CS499-MAJOR PROJECT

Hate,did you fake it?

Deeplearning model with multitasking for the detection of fake context and hate news from code-mixed speeches

GROUP MEMBERS

Harsha Vardhan Vytla	18bcs118
Pokala Dattatreya	18bcs067
Rama Dundi Saketh	18bcs076
Sakala Sampath	18bcs087

MENTOR

Dr.SUNIL SAUMYA
Asst Professor
Dep of Computer Science
Indian institute of Information Technology,Dharwad

Dharwad
ज्ञानेन विकासः

INDIAN INSTITUTE OF
INFORMATION
TECHNOLOGY

Contents

1 INTRODUCTION

1.1 Context

According article-19-1a of Indian constitution every one has right to express their views and opinions at any issue through any medium.
But it is not an absolute right because of the restrictions on it.
It was restricted where the subject related to Security of state,Foreign relations,Defamation,Contempt of court,Incitement to an offence,Sovereignty & integrity of state.

Hate Speech

Initially hate speech is not defined in any law in india.
But later law commission of India on 23 March,2017 in its 267th report stated that "Hate speech is any word written or spoken,signs,visible representations within the hearing or sight of the person with the intension to cause fear or alarm,or incitement to violence".
Hate speech can be derived as any sort of communication which derogates any person or a community based on nationality, race, caste, gender, ethnicity etc.

- Tell those idiots to go hell!
- That women should mind her own business.

Fake News

Fake news are the news which are not based on truths and facts.
News plays an important role in our day to day life and that is why fake news can create a significant problem to our society.
Few examples of fake news are

- Virat is dropped from T20 world cup.
- Ten rupee coin is no more a legal tender.

1.2 Problem Statement

Detection of fake,hate news that are widely spreading these days by people using social media platforms and other networks to sharing their views on many topics.while sharing their views,which includes fake,hate new in gathering more attention towards it which effects many individuals or organizations.

1.3 Motivation

With the advent of social-media the ease of publishing and distributing news increased over the years.As a result fake news and hate speeches spreading also increased drastically.
The laws that exist or not powerful enough to stop hate content spreading through web and of-course it is tough to identify & catch people behind it.With

the increasing volumes of such articles and posts it has become difficult to identify fake news and hate speeches manually and that is where Machine Learning and NLP has come to our aid.
Hindi is the fourth most spoken language in the world and of these people most of the speakers are from India and english is the 2nd most speaking language in India.
Thus, being a country with Hindi,English most spoken hate and fake news spreading widely in Hinglish and that need to be tackled.

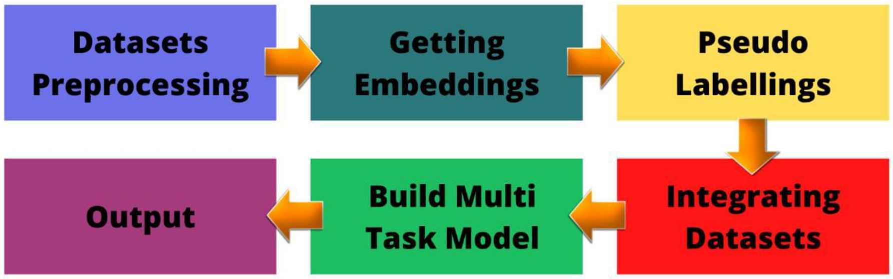
1.4 Task Description

Multi-label classification involves predicting zero or more class labels. Unlike normal classification tasks where class labels are mutually exclusive, multi-label classification requires specialized machine learning algorithms that support predicting multiple mutually non-exclusive classes or labels.
The multi-label classification models require dataset labelled with all required classes.
if the dataset with multi-class labels not found then need to collect data for each class of data and then need to generate pseudo-labels for one other.

2 PRACTICALITY OF IMPLEMENTATION

- We implement our model using python language.
- GPU is needed for practical Implementation.
- We use mBert,XLM-R from hugging face to get embeddings.
- We use LSTM ,Bi-LSTM with attention mechanism and BERT models from hugging face.

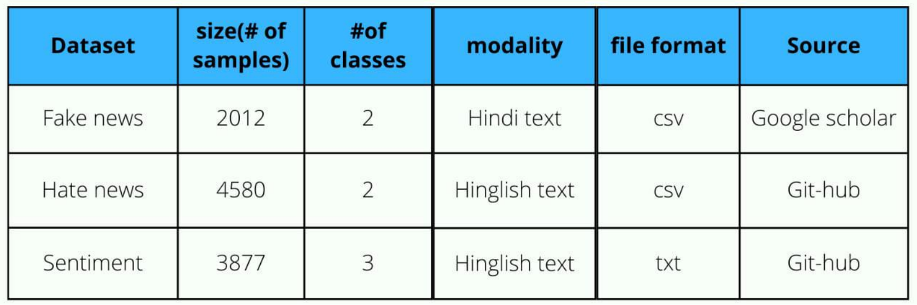
3 PROPOSED METHADOLOGY



4 DATA COLLECTION

Datasets were not found for code-mixed hate speeches labelled for fake,hate, sentiment classes.

