Natural Language Processing

Assignment 2 Report

Anna Fabris
anna.fabris@studio.unibo.it

Miki Mizutani
miki.mizutani@studio.unibo.it

Leonidas Gee
leonidas.gee@studio.unibo.it

Academic Year: 2021 / 2022

Abstract

In this assignment, a fact checking task was solved by training a classifier on a set of sentence embeddings. The claims and their evidence are first cleaned and encoded as sentence embeddings. Finally, a classifier is trained on these embeddings to predict if a claim is supported or rejected. Results show that the best approach achieves a validation AUC of 85% and F1-scores of 71%.

Description

Fact checking is an emerging NLP task that involves the verification of whether a fact is true or not given a set of evidence. For this dataset, a claim is either supported or rejected by its evidence. To solve this binary classification task, a classifier can be trained on claims and evidence that have been encoded as sentence embeddings.

To begin, the claims and evidence are cleaned via the following operations: removal of ID and keywords, lower casing of tokens, removal of special characters with space, removal of left and right spacing. The cleaned text is then tokenized and label encoded. The final tokens are padded to a max length of 100. Stopword removal and stemming was also tested, but produced a lower model performance.

A model architecture takes as input the claim and evidence as a sequence of tokens, with an embedding layer using GloVe embeddings, that then produces sentence embeddings. The sentence embeddings are produced using 4 specific methods:

1. **RNN final state**: the sentence embedding is the last state of an RNN.
2. **RNN average states**: the sentence embedding is the mean of the output states of an RNN.
3. **Multilayer Perceptron**: the sentence embedding is obtained by the output single layer MLP.
4. **Bag of Vectors**: the sentence embedding is the average of the token embeddings.

The claim and evidence embeddings must then be merged by concatenation, summation or averaging. Additionally, the cosine similarity between the claims and evidence are also calculated and concatenated to the embeddings.

Finally, a classifier is trained on the merged sentence embeddings to predict if a claim is supported or rejected. The classifier consists of 3 dense layers with 2 dropout layers in between. L2 regularisation was also applied to the first two dense layers. Dropout and regularisation were employed due to the highly imbalanced nature of the training data. AUC was used as the training metric due to the same reason.

The best sentence embedding is first determined by training and comparing the validation AUC of all 4 methods. The best sentence embedding method is chosen and then used to determine the best

merging method out of the 3 available. Finally, the best model is obtained by taking the merging method that produces the highest AUC on the best sentence embedding method.

The best model is then evaluated on the test set via a multi-input classification evaluation and a claim verification evaluation using majority voting. The non-concatenation of cosine similarity was also tested, but showed lower validation scores.

Tables of Findings

Below are the AUC scores obtained after training the models for 5 epochs, all with the cosine similarity concatenated. While the model were all judged after the same numbers of epochs it is noteworthy that the two models using a RNN took significantly longer to train that the other two.

RNN final state	**RNN averaged states**	**Multilayer Perceptron**	**Bag of Vectors**
0.8368	0.8549	0.7422	0.7009

Table 1: Validation AUC using different sentence embedding methods with concatenation merging.

Concatenation	**Sum**	**Mean**
0.8549	0.8481	0.8543

Table 2: Validation AUC of the best sentence embedding using different merging methods.

The model embeddings with the highest validation AUC score is the RNN with average states using concatenation and cosine similarity appended to the embeddings. The best model achieves a test AUC score of 85%.

Performance evaluation

The model with the highest AUC accuracy was the RNN with avereged states using concatenation and with cosine similarity appended. This model was then evaluated.

	Precision	**Recall**	**F1-score**
REFUTES	0.90	0.50	0.64
SUPPORTS	0.66	0.95	0.77
Macro average	0.78	0.72	0.71

Table 3: Evaluation results of the best model.

The difference of precision and recall between the lables REFUTES and SUPPORTS is because the vast majority of mistakes in the model are misclassification of the label REFUTES that are instead classified as SUPPORTS. This is probably caused by the training data being unbalanced.

The model was also evaluated using majority voting. The results of this evaluation are very similar to the previous one (just slightly better). This is mainly due to the small number of claims with subdivided evidence and also because most of this claims have the same label across all evidences.

Conclusion

In this assignment, we have shown that merged sentence embeddings can be used to train a classifier for fact checking a claim. The RNN with averaged states embedding method, concatenation merging method, and cosine similarity combined, produced the highest test AUC score.