NATURAL LANGUAGE PROCESSING (CS F429)

FIRST SEMESTER: 2021-22 RESEARCH PAPER

Fake News Detection Using Natural Language Processing

Group 38

Name of the Student	ID Number
Mahavir Chaudhari	2019A7PS0088H
Siddhesh Shukla	2019A7PS0099H
Anjel Patel	2019A7PS0126H
Parth Gedia	2019A7PS0151H

Department of Computer Science and Information Systems BITS Pilani Hyderabad Campus

BIRLA INSTITUTE OF TECHNOLOGY AND SCIENCE, PILANI
Hyderabad Campus

Title of the Project: Fake News Detection Using Natural Language Processing

Course: CS F429 NATURAL LANGUAGE PROCESSING

Names/ID Nos./Discipline of the students:

Mahavir Chaudhari	2019A7PS0088H	B.E. CSE
Siddhesh Shukla	2019A7PS0099H	B.E. CSE
Anjel Patel	2019A7PS0126H	B.E. CSE
Parth Gedia	2019A7PS0151H	B.E. CSE

Name of the Faculty: Prof. Aruna Malapati

Abstract:

Fake news detection is a critical yet challenging problem in Natural Language Processing (NLP). The rapid rise of social networking platforms has not only yielded a vast increase in information accessibility but has also accelerated the spread of fake news. Thus, the effect of fake news has been growing, sometimes extending to the offline world and threatening public safety. As the Internet community and the speed of the spread of information are growing rapidly, automated fake news detection on internet content has gained interest in the Artificial Intelligence research community. Given the massive amount of Web content, automatic fake news detection is a practical NLP problem useful to all online content providers, in order to reduce the human time and effort to detect and prevent the spread of fake news.

Date of Submission: 6/12/2021

Acknowledgement

Our immense gratitude to Prof. Aruna Malapati (Associate Professor, Department of Computer Science & Information Systems, BITS Pilani, Hyderabad Campus) for this excellent opportunity to work on the project, mentoring us and providing us with her support and guidance on the issues whenever required. Her help and guidance have been an integral part of the working of the project.

Table of Contents

Introduction

Automated fake news detection is the task of assessing the truthfulness of claims in news. This is a new but critical NLP problem because both traditional news media and social media have huge social-political impacts on every individual in the society. Fake news even relates to real-world violent events that threaten public safety. Fake news has thus proven to be a major threat to democracy, journalism, and freedom of expression. Detecting fake news is an important application in the world that NLP can help with, as it also creates broader impacts on how technologies can facilitate the verification of the veracity of claims while educating the general public. Until recently, the bottleneck for developing automatic methods for fact-checking has been the lack of large datasets for building machine learning models. Wang (2017) has introduced a large dataset (LIAR) of claims from POLITIFACT , the associated metadata for each claim and the verdict (6 class labels). Most work on the LIAR dataset has focused on modeling the content of the claim (including hedging, sentiment and emotion analysis) and the speaker-related metadata **(Wang 2017; Rashkin et al., 2017; Long et al., 2017)**. However, these approaches do not use the evidence and the justification provided by humans to predict the label.

The main hypothesis of this paper is that modeling the extracted justification in conjunction with the claim (and metadata) will provide a improvement both in binary and six-way classification task and BERT which has obtained new state-of-the-art results on eleven natural language processing tasks can sow better results on our chosen dataset.

Link to GitHub Repository:

https://github.com/Siddhesh-Shukla/Fake-News-Detection

Related Work

“A Survey on Natural Language Processing for Fake News Detection Ray Oshikawa”