Then started finding datasets for each class of data required individually and finally found datasets for all three required classes in various platforms.
Description of datasets shown in the table below.



Dataset	size(# of samples)	#of classes	modality	file format	Source
Fake news	2012	2	Hindi text	csv	Google scholar
Hate news	4580	2	Hinglish text	csv	Git-hub
Sentiment	3877	3	Hinglish text	txt	Git-hub

5 DATA PREPROCESSING

5.1 Data Transliteration

Code-mixed data which we found for fake data is in Hindi text.so inoder to make all the documents being processed to be in same language(english text) it was transliterated using Indic NLP.
Transliteration is simply the process of converting fields or characters from one alphabet to another without keeping the underlying meaning using Indic NLP library which is used to transliterate Indian languages.

5.2 Data Transformation

Code-mixed data which we found with sentiment labels is in txt file format.it was transformed into csv format.thus,dataset will be structured and accessible for further process.

5.3 Dataset Balancing

The datasets found for fake,hate are imbalanced.
Imbalanced data refers to those types of datasets where the target class has an uneven distribution of observations, i.e one class label has a very high number of observations and the other has a very low number of observations.
Generally for imbalanced datasets makes any algorithm tends to biased towards majority class and minimizes the accuracy.
So to maximize accuracy we used SMOTE oversampling technique to balance the datasets.

Dataset	BEFORE	AFTER
FAKE	(0,761)(1,1250)	(0,1250)(1,1250)
HATE	(0,2918)(1,1661)	(0,2918)(1,2918)

Size of Datasets before and after balancing

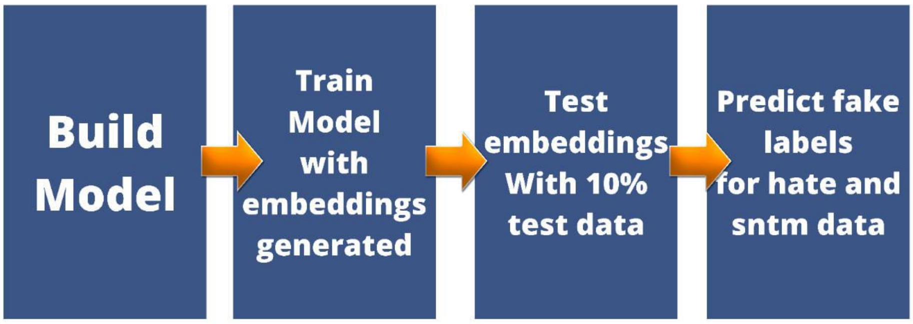
6 Embeddings & Pseudo labelling

6.1 Embeddings

Embeddings for all three datasets have been generated using mBert and XLM-R based sentence transformers.
These sentence transformers maped sentences to 768 dimensional vector space. Thus we get shape of embeddings like **(no:of sent,768)** and embeddings of type **numpy**.
Then averaged the embeddings generated by both the transformers for each class of datasets.
And finally reduced dimensions of embeddings by passing it through dense layers to **(no:of sent,200)**.

6.2 Pseudo Labelling

For each class of datasets pseudo labels have been generated for the other two classes.
Ex:
To generate fake labels for hate and sentiment class datasets we using LSTM model by passing embedding of fake dataset generated before.


Build Model
Train Model with embeddings generated
Test embeddings With 10% test data
Predict fake labels for hate and sntm data

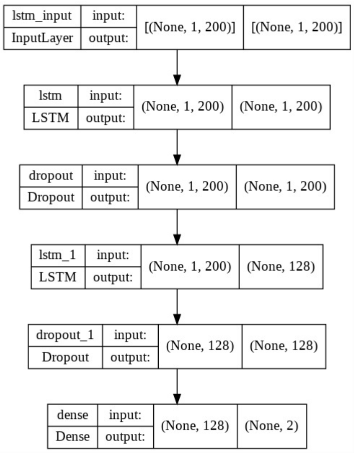
And the same process repeated for hate labels with sentiment,fake datasets and for sentiment labels with hate,fake datasets.

