Name : RESHMA MANOJ KUMAR
CWID : 20007266
Course : CS 541 – Artificial Intelligence
Email : rmanojku@stevens.edu

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

Deliverable - 3

Experiments & Results

Abstract :

The rampant integration of social media in our every day lives and culture has given rise to fast and easier access to the flow of information than ever in human history. However, the inherently unsupervised nature of social media platforms has also made it easier to spread false information and fake news. Furthermore, 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. We first introduce Sentimental LIAR, which extends the LIAR dataset of short claims by adding features based on sentiment and emotion analysis of claims. Furthermore, we propose a novel deep learning architecture based on the BERT-Base language model for classification of claims as genuine or fake. Our 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.

Tools and Technologies :

IDE used to run the code : Visual Studio Code – 1.63.0 (ipykernel)

Import using pandas, numpy.

DATA :

Dataset : LIAR dataset

Fake claim detection in short text has yielded a number of significant open-source datasets, some of the most notable of which are enumerated as follows:

1) FEVER [14]: Fact Extraction and VERification (FEVER) is a short claim dataset of 185,445 claims. This dataset is annotated with three labels: *Supported, Refuted,* and *Not Enough Info*. The claims are curated from Wikipedia. FEVER was constructed in two stages, the first one is Claim Generation, where extracted information Wikipedia is con- verted into claims. The

second stage is Claim Labeling, where each claim is labeled as 'supported' or 'refuted' according to Wikipedia. In cases where information were insufficient for determination, a label of *not enough information* is assigned.

2) PHEME [12]: is a dataset of 330 rumor threads com- posed of 4843 tweets associated with 9 newsworthy events. The annotators of this dataset were journalists who tracked the events in real time. Each entry in PHEME is labeled as either true or false.

3) LIAR [7]: is a publicly available short statement dataset that is derived from Politifact.com. Each of the 12,836 statements in LIAR is annotated based on data available on Politifact with one of the following six labels: pants-fire, false, barely true, half-true, mostly-true, and true.

The dataset contains the text of a claim, as well as relevant meta-data, structured as follows : ID, LABEL, STATEMENT, SUBJECT, SPEAKER, SPEAKER JOB, STATE INFO, PARTY AFFILIATION, BARELY TRUE COUNTS, FALSE COUNTS, HALF TRUE COUNTS, MOSTLY TRUE COUNTS, PANTS ON FIRE COUNTS, and CONTEXT.

Authors : Bibek Upadhayay, Vahid Behzadan.

Data is preprocessed.

CODE :

```
train_size = 1
train_dataset=df.sample(frac=train_size,random_state=200).reset_index
(drop=True)
test_dataset=df_test.sample(frac=train_size,random_state=200).reset_index
(drop=True)
valid_dataset=df_valid.sample(frac=1,random_state=200).reset_index
(drop=True)
```

OUTPUT :

```
FULL Dataset: (10232, 33)
TRAIN Dataset: (10232, 33)
TEST Dataset: (1264, 32)
VALID Dataset: (1280, 33)
```

METHODOLOGY :

Fake claims are often written in a style of exaggerated expressions and strong emotions. Style-based classification studies aim to assess news intention. The fake claims are written with an
intention to convince the audiences to read and trust the claims, for which fake claims are written with different styles and different sentiments and emotions. With the aim of developing a computationally feasible model for fake claim detection, we propose deep neural network architectures based on BERT- Base to analyze the deception in short-text claims. Our proposed models learn to detect deception based on attribute- based language features, such as sentiments and structure- based language features. In our approach, we rely on the representation learning capabilities of transformers to extract features from statements.

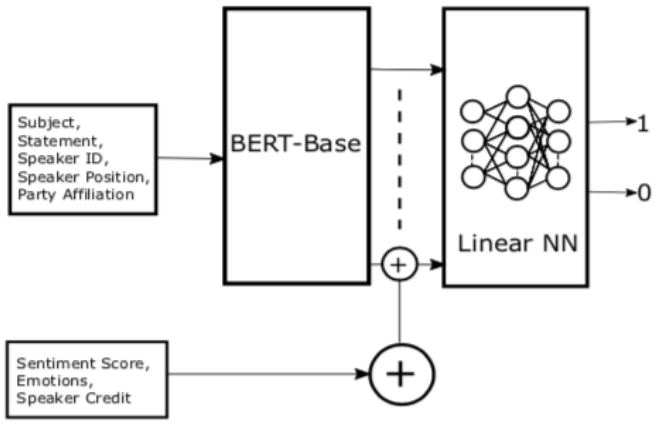
We also propose an extension to the LIAR dataset that includes additional features based on the sentiment and emotion analysis of the claim.