- by Ray Oshikawa, Jing Qian & William Yang Wang

This paper provides NLP solutions for automatic fake news detection and also categorizes and summarizes available datasets, NLP approaches, and results, providing first-hand experiences and accessible introductions for new researchers interested in this problem. It compares and discusses the most recent benchmark datasets and experimental results of different methods. **Available Datasets :** Fake-news datasets can be categorized into three categories: Claims, Entire Articles, and Social Networking Services (SNS) data. Claims Datasets: Claims are one or a few sentences including information worth validating. POLITIFACT, CHANNEL4.COM, SNOPES are a few such datasets. Entire-Article Datasets: Entire articles are composed of many sentences related to each other constituting information as the whole. FAKENEWSNET consists of headlines and body texts of fake news articles based on BuzzFeed and PolitiFact.BS_Detector searches all links on a web page at issue for references. SNS Datasets: SNS data are similar to claims in length but featured by structured data of accounts and posts, including a lot of non-text data. UZZFEEDNEWS5 collects posts from 9 news agencies on Facebook. Each post is fact checked by 5 BuzzFeed journalists. **Methods :** Preprocessing: It involves tokenization(TF-IDF, LIWC), stemming, and generalization or weighting words. Machine Learning Models: Non-Neural Network Models: SVM and NBC are frequently used classification models. LR and decision trees such as Random Forest Classifier are also used. Neural Network Models: RNN is very popular in NLP, especially LSTM. CNN is also widely used & is also used for extracting features with a variety of meta-data. Rhetorical Approach: RST combined with the VSM is also used for fake news detection. Collecting Evidence: The RTE-based method is frequently used to gather and to utilize evidence. **Results :** LIAR: LSTM based models achieve higher accuracy than CNN based models. FEVER: Attention-LSTM has the best score both of verification and evidence-collection tasks. FAKENEWSNET: The maximum accuracy is achieved by RST and LIWC methods.

“Liar, Liar Pants on Fire”

- by William Yang Wang

This paper presents LIAR: a new, publicly available dataset for fake news detection. It has around 12.8K manually labeled short statements in various contexts from POLITIFACT.COM. These statements are sampled from various contexts/venues, and the top categories include news releases, TV/radio interviews, campaign speeches, TV ads, tweets, debates, Facebook posts, etc. Each statement labeled for truthfulness, subject, context/venue, speaker, state, party, and prior history. This dataset is an order of magnitude larger than previously largest public fake news datasets making it suitable for selecting it as a benchmark for developing machine learning algorithms. It considered six fine-grained labels for the truthfulness ratings: pants-fire, false, barely true, half-true, mostly-true, and true. The distribution of labels in the LIAR dataset is also relatively well-balanced. The author used five baselines: a majority baseline, a regularized logistic regression classifier (LR), a support vector machine classifier (SVM), a bi-directional long short-term memory networks model (Bi-LSTMs), and a convolutional neural network model (CNNs). The Author observed that the majority baseline on this dataset gives about 0.204 and 0.208 accuracy on the validation and test sets respectively. Standard text classifiers such as SVMs and LR models obtained significant improvements. Due to overfitting, the Bi-LSTMs did not perform well and the CNNs outperformed all models. Results also showed that when combining meta-data with text, significant improvements can be achieved for fine-grained fake news detection.

“Where is your Evidence: Improving Fact-checking by Justification Modeling”

- Tariq Alhindi, Savvas Petridis and Smaranda Muresan

Wang (2017) introduced a large dataset of validated claims from the POLITIFACT .com website (LIAR dataset), enabling the development of machine learning approaches for fact-checking. However, approaches based on this dataset have focused primarily on modeling the claim and speaker-related metadata, without considering the evidence used by humans in labeling the claims. Authors extend the LIAR dataset by automatically extracting the justification from the fact-checking article used by humans to label a given claim. Most of the articles end with a summary that has a headline “our ruling” or “summing up”. This summary usually has several justification sentences that are related to the statement. Authors extract all sentences in these summary sections, or the last five sentences in the fact-checking article when no summary exists. They filter out the sentence that has the verdict and related words. These extracted sentences can support or contradict the statement.Their models use 4 different conditions: basic claim/statement representation using just word representations (S condition), enhanced claim/statement representation that captures additional information shown to be useful such as hedging, sentiment strength and emotion **(Rashkin et al., 2017)** as well as metadata information (S + M condition), basic claim/statement and the associated extracted justification (SJ condition) and finally enhanced claim/statement representation, metadata and justification (S + MJ condition). It was clear from the results shown that including the justification (SJ and S + MJ conditions) improves over the conditions that do not use the justification (S and S + M, respectively) for all models, both in the binary (true, false) and the six-way classification tasks (pants on fire, false, mostly false, half true,mostly true, true).

“BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding”

- Jacob Devlin, Ming-Wei Chang, Kenton Lee, Kristina Toutanova

The paper introduced a new language representation model called BERT, which stands for Bidirectional Encoder Representations from Transformers. Unlike recent language representation models, BERT is designed to pretrain deep bidirectional representations from unlabeled text by jointly conditioning on both left and right context in all layers. As a result, the pre-trained BERT model can be finetuned with just one additional output layer to create state-of-the-art models for a wide range of tasks, such as question answering and language inference, without substantial task specific architecture modifications. BERT’s model architecture is a multi-layer bidirectional Transformer encoder based on the original implementation described in **Vaswani et al. (2017)**. Authors pre-trained BERT using two unsupervised tasks: Masked Language Model and Next Sentence Prediction thus understanding language and context. Masked LM(MLM) often referred to as a Cloze task in the literature is a procedure where authors simply mask some percentage of the input tokens at random, and then predict those masked tokens. The final hidden vectors corresponding to the mask tokens are fed into an output softmax over the vocabulary, as in a standard LM. In prior NSP work, only sentence embeddings were transferred to down-stream tasks, where BERT transfers all parameters to initialize end-task model parameters. For the pre-training corpus authors used the BooksCorpus (800M words) and English Wikipedia (2,500M words). Fine-tuning is straightforward since the self-attention mechanism in the Transformer allows BERT to model many downstream tasks by swapping out the appropriate inputs and outputs.

“Sentimental LIAR: Extended Corpus and Deep Learning Models for Fake Claim Classification”

- *Bibek Upadhayay, Vahid Behzadan*

The paper starts with mentioning the high volume and velocity of information flow in such platforms make manual supervision and control of information propagation infeasible. This paper aims to address this issue by proposing a novel deep learning approach for automated detection of false short-text claims on social media. It proposes a novel deep learning architecture based on the **BERT-Base** language model for classification of claims as genuine or fake. The results demonstrate that the proposed architecture trained on Sentimental LIAR can achieve an accuracy of 70%, which is an improvement of 30% over previously reported results for the LIAR benchmark. Over the past few years, a number of datasets and models have been proposed for the classification of short-text claims, notable instances of which are the studies based on the LIAR dataset of short statements. However, the performance of machine learning models trained on this dataset remain at impractical levels, with the best accuracy values reported to be 41.5%. In this paper, authors introduce Sentimental LIAR, which extends the LIAR dataset by including new features based on the sentiment and emotion analysis of claims. The LIAR dataset is extended by adding emotions anger, sadness, fear, anger and disgust by using IBM NLP API and added sentiment score using Google NLP API. We also included speaker credit as an input attribute to our models. The experiments performed with BERT-Base + feedforward NN, the accuracy ranged from 68.8% to 69% within the five experiments. The experiments performed with BERT-Base + CNN, the accuracy ranged from 68.82% to 70% within six experiments, and also major improvements were observed in the F1 Score (0.5308 to 0.6430).

Datasets

Brief description of some shortlisted datasets we found on the web, mainly:

1. FEVER: dataset providing related evidence for fact-checking. In this point, it is similar to EMERGENT. Fever contains 185,445 claims generated from Wikipedia data. Each statement is labeled as Supported, Refuted, or Not Enough Info.
2. PHEME: It contains 330 twitter threads (a series of connected Tweets from one person) of nine newsworthy events, labeled as true or false.
3. LIAR: It has around 12.8K manually labeled short statements in various contexts from POLITIFACT.COM. It considered six fine-grained labels for the truthfulness ratings: pants-fire, false, barely true, half-true, mostly-true, and true.
4. LIAR Plus: The extended LIAR dataset for fact-checking and fake news detection. It contains an additional column “The Extracted Justification”. It gives significantly better accuracy as compared to LIAR.
5. Sentimental LIAR: It extends the LIAR dataset by including new features based on the sentiment and emotion analysis of claims. Results demonstrate that Sentimental LIAR can achieve better accuracy as compared to LIAR.