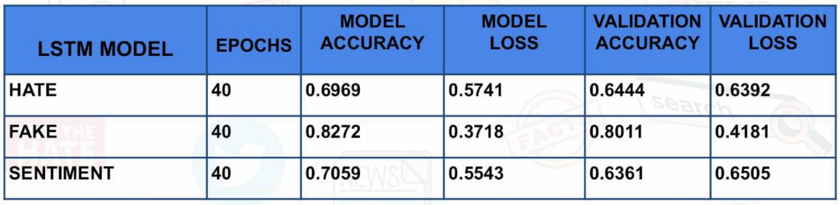
LSTM

- Long Short-Term Memory (LSTM) networks are a type of recurrent neural network capable of learning order dependence in sequence prediction problems.

- **Steps**

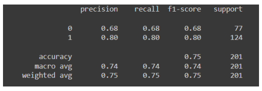
  1. Clean the data
  2. Get Embeddings
  3. We split the data as 90
  4. Train the model $(Epochs : 40, Softmax, Embeddings)$
  5. Test the model
  6. Predict the outputs for hate and sentiment data sets.


lstm_input | input: | [(None, 1, 200)] | [(None, 1, 200)]
InputLayer | output:
lstm | input: | (None, 1, 200) | (None, 1, 200)
LSTM | output:
dropout | input: | (None, 1, 200) | (None, 1, 200)
Dropout | output:
lstm_1 | input: | (None, 1, 200) | (None, 128)
LSTM | output:
dropout_1 | input: | (None, 128) | (None, 128)
Dropout | output:
dense | input: | (None, 128) | (None, 2)
Dense | output:

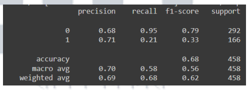


LSTM MODEL	EPOCHS	MODEL ACCURACY	MODEL LOSS	VALIDATION ACCURACY	VALIDATION LOSS
HATE	40	0.6969	0.5741	0.6444	0.6392
FAKE	40	0.8272	0.3718	0.8011	0.4181
SENTIMENT	40	0.7059	0.5543	0.6361	0.6505

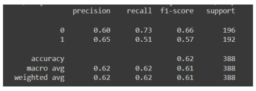
- classification report and confusion matrix for fake,hate,sentiment.



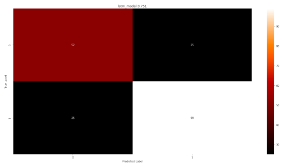
```
              precision    recall  f1-score   support

           0       0.68      0.68      0.68        77
           1       0.80      0.80      0.80       124

    accuracy                           0.75       201
   macro avg       0.74      0.74      0.74       201
weighted avg       0.75      0.75      0.75       201
```

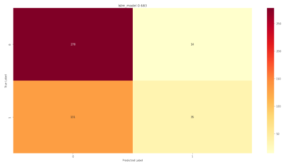


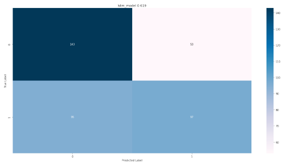
```
              precision    recall  f1-score   support

           0       0.68      0.95      0.79       292
           1       0.71      0.21      0.33       166

    accuracy                           0.68       458
   macro avg       0.70      0.58      0.56       458
weighted avg       0.69      0.68      0.62       458
```



```
              precision    recall  f1-score   support

           0       0.60      0.73      0.66       196
           1       0.65      0.51      0.57       192

    accuracy                           0.62       388
   macro avg       0.62      0.62      0.61       388
weighted avg       0.62      0.62      0.61       388
```

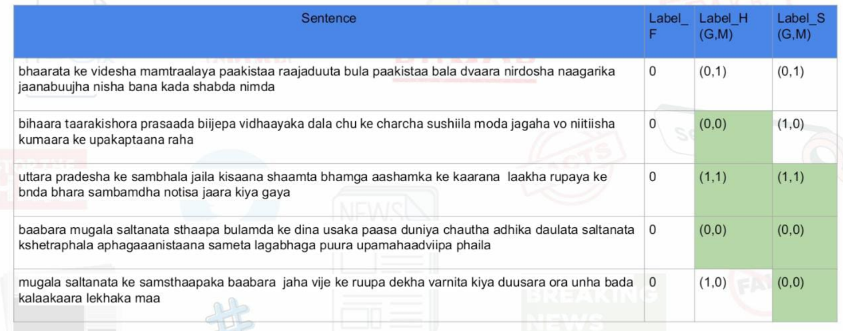






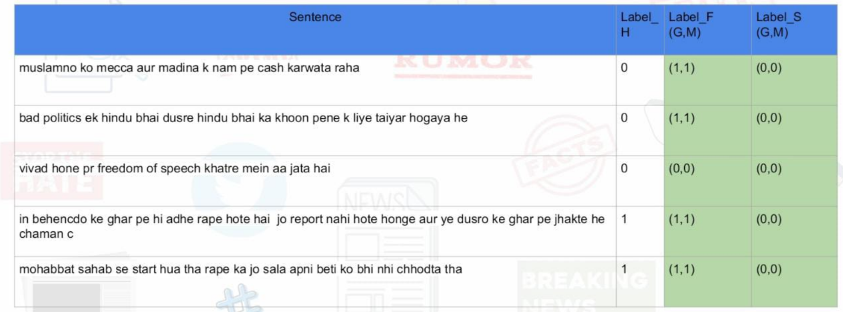
- Analysing pseudo labels generated for each class of dataset manually(m) with model generated(g) labels.

