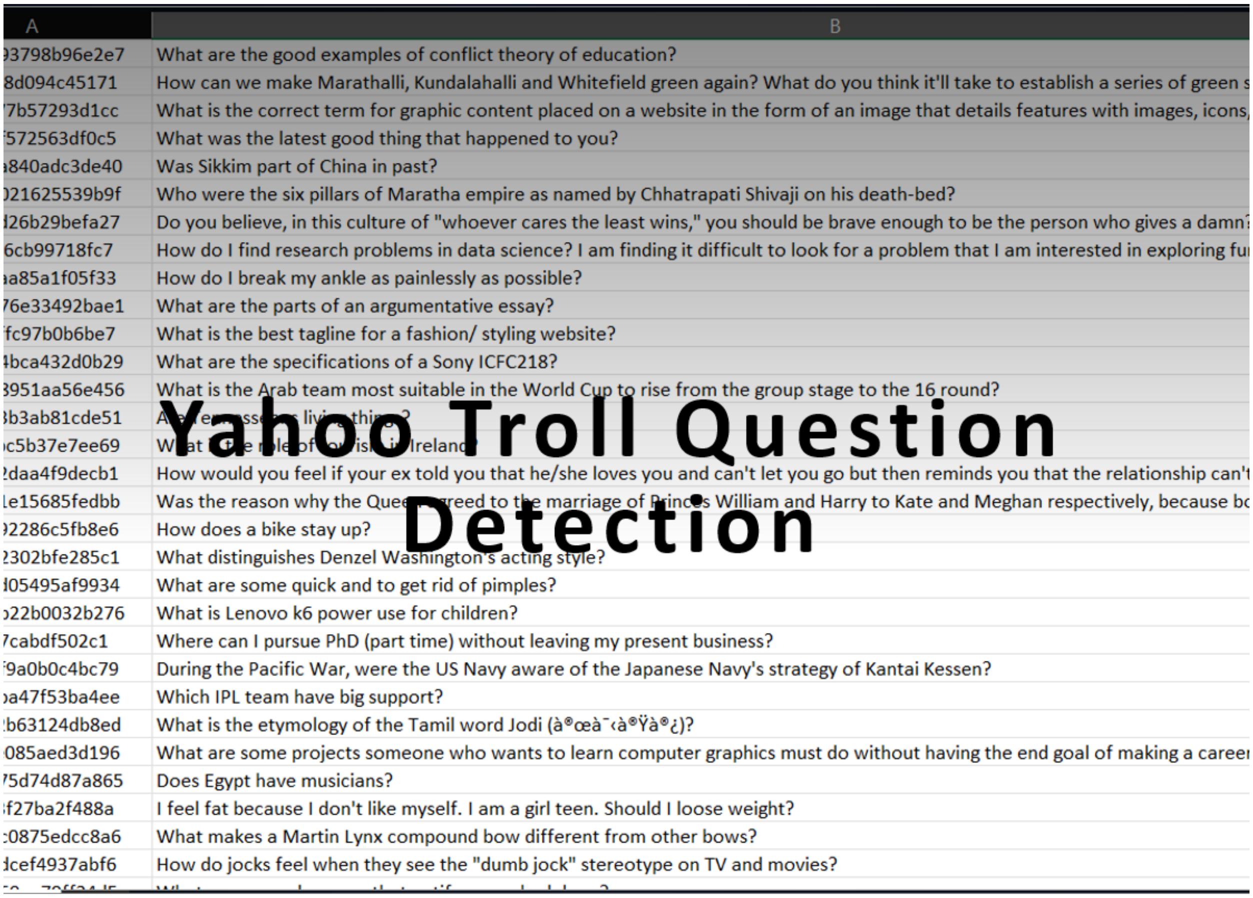
AI511 Machine Learning

Project Report

Yahoo Troll Question Detection



A	B
93798b96e2e7	What are the good examples of conflict theory of education?
8d094c45171	How can we make Marathalli, Kundalahalli and Whitefield green again? What do you think it'll take to establish a series of green s
7b57293d1cc	What is the correct term for graphic content placed on a website in the form of an image that details features with images, icons,
572563df0c5	What was the latest good thing that happened to you?
a840adc3de40	Was Sikkim part of China in past?
21625539b9f	Who were the six pillars of Maratha empire as named by Chhatrapati Shivaji on his death-bed?
126b29befa27	Do you believe, in this culture of "whoever cares the least wins," you should be brave enough to be the person who gives a damn?
6cb99718fc7	How do I find research problems in data science? I am finding it difficult to look for a problem that I am interested in exploring fu
a85a1f05f33	How do I break my ankle as painlessly as possible?
76e33492bae1	What are the parts of an argumentative essay?
fc97b0b6be7	What is the best tagline for a fashion/ styling website?
4bca432d0b29	What are the specifications of a Sony ICFC218?
3951aa56e456	What is the Arab team most suitable in the World Cup to rise from the group stage to the 16 round?
3b3ab81cde51	Are [illegible] living thing?
c5b37e7ee69	What is the role of tourism in Ireland?
2daa4f9decb1	How would you feel if your ex told you that he/she loves you and can't let you go but then reminds you that the relationship can't
1e15685fedbb	Was the reason why the Queen agreed to the marriage of Princes William and Harry to Kate and Meghan respectively, because bo
92286c5fb8e6	How does a bike stay up?
2302bfe285c1	What distinguishes Denzel Washington's acting style?
105495af9934	What are some quick and to get rid of pimples?
22b0032b276	What is Lenovo k6 power use for children?
7cabdf502c1	Where can I pursue PhD (part time) without leaving my present business?
9a0b0c4bc79	During the Pacific War, were the US Navy aware of the Japanese Navy's strategy of Kantai Kessen?
a47f53ba4ee	Which IPL team have big support?
b63124db8ed	What is the etymology of the Tamil word Jodi (à®œà¯‹à®Ÿà®¿)?
085aed3d196	What are some projects someone who wants to learn computer graphics must do without having the end goal of making a career