We also had options in choosing the datasets, namely FEVER, PHEME, LIAR PLUS, LIAR, LIAR Sentimental. Out of these datasets, LIAR PLUS provided the most value as it has justification data which is beneficial for the BERT language model. We experienced that LIAR PLUS and LIAR Sentimental are Significantly better than the rest with size and accuracy, in fact even with time. LIAR PLUS is 30% better at binary classification of the news articles into true or not true and significantly better than the state of art LIAR dataset. LIAR Sentimental was similarly better but had a different method of approach for solving the main problem. It focuses on sentiment analysis along with fact-checking and adding some extra features for the same in the dataset. We finally chose the LIAR PLUS Dataset and a Tri-BERT Siamese Model.

We decided to use LIAR plus to prove our hypothesis and improve the results on this dataset when compared to those mentioned in the research paper.

Approach / Methodology

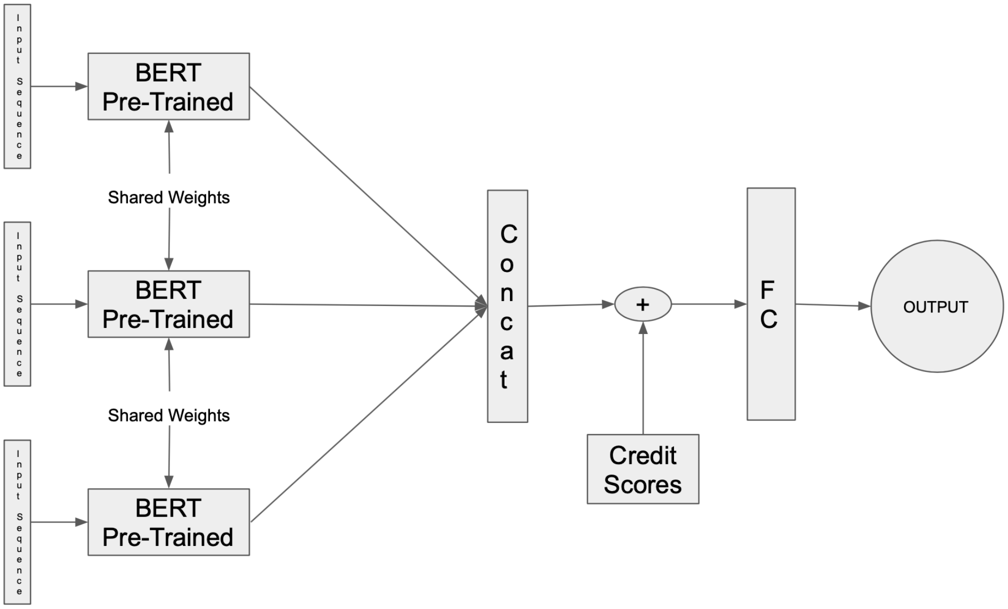
After looking at multiple datasets and reviewing many papers, we found out that methods like Logistic regression and Random Forests work good on datasets similar to LIAR (datasets like FNC1[9] and COVID-19 Fake News Dataset[10]). But we found out that none of these methods performed well on our test data. This suggests that most of the models just overfit the whole dataset and did not learn anything.

Additionally, our LIAR PLUS dataset provides output corresponding to the number of barely true counts, false counts, half true counts, mostly true counts, pants on fire counts made by the news source. Hence we implemented a credit score system as suggested by the paper[3].

$$\textbf{Credit Score} = \frac{\text{(mostly true counts)*0.2 + (half true counts)*0.5 + (barely true counts)*0.75 + (false counts)*0.9 + (pants on fire counts)*1}}{\text{mostly true counts + half true counts + barely true counts + false counts) + pants on fire counts]}}$$

The credit score tells us about how false or fake the news published by that author or the source is on average.

