NLP Report

Introduction

In this report we will explore fake news detection using a hierarchy of machine/deep learning models. We will present our solutions and accuracies and discuss our choices and results.

Problem Definition

Facilitation of fake news detection via stance prediction. Given a headline and an article body, the solution will predict whether the article agrees, disagrees, discusses or is otherwise unrelated to the headline. This is done in two halves: a unrelated/related classification problem and an agree/disagree/discuss problem.

Proposed Solutions

We propose a two-level hierarchy of models in order to categorise a headline/body pair into a labelled class. The first model takes a headline/body pair and predicts whether they are related or unrelated. Upon a related prediction, we then feed the pair into our second model, that classifies the pairing as one of: agree, disagree, or discuss. We trained multiple versions of the first model, using both TF-IDF and Transformer embeddings as input.

TF-IDF

Term Frequency – Inverse Document Frequency (TF-IDF) uses the rarity of a word over a document corpus in order to measure the importance of the information it carries i.e. the rarer the word in the corpus, the more weight it carries. This is a simple to compute measure that represents individual word value, however, it does not capture the overall word meaning i.e. tf-idf is a lexical, not a semantic measure.

$$tf(t,d) = \log\big(1 + f(t,d)\big) \; where \; f(t,d) = \; frequency \; of \; term \; t \; in \; document \; d$$

$$idf(t,D) = \log\left(\frac{N}{count(d \in D : t \in d)}\right)$$

$$where \; N = \; number \; of \; documents \; in \; the \; corpus \; D$$

$$tfidf(t,d,D) = tf(t,d) \cdot idf(t,D)$$

Equation 1: TF-IDF

SBERT

Bidirectional Encoder Representations from Transformers (BERT) is a transformer-based language model capable of embedding words based on previous and future contexts. Pretrained BERT models allows us to produce embeddings for words.

Sentence-BERT (SBERT) is a “modification of the pretrained BERT network” [4] that instead of embedding words, embeds full sentences which “can be compared using cosine-similarity”. This ties in well with our “unrelated” problem definition.

Cosine Similarity Distance

If we consider our word/sentence embeddings to be a vector in the language space, we can computer the relatedness between a pair by calculating the vector distance between them. This is achieved using the Cosine Similarity Distance (Eq 2).

$$simlarity = \frac{A \cdot B}{\|A\|\|B\|} \; where\ A, B\ are\ embeddings$$

Equation 2: Cosine Similarity Distance

Logistic Regression

We chose Logistic Regression as our machine learning method due to the problem definition of unrelated classification being a binary problem. Logistic Regression is a powerful method for 0-1 classification tasks, especially given our data input comes in the form of embeddings.

Processing

The only pre-processing step we use is to remove stop words.

Solution 2ai(1a): TF-IDF Machine Learning Unrelated/Related Classification

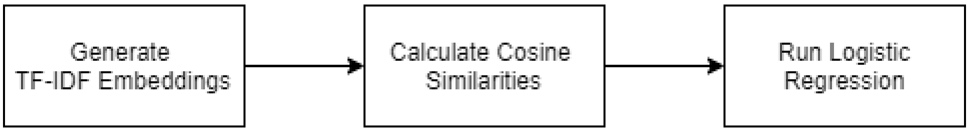
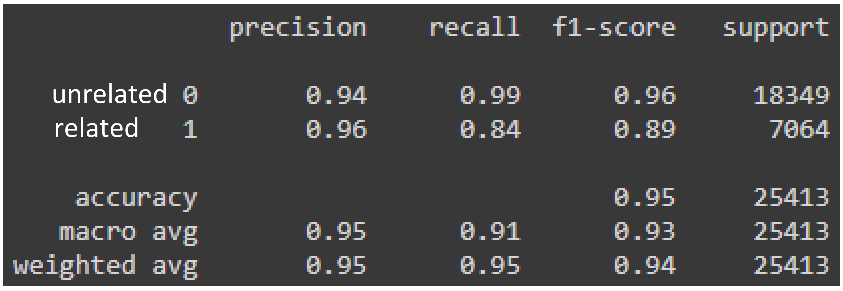


