REPORT

Binary Classification:

Binary Test Accuracy-64.9

Model : Using a CNN for text classification with additional Auxiliary branch for metadata features

Feature Selection: According to the results mentioned in the paper, we can see that using only statement provides a test score about 0.61 and using statement(plus) + meta data provides a max test score of 0.62 . From these results it can be intuited that statement alone carries the bulk of the score and metadata marginally helps it to improve the score. When justification is added along with the statement , it increases the accuracy to .70 for binary classification using a LSTM model. However, in my case only adding the justification with statement did not improve the score by much.

So in my work, I used Statement,Justification,and Metadata. Metadata variables were selected after several exploratory data analysis of the Liar Plus dataset. There are 11 meta variables Namely: subjects,speaker,speakers's job,state,party,venue,barely true counts ,false counts ,half true counts,mostly true counts pants on fire counts.

For variables: [barely true counts, false counts,half true counts,mostly true counts,pants on fire counts] , I used them directly with feature wise normalization.
For rest of the meta variables. I categorized them into discrete classes and a 'rest' class and later one-hot-encoded them.

Details of meta classes:
Speakers-Top 20 speakers based on value counts
Jobs-Top 10 based on value counts
Party-Top 5 based on value counts
States-50 states
Subject List- Top 12 subjects based on value counts
Venues-Top 10 venues based on value counts

All discretization was done manually and for some instances like 'TV interviews' and 'Television interviews' were mapped to the same class 'TV'

For sentences I used pretrained 300 dimensional Glove Embeddings trained on 840 billion text, to extract text features

Experiments while training: To begin with , I first used only statements which yielded me a score around 60%. Next I included the metadata which boosted my accuracy to 63%. Later, I included justification by adding it with statement which helped me increase my test accuracy to 64.9%.Initially, I directly concatenated text features from CNN with Meta features and passed it to final softmax layer as mentioned in the paper but that decreased the performance. So , it was important that meta features are first passed through an intermediate fully connected linear layer before concatenating with Text features. I also used step learning rate decay policy which was found to be important. Using a constant learning rate yielded me around 63.5-64.1%. I also experimented with cosine learning rate decay policy which though reduced overfitting but didn't increase test accuracy much so i kept with step lr decay .

Training hyperparameters: Embedding Dimension-300,
batch_size-48,dropout_value-0.7,kernel size for different CNN layers-[2,5,8] , lr-0.025
The training hyperparameters were experimented with immensely and this is the best set of hyperparameters which I found to work well. The model is so sensitive to these that if we change any of the one, the performance would drop considerably. I attribute this to the lack of data and so the model overfits very easily.

Other experiments I tried for binary classification : Used Logistic Regression, Bi-LSTM models, SVM,Gradient Boosting Classifier, Feature selection strategies for classifiers except Bi-LSTM, Unigram features,Bigram features,tf-idf features .

Remark: Logistic Regression is found to be the most stable classifier even for logistic regression. For binary classification with Statement+Justification+Metadata and using Glove embeddings(100d), a test accuracy of 64.40% is achieved. The training accuracy was 64.76 and a val accuracy of 62 ,which indicates that in this case more data will improve the scores.

Multi Class Classification-

Best Test Accuracy-31.6

Model- Logistic Regression with Statement+Justification+Metadata

Feature Selection- I used the same strategy as before for the metadata variables. For text data, I simply added the justification to its original statement. I used 100 dimensional glove embeddings to extract text features.

Experiments while training: For this experiment,I initially tried with unigram features for Statement only with different classifiers: Logistic Regression,SVM,Gradient Boosting Classifier, Naive Bayes.Logistic Regression worked out to be the best with a value around 24.7. I added justification and metadata features along with statement features but didn't find any major improvements. I attribute this behaviour to high dimensional feature spaces and less data . So I dropped the idea of using unigram features and used text features extracted from 100 dimensional glove word embeddings. I concatenated statement and justification and extracted text features from it . These features were appended with metadata features. Logistic Regression yielded a test accuracy of 31.57 with a validation score about 32.6. I a quest to understand how much effect does meta variables : Speakers,Job,Party, State,Subject,Venues had, I removed them from metadata features and got a score around 60. However, I did not experiment much with other hyperparameters in this setting so this value could be even higher.

Other experiments I tried for multiclass classification- Feature selection strategies for classifiers, PCA,tf-idf features,Bigram features, Bi-LSTM, CNN, ensembling different models like Logistic Regression,SVM,Decision Trees but didn't find much improvement so stuck to Logistic Regression.

Remark: CNN model used for binary classification as described above when applied to multi class classification yielded a test score of 28.3. LSTM and CNN model perform similarly with CNN marginally outperforming LSTM models.

References-

1.https://arxiv.org/abs/1408.5882
2.https://github.com/AnubhavGupta3377/Text-Classification-Models-Pytorch
3.https://github.com/vivek1729/fake-news-classifier
4.https://scikit-learn.org/
5. https://spacy.io/
6. https://pytorch.org/
7. https://torchtext.readthedocs.io/en/latest/
8. https://www.nltk.org/