We wanted our model to capture the meaning of a sentence and upon reading more about how to do that we discovered that we needed a language model. BERT is a good language model and is still considered state of the art and we found an interesting implementation that uses BERT. We decided to use this approach, the google's pretrained BERT model using the library pytorch-pretrained-bert from huggingface.co and parallelly creating 3 instances of the same. This is used to run a tri-bert siamese model.



Final Architecture

Evaluation Method/Metric

Built a siamese network with three branches with each branch containing BERT as the base models. Input of the first branch will be the tokens corresponding to the news statements on which we need to predict the labels. Input of the second branch will be the tokens corresponding to the justification of the particular news statement passed to the first branch. The input to the third branch will be the remaining meta data available like speaker, source, affiliation, etc. apart from the justification. In this architecture, both branches share the same weights between them. The output of each BERT layer branch will be a 1D tensor of shape (768). These three outputs are concatenated and passed through a Linear layer(fully connected layer), from this we get two logits and a ‘softmax’ activation is applied to get the output probabilities. We Trained and Saved the model for 8 epochs, with noting down loss and accuracy with train and validation at each epoch, the same is shown in the graphs below.

Experimental setup

We used the pytorch-pretrained-bert python library in order to use the pretrained BERT model developed by google. This is a type of transfer learning and allows us to create a representation of our input text and further fine tune the model for our specific purpose. We experimented with different Models namely, BERT-fine tuning model, dual-BERT Siamese, Tri-BERT Siamese, 4-Parallel-BERT Siamese Model, and we found Tri-BERT to be best model among the rest in terms of training time and accuracy.

We also experimented with different Datasets, explained above in the dataset section, namely FEVER, PHEME, LIAR PLUS, LIAR, LIAR Sentimental. Out of these datasets, LIAR PLUS provided the most value as it has justification data which is beneficial for the BERT language model.

We use the “Focal Loss” loss function which is an improvement over the classical Cross Entropy loss function. Our focal loss is set with alpha of 0.5 and gamma of 2.0. We set the learning rate scheme to decaying learning rate starting from 0.001 and ending at 0.00001.

BERT:

In terms of hyper-parameters, we initialized our BERT model with stochasticity(epsilon) of 1 x e^{-12}. We used a batch size of 16 considering that we used a GPU for training. Our BERT takes a 32000 sized input(tokenized vector). It has a output layer size of 768 and an intermediate layer size of 3072. These were the hyperparameters that we found worked best and these were also suggested to work good in our reference paper[3]. There are 12 layers in the BERT model and 12 attention heads.

LINEAR:

After we get the three outputs from our BERT models, we concatenate them into one single vector of size 768*3. These go directly into our linear classifier which outputs either 2 nodes or 6 nodes depending on the way we classify our data(LIAR PLUS provides an option for 6 way classification apart from the standart 2 ways binary classification).

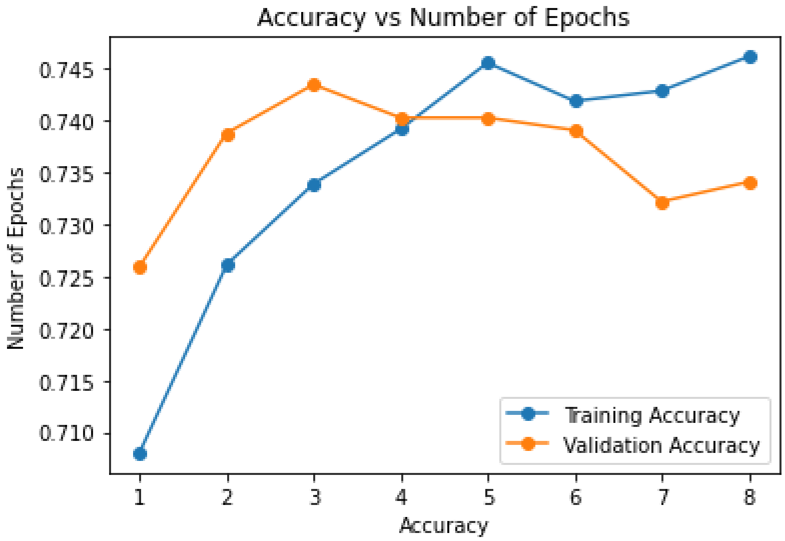
For our input, we are using different sequence sizes for each of the sentences (4 sentences). Our final model uses 3 identical BERT branches in a shared weight siamese fashion. Our code is written in a separate repository. We tried training on our own machine but due to compute limitations, we imported the code into Google Colab where we can use Colab’s GPU.