Figure 1: Pipeline of Solution 2ai(1a)



```
              precision    recall  f1-score   support

 unrelated 0       0.94      0.99      0.96     18349
 related   1       0.96      0.84      0.89      7064

    accuracy                           0.95     25413
   macro avg       0.95      0.91      0.93     25413
weighted avg       0.95      0.95      0.94     25413
```

Table 2: Classification Metrics of Solution 2ai(1a)

Solution 2ai(1b): SBERT Machine Learning Unrelated/Related Classification

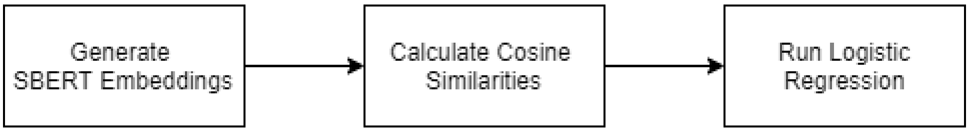
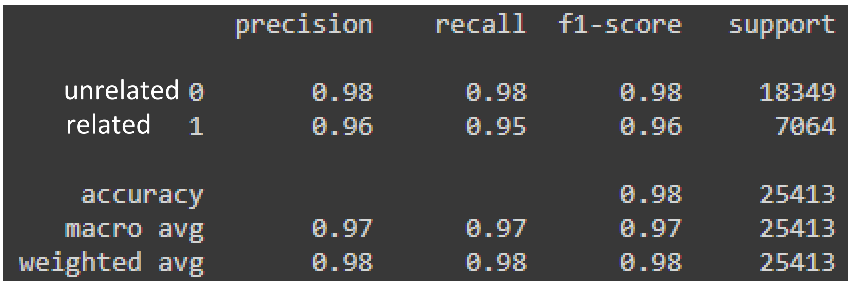


Figure 2: Pipeline of Solution 2ai(1b)



```
              precision    recall  f1-score   support

 unrelated 0       0.98      0.98      0.98     18349
 related   1       0.96      0.95      0.96      7064

    accuracy                           0.98     25413
   macro avg       0.97      0.97      0.97     25413
weighted avg       0.98      0.98      0.98     25413
```

Table 3: Classification Metrics of Solution 2ai(1b)

Solution 2aii(1a): TF-IDF Deep Learning Unrelated/Related Classification

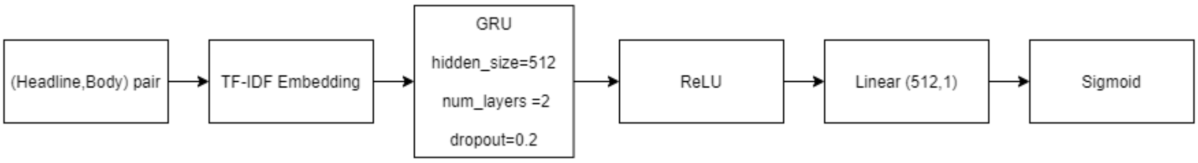


Figure 3: Model of Solution 2aii(1b)

Test Accuracy = 72%

Solution 2aii(1b): SBERT Deep Learning Unrelated/Related Classification

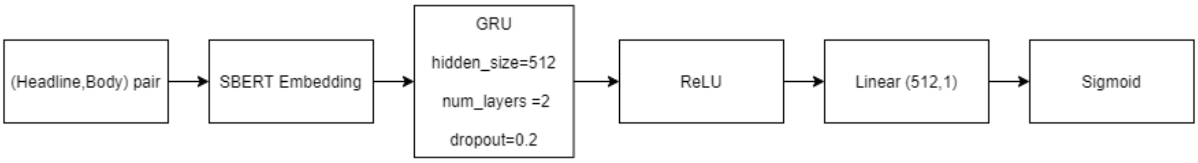


Figure 4:Model of Solution 2aii(1b)

Test Accuracy = 87%

Model Justification

We use a GRU due to the long sequences found in the body text – alternative models such as RNNs suffer from short term memory loss and we wish to avoid that. We also favour the GRU over LSTMs due to its similar performance but superior efficiency.