- Hom many of the analysed labels were matched to model generated labels were highlighted in below analysis table.



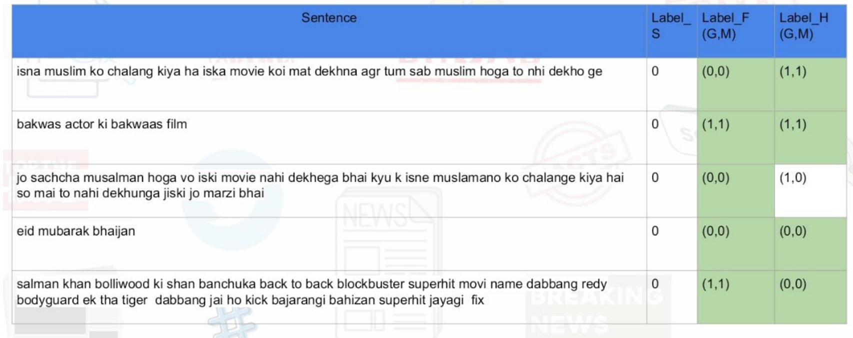
Sentence	Label_ F	Label_H (G,M)	Label_S (G,M)
bhaarata ke videsha mamtraalaya paakistaa raajaduuta bula paakistaa bala dvaara nirdosha naagarika jaanabuujha nisha bana kada shabda nimda	0	(0,1)	(0,1)
bihaara taarakishora prasaada biijepa vidhaayaka dala chu ke charcha sushiila moda jagaha vo niitiisha kumaara ke upakaptaana raha	0	(0,0)	(1,0)
uttara pradesha ke sambhala jaila kisaana shaamta bhamga aashamka ke kaarana laakha rupaya ke bnda bhara sambamdha notisa jaara kiya gaya	0	(1,1)	(1,1)
baabara mugala saltanata sthaapa bulamda ke dina usaka paasa duniya chautha adhika daulata saltanata kshetraphala aphagaaanistaana sameta lagabhaga puura upamahaadviipa phaila	0	(0,0)	(0,0)
mugala saltanata ke samsthaapaka baabara jaha vije ke ruupa dekha varnita kiya duusara ora unha bada kalaakaara lekhaka maa	0	(1,0)	(0,0)

Analysing pseudo labels hate,sentiment over fake dataset



Sentence	Label_ H	Label_F (G,M)	Label_S (G,M)
muslamno ko mecca aur madina k nam pe cash karwata raha	0	(1,1)	(0,0)
bad politics ek hindu bhai dusre hindu bhai ka khoon pene k liye taiyar hogaya he	0	(1,1)	(0,0)
vivad hone pr freedom of speech khatre mein aa jata hai	0	(0,0)	(0,0)
in behencdo ke ghar pe hi adhe rape hote hai jo report nahi hote honge aur ye dusro ke ghar pe jhakte he chaman c	1	(1,1)	(0,0)
mohabbat sahab se start hua tha rape ka jo sala apni beti ko bhi nhi chhodta tha	1	(1,1)	(0,0)

Analysing pseudo labels fake,sentiment over hate dataset



Sentence	Label_ S	Label_F (G,M)	Label_H (G,M)
isna muslim ko chalang kiya ha iska movie koi mat dekhna agr tum sab muslim hoga to nhi dekho ge	0	(0,0)	(1,1)
bakwas actor ki bakwaas film	0	(1,1)	(1,1)
jo sachcha musalman hoga vo iski movie nahi dekhega bhai kyu k isne muslamano ko chalange kiya hai so mai to nahi dekhunga jiski jo marzi bhai	0	(0,0)	(1,0)
eid mubarak bhaijan	0	(0,0)	(0,0)
salman khan bolliwood ki shan banchuka back to back blockbuster superhit movi name dabbang redy bodyguard ek tha tiger dabbang jai ho kick bajarangi bahizan superhit jayagi fix	0	(1,1)	(0,0)