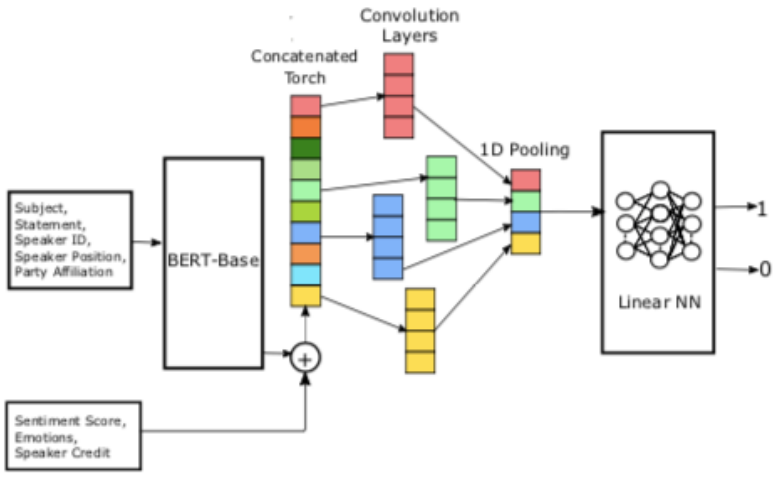
EXPERIMENTAL RESULTS :

The dataset was split into 80% train set, 10% valid set, and 10% test set. The train batch size a nd the test batch size were set to 8, with a learning rate of 1e-05. The BERT-Base was configured with a dropout of 0.3 and model used Sigmoid as its activation function. The loss function used in both architectures was the binary cross entropy loss optimized using the Adam optimizer.

Convolutional Neural Networks (CNNs) are generally known for their applications in image processing, where CNNs can learn representations of localized features from raw input while preserving the relationship between them. However, this capability of CNNs is not constrained to computer vision, and can also be adopted for NLP tasks.

MODEL :





Below demonstrates a sample record in Sentimental LIAR for a short claim in the LIAR dataset :

```
statement="McCain opposed a requirement that the government
buy American-made motorcycles. And he said all buy-American
provisions were quote 'disgraceful.' "
subject: federal-budget
speaker id: 2
speaker job: President
state info: Illinois
party affiliation: democrat
sentiment: NEGATIVE
anger: 0.1353
disgust: 0.8253
sad: 0.1419
fear: 0.0157
joy: 0.0236
barely true counts: 70
false counts: 71
half true counts: 160
mostly true counts: 163
pants on fire counts: 9
SEN sentiment score: -0.7
```

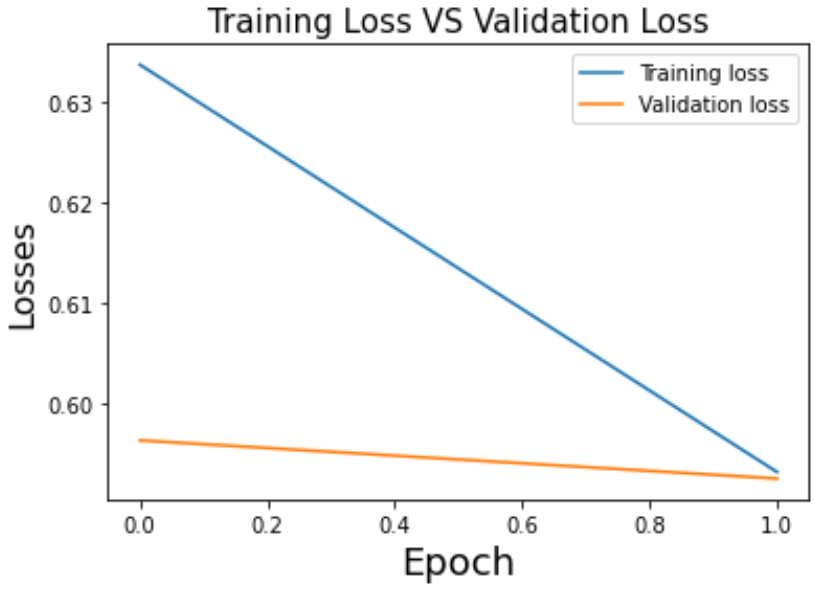
Experiment

This includes the training details (epochs, learning rate, dropout etc.)

```
Epoch: 0  is Started:
   Epoch: 0 Train loss is :0.6337657139998474
   Epoch 0 took: 0:06:17

Epoch: 0 - Accuracy on Testing Data Score = 0.6693037974683544
Epoch: 0 - F1 Score on Testing Data (Micro) = 0.6708860759493671
Epoch: 0 - F1 Score on Testing Data (Macro) = 0.6011597099211216

         Epoch 0 : Train Loss (Training Data):0.6337657139998474, Validatio
n Loss (Testing Data): 0.5962660192693554
_________________________________________________

         Epoch: 1  is Started:
   Epoch: 1 Train loss is :0.5931087791849804
   Epoch 1 took: 0:06:16

Epoch: 1 - Accuracy on Testing Data Score = 0.6914556962025317
Epoch: 1 - F1 Score on Testing Data (Micro) = 0.6958579881656805
Epoch: 1 - F1 Score on Testing Data (Macro) = 0.6538011695906432

         Epoch 1 : Train Loss (Training Data):0.5931087791849804, Validatio
n Loss (Testing Data): 0.5924532325387751
_________________________________________________
```

Result

The plot between Losses and Epoch.




```
Epoch:  1, Accuracy Score on validation data = 0.69921875
Epoch:  1, F1 Score on Validation Data (Macro) = 0.6430236972753443
```

Problems / Issues

Issues were faced when deploying certain packages and importing packages. It threw various exceptions.