We have used PyTorch Lightning [6] in our solution which comes equipped with an automatic learning rate identifier for our ADAM optimiser. Therefore, we do not state the learning rate for our deep learning models are they are determined at runtime.

For our loss function, we opted to use CrossEntropyLoss – the standard for RoBERTa and BinaryCrossEntropy for the unrelated/related models. We do this as we have a multiclass problem and the definition of the loss goes hand-in-hand with the problem.

RoBERTa

For Agree/Disagree/Discuss classification, we require the semantic meaning of individual words – not sentences – in order to predict stance. Therefore, SBERT is no longer applicable and we can revert back to BERT. However, BERT has been improved over time, and one group has proposed RoBERTa: Robustly Optimized BERT Pretraining Approach [5]. This pretrained-model is claimed to be a direct improvement over BERT, so we use it in our solution. For shorter training times, we use DistilRoBERTa [8].

Solution 2bi: RoBERTa Deep Learning Agree/Disagree/Discuss Classification

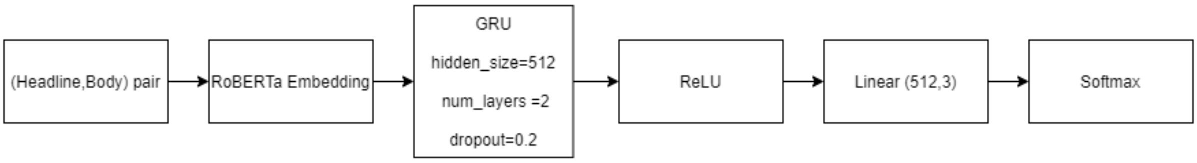
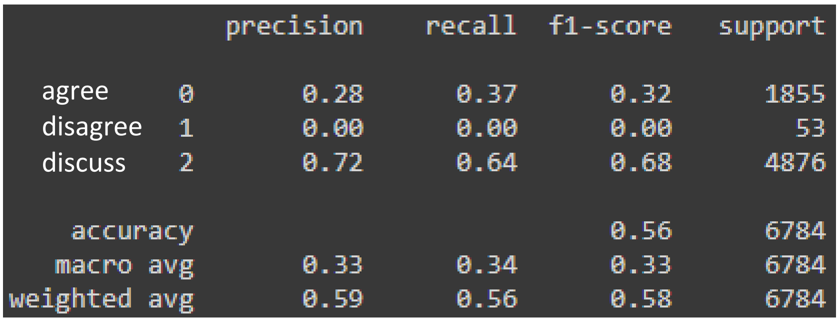


Figure 5: Model of Solution 2bi

Due to the data imbalance in classes, we use an oversampling technique wherein we forcibly return disagree training data 10% of the time and agree 20% of the time. Tweaking these values may lead to better results.



```
                 precision    recall  f1-score   support

  agree      0        0.28      0.37      0.32      1855
  disagree   1        0.00      0.00      0.00        53
  discuss    2        0.72      0.64      0.68      4876

       accuracy                           0.56      6784
      macro avg       0.33      0.34      0.33      6784
   weighted avg       0.59      0.56      0.58      6784
```

Table 4: Classification Metrics of Solution 2b

Analysis of Results

Model	Overall Accuracy(%)	Unrelated Accuracy(%)	Related Accuracy(%)	Time to Train(s)
TF-IDF Logit Regression	95	94	96	0.12
SBERT Logit Regression	98	96	98	0.08
TF-IDF Deep Learning	72	-	-	~2000
SBERT Deep Learning	87	-	-	~2000

Table 5: Machine Learning Accuracies and Training Times

Our machine learning method have superior accuracy to our deep learning methods (95%/98% vs 72%/87%). The deep learning accuracies may improve further with longer training times and further hyperparameter tweaking but the computational efficiency (training in seconds vs minutes) coupled with great accuracy of the machine learning method leads us to conclude that Logistic Regression with SBERT embeddings is the superior method.

Model	Overall Accuracy(%)	Agree Accuracy(%)	Disagree Accuracy(%)	Discuss Accuracy(%)	Time to Train(s)
RoBERTa Deep Learning	56	28	0	72	~1200

Table 6: RoBERTa Accuracy and Training Time

Our RoBERTa model has competitive accuracy in the discuss category when compared to the models in Table 5. We believe the disagree accuracy is an error, given the result in Figure 7. Agree accuracy, while low, may be improved via further training – as the loss in Figure 6 shows learning potential.