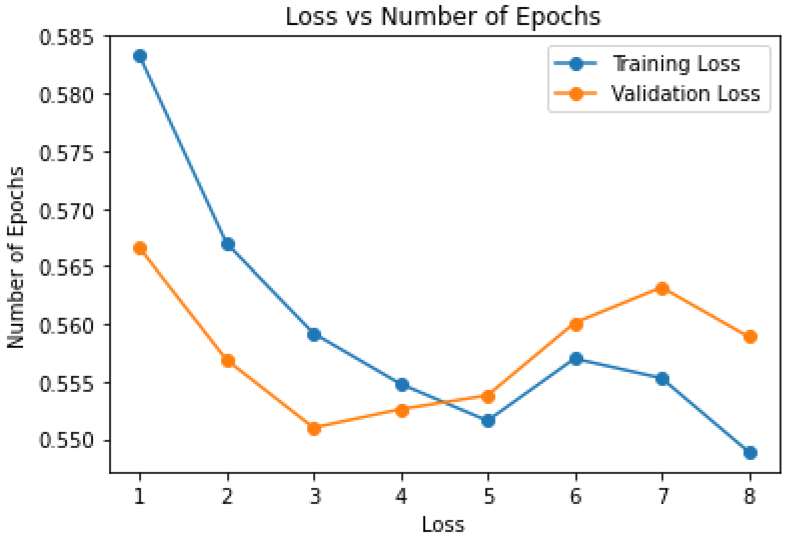
Results and discussion:

After choosing the final dataset to be LIAR PLUS (variant of LIAR, explained above), and the tri-bert siamese model.

We observe peak accuracy at 5th epoch at once, then at 8th for training data FOR 6 way classification and 9th epoch for 2 way classification. as can be seen in the graphs below.

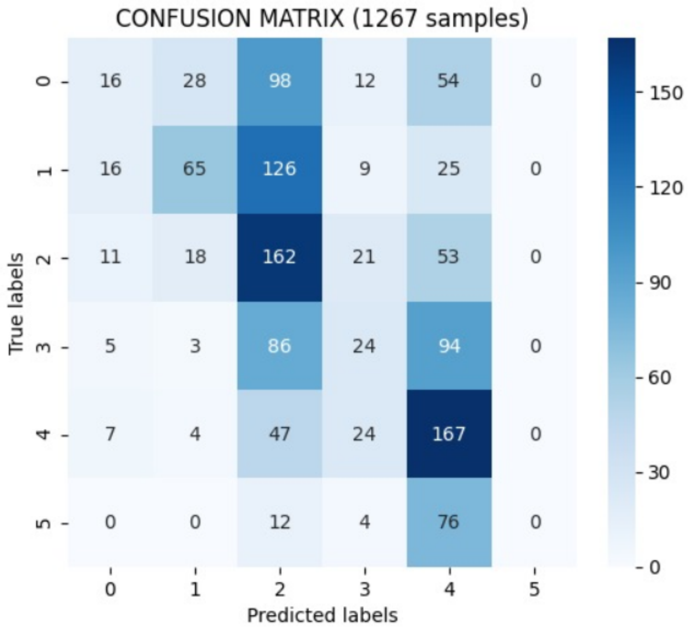
This is a significant improvement from other models of Dual-BERT and BERT with fine tuning.





We can observe the confusion matrix in 2 ways and 6 ways classification. We observed a significant improvement from other datasets and models.