Analysing pseudo labels fake,hate over sentiment dataset

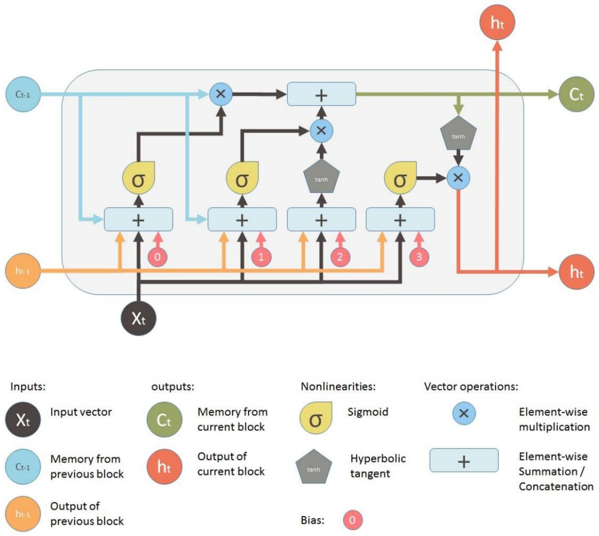
7 Multi Label Classification

LSTM Based Models

LSTM

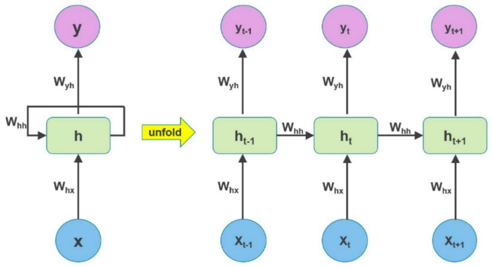
- Long Short-Term Memory (LSTM) networks are a type of recurrent neural network capable of learning order dependence in sequence prediction problems.

- **LSTM Architecture**
  The following model is built up on Long short-term memory shortly LSTM, a very good algorithm especially for text classification jobs. LSTM is a type of artificial neural network ar- chitecture used to process multiple input data points in images, speech, audio as well as text. It consists of a cell and three gates, an input gate, forget gate and output gate.
  Unlike other architectures, LSTM has con- nections for feedback which are helpful regu- lating the information flow through the gates.



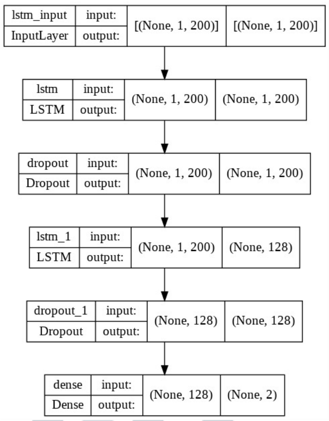
The architecture is designed in such a way that it can remember the long-term dependencies of the data being presented to it.
It could over- come the vanishing gradient problem that arises when using Recurrent Neural Networks. These models can be trained in both supervised and unsupervised manner.
A simple workflow of LSTM can be visual- ised in the below picture and the weights used at each layer can be visualised in the following table.

- LSTM hidden layers



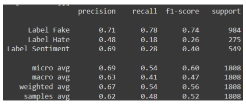
- **Steps**

  1. Clean the data
  2. Get Embeddings
  3. We split the data as 90
  4. Train the model $(Epochs : 30, Softmax, Embeddings)$
  5. Test the model
  6. Predict the outputs for hate and sentiment data sets.


lstm_input | InputLayer | input: | output: | [(None, 1, 200)] | [(None, 1, 200)]
lstm | LSTM | input: | output: | (None, 1, 200) | (None, 1, 200)
dropout | Dropout | input: | output: | (None, 1, 200) | (None, 1, 200)
lstm_1 | LSTM | input: | output: | (None, 1, 200) | (None, 128)
dropout_1 | Dropout | input: | output: | (None, 128) | (None, 128)
dense | Dense | input: | output: | (None, 128) | (None, 2)

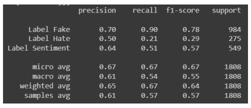
MODEL	model acc	model loss	validation acc	validation loss
LSTM-softmax	0.7310	0.4999	0.7225	0.900
LSTM-sigmoid	0.7302	0.4994	0.7371	0.4907

comparing LSTM with softmax and sigmoid



```
                precision    recall  f1-score   support

     Label Fake       0.71      0.78      0.74       984
     Label Hate       0.48      0.18      0.26       275
Label Sentiment       0.69      0.28      0.40       549

      micro avg       0.69      0.54      0.60      1808
      macro avg       0.63      0.41      0.47      1808
   weighted avg       0.67      0.54      0.56      1808
    samples avg       0.62      0.48      0.52      1808
```

