

Sarcastic News Headline Classification

Vishal Verma | Data Science

# Problem Statement

We have a dataset with News Headlines and Labels if the headline is sarcastic or not and using that data we have to make a classification model to check whether the news headline is sarcastic or not.

## Concept Used

It is an NLP Text classification problem.

I have used different type of text cleaning and preprocessing to get the best outcome or Accuracy from the trained model.

I have used machine learning as well as deep learning with multiple layers to train the models and bring out the best accuracy in classification of the news headlines.

## Expectations & Solutions

1. **Find the 6 topics related to news article headlines.**

* I was able to find the six topics related to the news headlines using the LDA Model which states that each document is a collection of topics and each word attribute to a topic.
* The Topics were correlated as most of the words were being used in multiple topics
* The list of topic labels defined manually are mentioned below :
  1. White House
  2. Republican Government
  3. American Schooling
  4. Donald Trump
  5. American Infrastructure
  6. Police Brutality

1. **Do EDA to find the top 50 sarcastic words. Make a word cloud for top 200 frequent words.**

* To find the top 50 sarcastic words I used a bag of words model to create a document term matrix and then used a sum function to add all the values of the columns to get the count of the words.
* Once the count of words was generated I sorted the words in descending order and got the top 50 sarcastic words using the slicing of the list.
* For top 200 frequent words simply used the corpus of the clean text to generate the word cloud of top 200 words. Which is also attached at the top of Report

1. **Can you identify sarcastic sentences? Can you distinguish between fake news and legitimate news?**

* We were able to identify the sarcastic sentences based on the label provided to us which helped the model to learn about the specific words and how they can be termed as sarcastic.
* We cannot distinguish between fake news and legitimate news as we are not given any kind of label which will help the model learn about the news being fake. If we were provided certain kind of feature which would depict the fakeness of document or even words, then we could have had the means to identify the news headline.

## Steps INvolved in Project

1. The First step was to read the data and do some exploratory data analysis. Where I checked for the distribution of the labels in the data set provided with that I figured out that the dataset was balanced and had approximately equal number of positive and negative cases.
2. Then we moved on to see the different variable or features provided in the data set as we were just supposed to tell the news headlines is sarcastic or not I discarded the article link feature which had the article related to news headline.
3. Once I finished with the task I moved on to the text cleaning process where I removed numbers, punctuations, stopwords and extra spaces in the headlines and created a corpus with all the text
4. After that I used Bag of Words model to create a document term matrix and summed all the columns to get the top 50 sarcastic words in the dataset and also used the same corpus to create the word cloud of top 200 most frequent words in the news headlines
5. Used TFIDF matrix to create topic modelling with LDA concept and came up with 6 different topics that were related to the dataset.
6. Implemented different machine learning models and found the different accuracy score. I chose F1 score and ROC AUC score to evaluate the model as in classification problem we cannot trust accuracy that well.
7. After implementing machine learning models I didn’t get a good F1 score so I used deep learning model to train the neural network and get better F1 score.

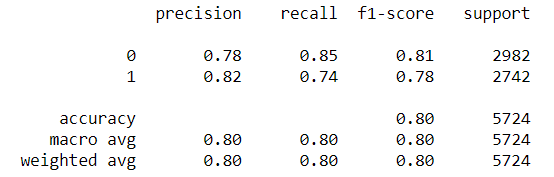
## Models:

### Logistic Regression:

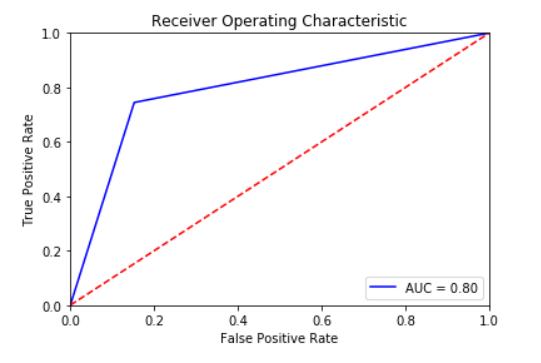
The First model that was used for the classification was Logistic Regression.

Which gave the following results:





1Logistic Regression Scores

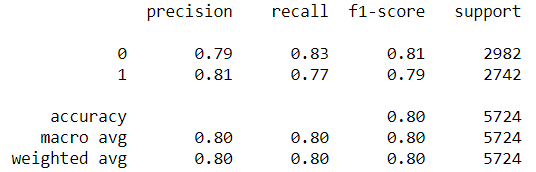


### Naïve Bayes

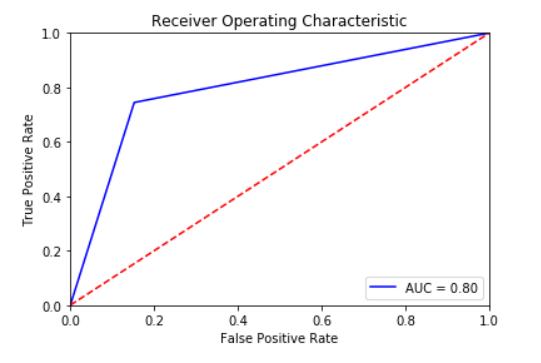
As the accuracy was not that high we used Naïve Bayes model.