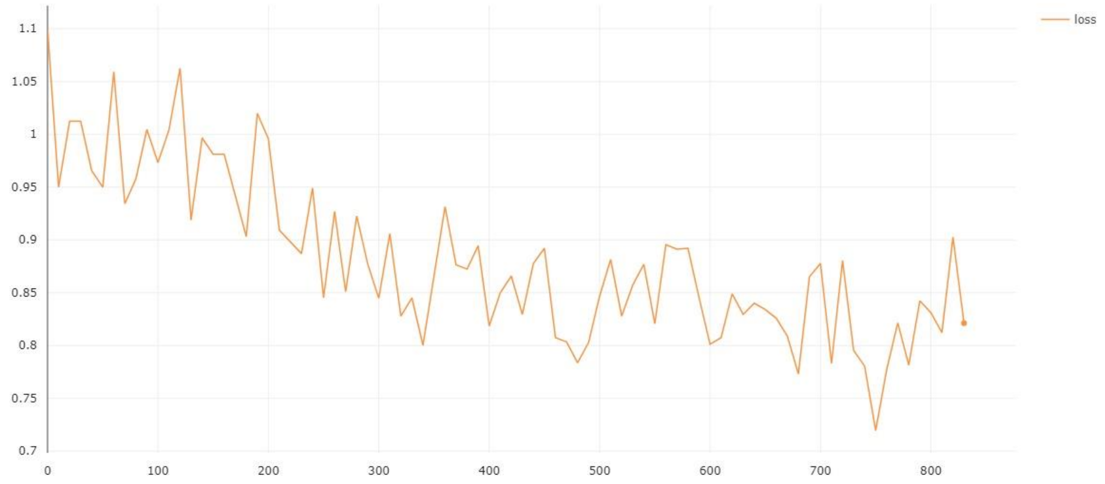
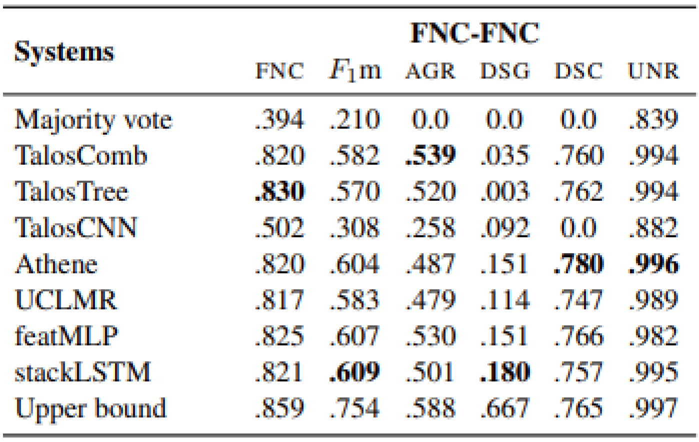
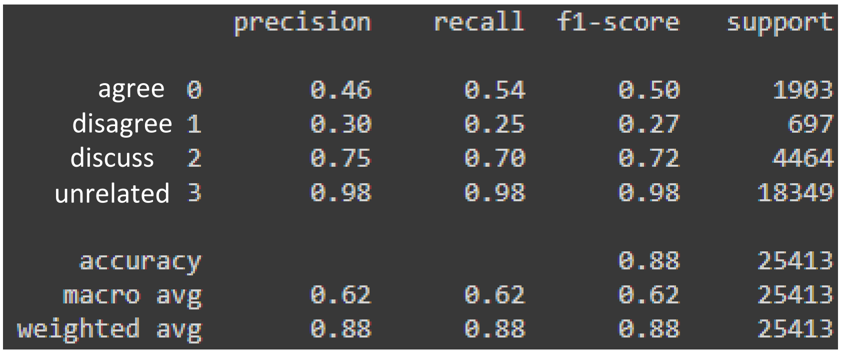