Test Confusion matrix for 6-way classification



Precision : 0.329046

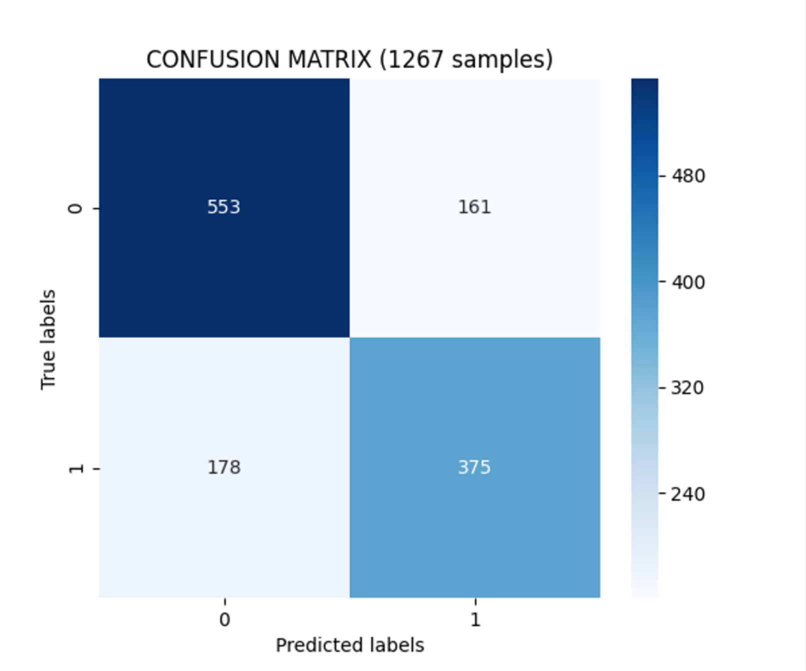
Recall : 0.342541

F1 Score : 0.291655

AUC-ROC Score :
0.590121

Accuracy : 34.2541

Test Confusion Matrix for Two way Classification



Precision : 0.731676

Recall : 0.732439

F1 Score : 0.731924

AUC-ROC Score :0.726315

Accuracy : 73.2439%

The BERT model showed accuracy of 73.24% on binary classification which is around 7 percent more than the LR model mentioned in the paper **(Tariq Alhindi et. al. 2018)** and 34.25% on six-way classification which is close to the highest accuracy mentioned in the same paper but much more than state of the art accuracy on the LIAR Dataset (24.8%achieved using Hybrid CNNs).

Conclusion

We presented a study that shows that modeling the human-provided justification from the fact checking article associated with a claim is important leading to significant improvements when compared to modeling just the claim/statement and metadata both in a binary and a six-way classification task. We used LIAR-PLUS, the extended LIAR dataset that contains the automatically extracted justification. We also provided an error analysis and confusion matrix of per-class performance. BERT showed improvements in binary classification when compared to Logistic Regression mentioned in the paper **(Tariq Alhindi et. al. 2018)** by a noteworthy margin.

The simple method for extracting the justification from the fact-checking article leads to slightly noisy text (for example it may contain a repetition of the claim or it may fail to capture the entire evidence).To further improve the results we can refine the justification extraction method so that it contains just the summary evidence.

References

1. Ray Oshikawa, Jing Qian, William Yang Wang 2018 A Survey on Natural Language Processing for Fake News Detection
2. William Yang Wang 2017. "Liar, Liar Pants on Fire": A New Benchmark Dataset for Fake News Detection
3. Tariq Alhindi, Savvas Petridis and Smaranda Muresan 2018. Where is your Evidence: Improving Fact-checking by Justification Modeling
4. BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding
5. Sentimental LIAR: Extended Corpus and Deep Learning Models for Fake Claim Classification
6. Ashish Vaswani, Noam Shazeer, Niki Parmar et. al. 2017. Attention is All you Need
7. Hannah Rashkin, Eunsol Choi, Jin Yea Jang et. al. 2017. Truth of varying shades: Analyzing language in fake news and political fact-checking.
8. Yunfei Long, Qin Lu, Rong Xiang, Minglei Li et. al. 2017. Fake news detection through multi-perspective speaker profiles.
9. https://github.com/FakeNewsChallenge/fnc-1.git
10. https://paperswithcode.com/dataset/covid-19-fake-news-dataset