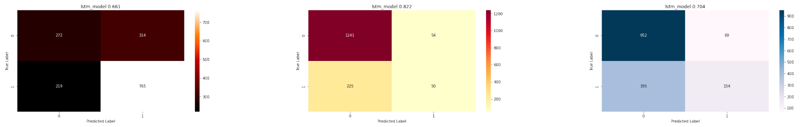
confusion matrix,LSTM-Softmax



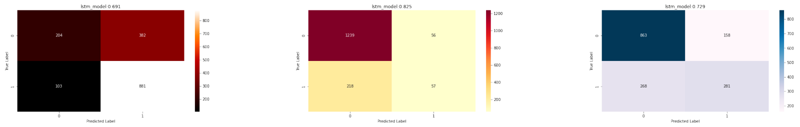
```
                precision    recall  f1-score   support

     Label Fake       0.70      0.90      0.78       984
     Label Hate       0.50      0.21      0.29       275
Label Sentiment       0.64      0.51      0.57       549

      micro avg       0.67      0.67      0.67      1808
      macro avg       0.61      0.54      0.55      1808
   weighted avg       0.65      0.67      0.64      1808
    samples avg       0.61      0.57      0.57      1808
```

confusion matrix,LSTM-Sigmoid

- Confusion matrix for fake,hate,sentiment of LSTM-softmax model.



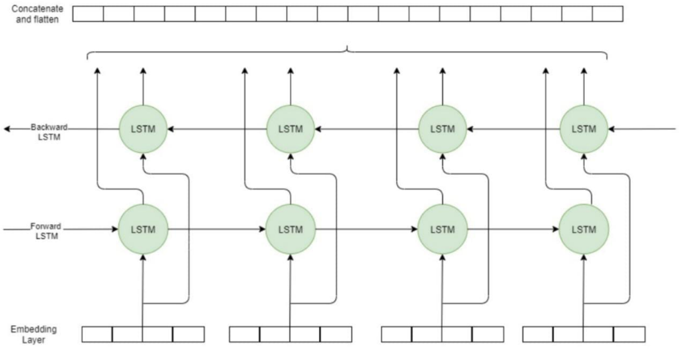
- Confusion matrix for fake,hate,sentiment of LSTM-sigmoid model.



Bi-LSTM With Attention

- BiLSTM is simply bidirectional LSTM, which means the signal propagates backward as well as forward in time.

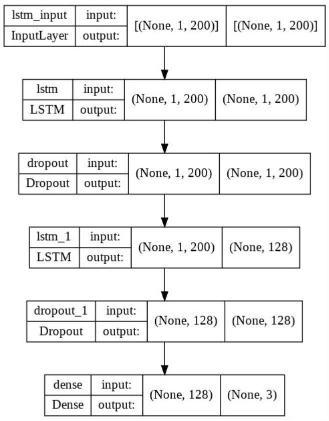
- **Bi-LSTM Architecture**
  A Bidirectional LSTM, or BiLSTM, is a sequence processing model that consists of two LSTMs: one taking the input in a forward direction, and the other in a backwards direction. The first on the input sequence as-is and the second on a reversed copy of the input sequence. This can provide additional context to the network and result in faster and even fuller learning on the problem



Bi-LSTM architecture

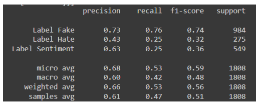
- **Steps**

  1. Clean the data
  2. Get Embeddings
  3. We split the data as 90
  4. Train the model $(Epochs : 22, Softmax, Embeddings)$
  5. Test the model



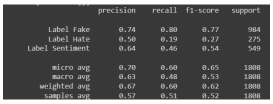
MODEL	model acc	model loss	validation acc	validation loss
BiLSTM-softmax	0.7286	0.4941	0.7146	0.4880
BiLSTM-sigmoid	0.7295	0.4949	0.7331	0.4877

comparing Bi-LSTM with softmax and sigmoid



```
                 precision    recall  f1-score   support

      Label Fake       0.73      0.76      0.74       984
      Label Hate       0.43      0.25      0.32       275
 Label Sentiment       0.63      0.25      0.36       549

       micro avg       0.68      0.53      0.59      1808
       macro avg       0.60      0.42      0.48      1808
    weighted avg       0.66      0.53      0.56      1808
     samples avg       0.61      0.47      0.51      1808
```

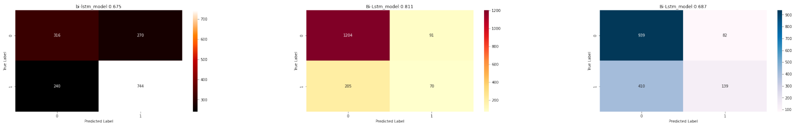
classification report,BiLSTM-Softmax



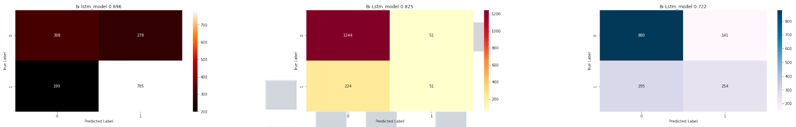
```
                 precision    recall  f1-score   support

      Label Fake       0.74      0.80      0.77       984
      Label Hate       0.50      0.19      0.27       275
 Label Sentiment       0.64      0.46      0.54       549

       micro avg       0.70      0.60      0.65      1808
       macro avg       0.63      0.48      0.53      1808
    weighted avg       0.67      0.60      0.62      1808
     samples avg       0.57      0.51      0.52      1808
```

classification report,BiLSTM-Sigmoid

- Confusion matrix for fake,hate,sentiment of BiLSTM-softmax model.