Figure 6:RoBERTa loss graph



Systems	**FNC-FNC**					
	FNC	F_1m	AGR	DSG	DSC	UNR
Majority vote	.394	.210	0.0	0.0	0.0	.839
TalosComb	.820	.582	**.539**	.035	.760	.994
TalosTree	**.830**	.570	.520	.003	.762	.994
TalosCNN	.502	.308	.258	.092	0.0	.882
Athene	.820	.604	.487	.151	**.780**	**.996**
UCLMR	.817	.583	.479	.114	.747	.989
featMLP	.825	.607	.530	.151	.766	.982
stackLSTM	.821	**.609**	.501	**.180**	.757	.995
Upper bound	.859	.754	.588	.667	.765	.997

Table 7: Accuracies of State of the Art models [7]



```
              precision    recall  f1-score   support

    agree  0       0.46      0.54      0.50      1903
 disagree  1       0.30      0.25      0.27       697
  discuss  2       0.75      0.70      0.72      4464
unrelated  3       0.98      0.98      0.98     18349

    accuracy                           0.88     25413
   macro avg       0.62      0.62      0.62     25413
weighted avg       0.88      0.88      0.88     25413
```

Figure 7: Classification Metrics of End-to-End

Model	Overall Accuracy(%)	Agree Precision(%)	Disagree Precision (%)	Discuss Precision (%)	Unrelated Precision (%)
End-to-End	88	46	30	75	98

Table 8: Summary of Figure 7

Discussion

Our combined model has great performance, especially considering a major part of it is machine learning – not deep learning. The end-to-end model only requires two sets of embeddings - SBERT and RoBERTa – and a single pre-processing step, making for a (relatively) light system, especially considering that we only need RoBERTa embeddings for related news articles.

WE used distilled – trimmed down - variants of the transformers, it may be possible to achieve better accuracy with complete set of parameters for the transformers.

Ethical Implications

Fake News is a modern and recurring problem in our society (as shown in studies [2]) and the proposed solutions can be used to highlight potential fake news for readers and additionally highlight any bias toward the story it contains, thus helping to tackle the problem. However, the foundation of our Deep Learning models (BERT) has been shown to hold discriminatory bias (gender [1] and racial [3]) within its pre-trained models. Therefore, tackling fake news in this way could lead to mislabelling news based on false biases. Furthermore, our solutions are not 100% accurate, especially when the prediction is agree/disagree/discuss. Incorrect labelling of news could damage the reputation of journalists and poor decisions could be made as a result of trusting the output of the system as truth. There are even environmental concerns: the computing power to train these models are significant and the prevalence of deep learning drives demand for electricity-hungry computing farms.

Conclusion

We have created a two-level hierarchal fake news detection model capable of labelling headline/body news article pairs into 4 classes with an overall accuracy of 88%. With tweaking of hyperparameters and additional training, this accuracy may be increased.

References

[1] Kurita, K., Vyas, N., Pareek, A., Black, A. W., & Tsvetkov, Y. (2019). Measuring bias in contextualized word representations. *arXiv preprint arXiv:1906.07337*.

[2] Vosoughi, S, Roy, D, Aral, S. (2018). The spread of true and false news online. 10.1126/science.aap9559

[3] https://towardsdatascience.com/racial-bias-in-bert-c1c77da6b25a (Accessed 20/05/21)

[4] Reimers, N., & Gurevych, I. (2019). Sentence-bert: Sentence embeddings using siamese bert-networks. *arXiv preprint arXiv:1908.10084*.

[5] Liu, Y., Ott, M., Goyal, N., Du, J., Joshi, M., Chen, D., ... & Stoyanov, V. (2019). RoBERTa: A robustly optimized BERT pretraining approach. *arXiv preprint arXiv:1907.11692*.

[6] https://pytorch-lightning.readthedocs.io/en/latest/ (Accessed 10/05/21)

[7] Hanselowski, A., PVS, A., Schiller, B., Caspelherr, F., Chaudhuri, D., Meyer, C. M., & Gurevych, I. (2018). A retrospective analysis of the fake news challenge stance detection task. *arXiv preprint arXiv:1806.05180*.

[8] https://huggingface.co/distilroberta-base (Accessed 20/05/21)