Which gave the following results





2Naive Bayes Score

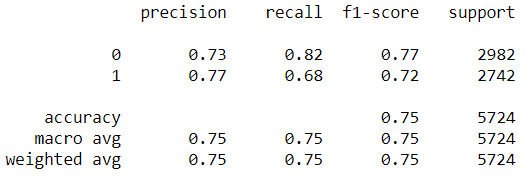


### Random Forest

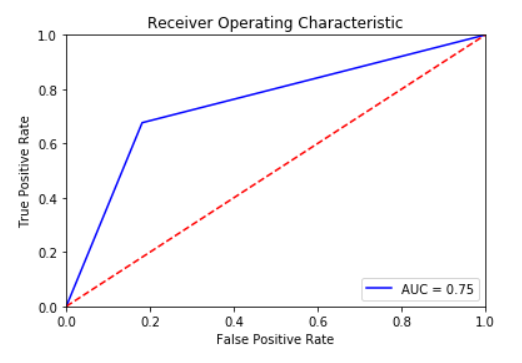
To further enhance the accuracy scores I tried going for more complex models to get a better model for classification

Which gave the following results





3Random Forest Scores

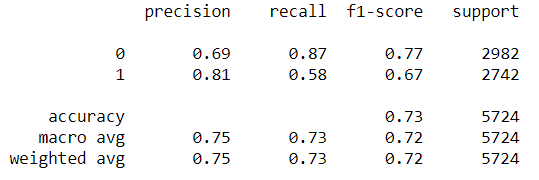


### XG Boost

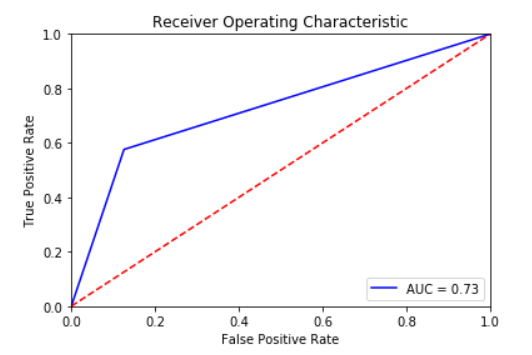
I used XG boost to further enhance the accuracy score.

The results are as follows:





4XG Boost Scores

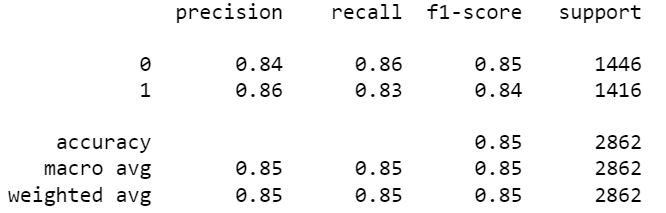


### Deep Learning

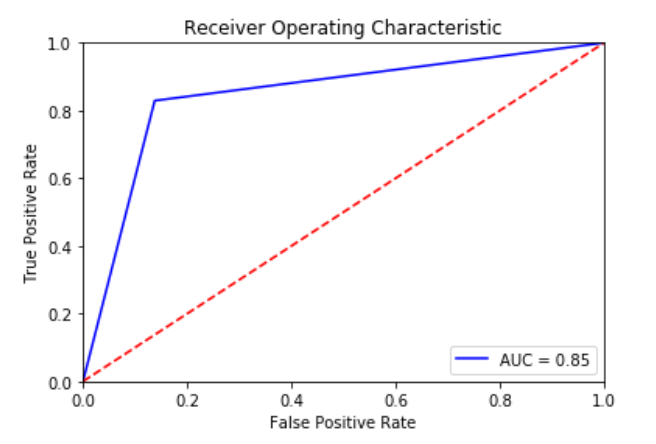
As I wasn’t satisfied with the accuracy the machine learning models were providing I tried to use the power of neural networks to get a model which can predict the news headlines better.

I had the basic understanding of multiple neural networks and with the help of that I tried multiple layers of neural network using RNN for better predictions.





5Deep Learning Scores



## Summary

After trying multiple machine learning models and deep learning models we can say that out of all the machine learning models Naïve Bayes gave the best classification model.

Whereas in deep learning RNN gave the best predictions out of them all.

# Instructions to run the code

## Python

* Open Jupyter notebook in the directory where you have stored the solution file
* Change the working directory to get the data from the dataset provided
* Run the cells one by one using shift + enter.

## R

* Open R studio
* Open the R solution file
* Change the working directory to get the data from the dataset provided
* Run the commands in R studio one by one using ctrl+enter