on camera. And then there’s really, really not being ready. Like, not even a little bit. This was the unfortunate case experienced by meteorologist Mike Seidel this weekend while reporting on the stormy weather in North Carolina. . . ”

which was transformed to:

> ”There is no live camera on the camera. Then there really is, honestly not being prepared. Like, do not wash for a while. It was an ominous live costume from the meteorologist Seidel’s microphone over the weekend, while covering the stormy wind in North Carolina. . . ”

As can be seen, the transformed body texts did not make total linguistic sense, but the hope was that it would introduce more learning ”curve balls” to the model, encouraging it to understand how to recognize a disagreeing stance. The articles that corresponded to disagreeing stances in the original dataset were transformed only once (due to time constraints), so the overall number of body texts increased by 291 articles to a total of 1974 unique articles.

The model was trained for 3 epochs. Table 8 shows a noticeable improvement in predicting disagreeing stances from experiment 6.

	Precision	Recall	f1-score
Agree	0.56	0.75	0.64
Disagree	0.50	0.11	0.17
Discuss	0.87	0.85	0.86

Table 8: Classification report for the ternary model with data augmentation on the disagreeing headlines and body articles.

4.8 Experiment 8 - Additional Loss Balancing with Data Augmentation

An extension to experiment 8 was performed where the loss function weights were shifted to be even more biased towards the disagree class. The weights were calculating using equation 1, and scaling factors of 0.6 and 1.4 were applied to the calculated weights for agree and disagree, respectively. The resulting weights were: [0.9122, 1.8692, 0.6276]. Comparing Table 9 to Table 8, the additional loss balancing generally worsened the model predictions.

	Precision	Recall	f1-score
Agree	0.57	0.69	0.62
Disagree	0.52	0.07	0.12
Discuss	0.83	0.88	0.85

Table 9: Classification report for the ternary model with data augmentation on the disagreeing headlines and body articles with additional loss balancing.

4.9 Evaluation Metrics

The metrics used for evaluation were precision, recall, f1 score, and accuracy. In addition, the challenge used a scoring metric as follows: 0.25 for each correct unrelated prediction, 0.25 for each correct related prediction, and 0.75 for each correct agree, disagree, or discuss prediction.

4.10 Results

Reviewing all the data so far, experiment 7 was the most promising at it provided the best scores for the related classes (even in comparing to the baseline BERT model). Table 10 shows a comparison of f1-scores from experiments 1-8 for only the three related classes. These results verified that augmenting both the headline and body article datasets to have more disagreeing training examples helped the model perform better on the test set with respect to the related categories.

Exp. No.	Agree	Disagree	Discuss
1	0.61	0.16	0.80
2	0.56	0.12	0.74
3	0.52	0.13	0.63
4	0.47	0.15	0.69
5	0.62	0.11	0.83
6	0.64	0.13	0.85
7	0.64	0.17	0.86
8	0.62	0.12	0.85

Table 10: Comparing f1-scores for the agree, disagree, and discuss predictions made throughout experiments 1-8.

After reconciling the predictions from both BERT models using the methodology mentioned in Section 4.4, the ensemble predictions achieved an accuracy score of 0.910, or a test score of 9952.75. The confusion matrix is shown in Table 11.

	Agree	Disagree	Discuss	Unrelated
Agree	1374	34	395	100
Disagree	425	61	133	78
Discuss	627	26	3578	233
Unrelated	68	6	156	18119

Table 11: Confusion matrix for the 2 BERT model.

The model performed well overall on all classes except for the disagree class, as it still got predicted as agree 425 times and discuss 133 times.

5 Conclusion

5.1 Limitations

Some of the experiments were only trained for the minimum of 2 epochs due to hardware and time constraints. The results for the experiments may be improved with more extensive training, for example, 5 or more epochs.

5.2 Learnings

These experiments illustrate the challenge of handling dataset imbalances. Since there was an extremely small amount of data points for the disagree category, the model poorly predicted this class. In all the experiments, most of disagreeing stances were incorrectly classified as agree, which indicated that the agree and disagree samples were more similar than expected. Surprisingly, standard techniques to counter imbalanced datasets such as loss balancing and downsampling were ineffective. Experiments 4-8 showed that an ensemble approach may be useful for increasing training speed without a decrease in performance in the case of a big class imbalance. In addition, the experiments demonstrated that leveraging data augmentation to generate more samples is a good technique to handle class imbalance.

5.3 Recommendations

Collect more disagreeing samples - In the future, a lot more samples should be collected on disagreeing headline-article pairs, as this was the main bottleneck for model performance. Having more samples would help the model to better distinguish between agreeing and disagreeing stances.

Explore more data augmentation techniques - In the event that collecting more samples is not possible, more data augmentation techniques should be explored. For example, the use of replacing synonyms and machine translation created some samples that were unlikely to be written by a reporter, or did not make sense at all. Perhaps the use of translating to different languages may be more effective than others. **Use data from a sentiment analysis corpus** - Another interesting idea that may be explored is the use of borrowing data from a sentiment analysis corpus. Positive and negative sentiment may be similar enough to agreeing and disagreeing stances and the use of such data may help increase performance. In addition, sentiment analysis has been widely studied and there are many large corpora available online.

References

[1] D. Pomerlau and D. Rao, “Fake news challenge stage 1 (FNC-I): Stance detection,” *Fake News Challenge.* [Online]. Available: http://www.fakenewschallenge.org/.

[2] U. Kamath, K. L. Graham, and W. Emara, “Bidirectional encoder representations from Transformers (Bert),” Transformers for Machine Learning, pp. 43–70, 2022.

[3] Google colaboratory, Google Colab Available: https://colab.research.google.com/github/kiasar/NLP-tutorials-waterloo-MSCI-598—Winter-2022/blob/main/Week%2012/08_sentiment_analysis_with_bert.ipynb.

6 Supplmentary Material

The following link leads to the project GitHub repository where three Google Colab notebooks can be found:

https://github.com/S-Li/msci-nlp-w22/tree/main/fnc-1

The three notebooks pertain to:

- training the binary model
- training the ternary model
- reconciling the predictions