Fake news Classification
Project Overview

Business Overview

What is Fake News?
Fake news is the deliberate presentation of (typically) false or misleading claims as news, where the claims are misleading by design.

How News and digital media evolved?
The news media evolved from newspapers, tabloids, and magazines to a digital form such as online news platforms, blogs, social media feeds, and many news mobile apps. News outlets benefitted from the widespread use of social media/mobile platforms by providing updated news in near real time to its subscribers.
It became easier for consumers to acquire the latest news at their fingertips. So, These digital media platforms become very powerful due to their easy accessibility to the world and ability to allow users to discuss and share ideas and debate over issues such as democracy, education, health, research and history.
However, apart from advantage, false/fake news articles on digital platforms are getting very common and mainly used with a negative intent for their own benefit such as political and financial benefit, creating biased opinions, manipulating mindsets, and spreading absurdity.

How big is this Problem ?
With the rapid adoption of Internet, social media and digital platforms (such as Facebook, Twitter, news portals or any social media), anybody can spread untrue and biased information. It is virtually impossible to prevent Fake News from being created. There has been a rapid increase in the spread of fake news in the last decade, it's not limited to any one domain like politics but covering various other domains such as sports, health, history, entertainment and also science and research. If we take the 2016 US presidential election, there were lots biased and fake news published to influence. Another example could be of COVID-19, we generally come across many misleading/fake news everyday which can have serious consequences and may lead to create panic among people and spread pandemic more rapidly.

What is Solution?

Therefore, It is important and absolutely necessary to identify and differentiate Fake News from real news. One of the ways is to determine by expert and fact check of every news, but this is time consuming and requires skills which can not be shared. Second, we can automate the detection of Fake News by using the techniques of Machine learning and Artificial Intelligence. The Online news content has diverse

unstructured format data(such as documents, videos, and audios), here we will concentrate on text format news. With the advancement of and Natural language processing It is possible now that we can identify the deceptive and fake nature of articles or sentences.

There is widespread study and experimentation happening in this area to identify the Fake news for all medium(Video, audio and Text) news.

Data Description

In our study we used the Fake news dataset from Kaggle to classify unreliable news articles as Fake news using Deep learning Technique Sequence to Sequence programming.

A full training dataset with the following attributes

- id : unique id for a news article
- title: the title of a news article
- author: author of the news article
- text : the text of the article; could be incomplete
- label : a label that marks the article as potentially unreliable
    - 1 : unreliable
    - 0 : reliable

Tech Stack

➔ Language : Python

➔ Libraries : Scikit-learn , Tensorflow , Keras, Glove, Flask, nltk, pandas, numpy

Approach

1. Data cleaning / Pre-processing (outlier/missing values/categorical) -
    a. Removing Missing record
    b. Merge all text together
    c. Removing special character from text
2. Sequence Data Preparation
    a. Tokenizing text after preprocessing
    b. Build Vocabulary to filter text sets: Choose length of maximum vocabulary size
    c. Sequence data preparation:
        i. Use vocab
        ii. Maximum sequence length
        iii. Padding

3. Word Embedding - This is a step where we convert text data to meaningful numerical vectors. We use a pre-trained glove to convert into a numeric vector.

4. Build Sequence Model - Building Sequence layer with embedding, Dense, Dropout with below sequence layer:
    a. Simple RNN
    b. LSTM
    c. GRU

5. Validate Model Training - Which model will be finalized on the basis of the following
    a. Confusion matrix
    b. Accuracy

6. Model comparison - Model comparison in terms of performance, stability and computation time

Modular code overview

```
input
  |__glove
        |__glove.6B.100d.txt
  |__submit.csv
  |__test.csv
  |__train.csv


src
  |__engine.py
  |__ML_pipeline
           |__build_model.py
           |__clean_data.py
           |__constants.py
           |__data_preprocessing.py
           |__evaluate_model.py
           |__text_statistics.py
           |__text_tokenizer.py
           |__train_model.py
           |__utils.py
           |__word_embedding.py

lib
  |__fakenews_seq_classification.ipynb
output
  |__models
  |__reports
```

Once you unzip the modular_code.zip file you can find the following folders within it.

1. input
2. src
3. output
4. lib

1. The input folder contains all the data that we have for analysis. In our case, it will contain three csv files which are
    a. submit.csv
    b. test.csv

    c. train.csv

   It also has another folder called glove which contains the glove embedding file

2. The src folder is the heart of the project. This folder contains all the modularized code for all the above steps in a modularized manner. It further contains the following.
    a. ML_pipeline
    b. engine.py

The ML_pipeline is a folder that contains all the functions put into different python files which are appropriately named. These python functions are then called inside the engine.py file

3. The output folder contains two folders. They are :
    a. models - The models folder contains all the models that we trained for this data saved as reusable files. These models can be easily loaded and used for future use and the user need not have to train all the models from the beginning.
    b. reports - The report folder contains a csv file which stores all the models trained along with their accuracy and other details.
4. The lib folder is a reference folder. It contains the original ipython notebook that we saw in the videos.

Project Takeaways

1. Understanding the problem statement
2. Understanding the Sequence problem and their types
3. Understanding Sequence neural network approach like: RNN, GRU and LSTM.
4. Importing the dataset and importing libraries
5. Performing basic text cleaning
6. Perform text preprocessing: Stop word removal, Stemming etc.
7. Text tokenization using keras tokenizer
8. Sequence data preparation with tokenizer and padding
9. Train and test split for model validation
10. Explaining Text vectorization and Word embedding
11. Build Word embedding layer with Glove
12. Important parameter before train model.
13. Explaining Sequence model steps
14. Implementing Simple RNN
15. Implementing LSTM and GRU
16. Making Test predictions using the trained model.
17. Comparing Model training and validation loss and performance.
18. Comparing LSTM and GRU performance and computation Time.