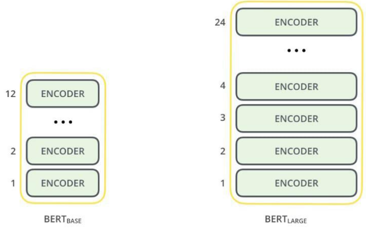
- Confusion matrix for fake,hate,sentiment of BiLSTM-sigmoid model.



BERT Based Models

BERT

- BERT, which stands for Bidirectional Encoder Representations from Transformers, is based on Transformers, a deep learning model in which every output element is connected to every input element, and the weightings between them are dynamically calculated based upon their connection.
- Both Bert Base and Large models have a large number of encoded layers where 12 for Base and 24 for larger version.
- Both versions have larger feed forward networks 768 and 1024 hidden units.

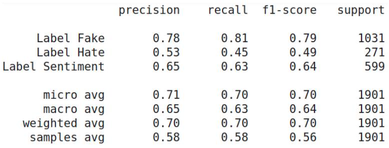


Roberta

- RoBERTa stands for Robustly Optimized BERT Pre-training Approach.
- The goal was to optimize the training of BERT architecture in order to take lesser time during pre-training.
- **Implementation of BERT and Transformer models**
  Simple Transformer models are built with a particular Natural Language Processing (NLP) task in mind.
  To create a task-specific Simple Transformers model,you will typically specify a **model-type** and a **model-name**.
- **STEPS**
  1. Divide the train data and test data as 60
  2. Load a pre trained Models(Bert/Roberta)
  3. $Trainthemodel(Epochs : 5, Max_SequenceLength : 200, Learning_Rate : 3e - 5)$
  4. Evaluate the model and then test
  5. Predict the Output.

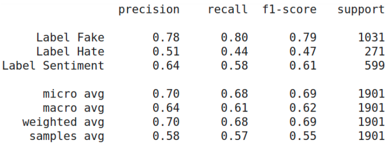
MODEL	layers	hidden layers	attention heads
BERT-base	12	768	12
BERT-large	24	1024	16
m-BERT	12	768	12
Roberta-base	12	768	12
Roberta-large	12	768	12

Parameters of BERT models



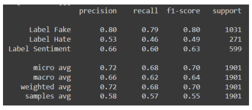
```
                 precision    recall  f1-score   support

      Label Fake       0.78      0.81      0.79      1031
      Label Hate       0.53      0.45      0.49       271
 Label Sentiment       0.65      0.63      0.64       599

       micro avg       0.71      0.70      0.70      1901
       macro avg       0.65      0.63      0.64      1901
    weighted avg       0.70      0.70      0.70      1901
     samples avg       0.58      0.58      0.56      1901
```

classification report,BERT-base



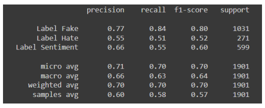
```
                 precision    recall  f1-score   support

      Label Fake       0.78      0.80      0.79      1031
      Label Hate       0.51      0.44      0.47       271
 Label Sentiment       0.64      0.58      0.61       599

       micro avg       0.70      0.68      0.69      1901
       macro avg       0.64      0.61      0.62      1901
    weighted avg       0.70      0.68      0.69      1901
     samples avg       0.58      0.57      0.55      1901
```

classification report,BERT-large



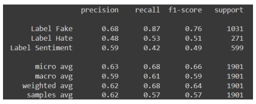
```
                 precision    recall  f1-score   support

      Label Fake       0.80      0.79      0.80      1031
      Label Hate       0.53      0.46      0.49       271
 Label Sentiment       0.66      0.60      0.63       599

       micro avg       0.72      0.68      0.70      1901
       macro avg       0.66      0.62      0.64      1901
    weighted avg       0.72      0.68      0.70      1901
     samples avg       0.58      0.57      0.55      1901
```

classification report,m-BERT



```
                 precision    recall  f1-score   support

      Label Fake       0.77      0.84      0.80      1031
      Label Hate       0.55      0.51      0.52       271
 Label Sentiment       0.66      0.55      0.60       599

       micro avg       0.71      0.70      0.70      1901
       macro avg       0.66      0.63      0.64      1901
    weighted avg       0.70      0.70      0.70      1901
     samples avg       0.60      0.58      0.57      1901
```

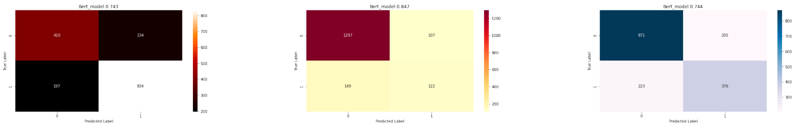
classification report,Roberta-base



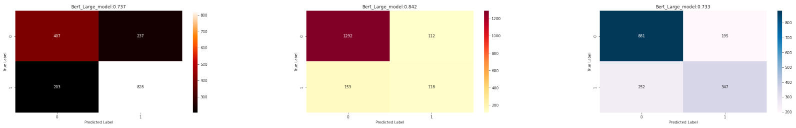
```
                 precision    recall  f1-score   support

      Label Fake       0.68      0.87      0.76      1031
      Label Hate       0.48      0.53      0.51       271
 Label Sentiment       0.59      0.42      0.49       599

       micro avg       0.63      0.68      0.66      1901
       macro avg       0.59      0.61      0.59      1901
    weighted avg       0.62      0.68      0.64      1901
     samples avg       0.62      0.57      0.57      1901
```

classification report,Roberta-Large

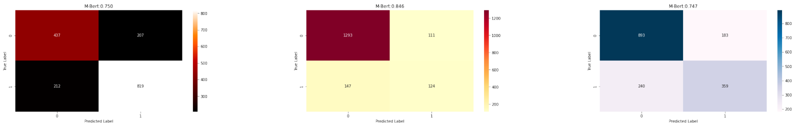
- Confusion matrix for fake,hate,sentiment of BERT-base model.



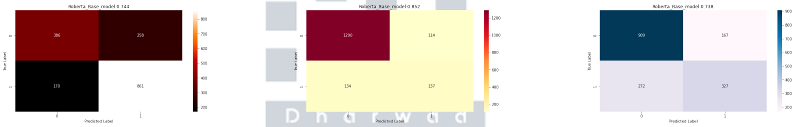
- Confusion matrix for fake,hate,sentiment of BERT-large model.



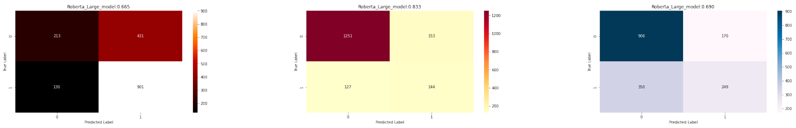
- Confusion matrix for fake,hate,sentiment of mBERT model.



- Confusion matrix for fake,hate,sentiment of Roberta-base model.



- Confusion matrix for fake,hate,sentiment of Roberta-large model.



Comparing all models:

MODEL	Epochs	Accuracy score	Micro average F1 score	Macro average F1 score
LSTM-softmax	30	0.3649	0.60	0.45
LSTM-sigmoid	30	0.4070	0.65	0.51
BiLSTM-softmax	22	0.3605	0.60	0.49
BiLSTM-sigmoid	22	0.3847	0.62	0.48
BERT-base	5	0.4728	0.70	0.63
BERT-large	5	0.4788	0.70	0.64
m-BERT	5	0.4985	0.0.72	0.66
Roberta-base	5	0.4692	0.71	0.64
Roberta-large	1	0.4713	0.66	0.59

Comparing BERT models

8 Conclusion & Future Scope

- Identifying fake and hate news generating in social media platforms can help to reduce its impact over that thing,area,person or group of people targeted by such news.

- And it can further developed along with image processing to read the thumbnails of videos to avoid misleading public towards fake news.

9 Links to refer:

- *to find dataset & code at:click here*

10 References:

- https://github.com/drimpossible/Sub-word-LSTM
- https://www.analyticsvidhya.com/blog/2020/01/3-important-nlp-libraries-indian-languages-python/
- https://huggingface.co/sentence-transformers/paraphrase-xlm-r-multilingual-v1
- https://simpletransformers.ai/docs/multi-modal-classification-model/
- https://github.com/ThilinaRajapakse/simpletransformers
- https://towardsdatascience.com/evaluating-multi-label-classifiers-a31be83da6ea
- https://towardsdatascience.com/evaluating-multi-label-classifiers-a31be83da6ea
- https://arxiv.org/ftp/arxiv/papers/2011/2011.03327.pdf
- https://medium.datadriveninvestor.com/installing-pytorch-and-tensorflow-with-cuda-enabled-gpu-f747e6924779
- https://github.com/Siddhartha15/Hindi-Fake-News-Detection/tree/main/Data