75d74d87a865	Does Egypt have musicians?
f27ba2f488a	I feel fat because I don't like myself. I am a girl teen. Should I loose weight?
c0875edcc8a6	What makes a Martin Lynx compound bow different from other bows?
dcef4937abf6	How do jocks feel when they see the "dumb jock" stereotype on TV and movies?

Team Members

- Karanjit Saha (IMT2020003)
- Netradeepak Chinchwadkar (IMT2020014)

Dataset

The training dataset consists of 10 lakh rows.

- 61780 rows are of class 1 (i.e. TROLL questions)
- 938130 rows are of class 0 (i.e. NON-TROLL questions)

The testing dataset has 306122 rows.

EDA of training dataset

The dataset has no null values and no repeated rows.

EDA of testingdataset

The dataset has no null values and no repeated rows.

Preprocessing Steps

- **Text Cleaning:-**

    1. Remove punctuations
    2. Remove numerical values
    3. Remove common non-sensical text (e.g. “/n”)
    4. Tokenize text
    5. Lemmatization

    We used “re” and “nltk” modules for applying all the above mentioned operations.

- **Stop Words removal**

    We removed stop words present in the nltk corpus from the text.

- **Combining words**

We combined all the words before giving the data to the vectorizers.

- **Doing Spell check**

Used “symspell” to enable spell checking on the words in the text.

- **Removing Non - english words**

We tried to check our model’s working by removing the non english words as well. For this we are using “nltk.corpus”.

- **Vectorization of the dataset**

Different Vectorizers we have tried are as follows:-

Count Vectorizer

Count Vectorizer creates a matrix in which each unique word is represented by a column of the matrix, and each text sample from the document is a row in the matrix, and each text sample from the document is a row in the matrix. The value of each cell is nothing but the count of the word in that particular text sample.

TFIDF Vectorizer

TFIDF, is a numerical statistic that’s intended to reflect how important a word is to a document.

We tried to see the effect of different parameters of these vectorizers for the same models. Some of the models we explored are:-

1. max_df
2. min_df
3. strip_accents
4. analyzer

5. ngram_range
6. max_features

- **Creating Hstack of vectorizers:-**

  Hstack stacks arrays in sequence horizontally (column wise). Here we use it to combine the sparse matrices obtained from the vectorizers.

- **Doing train-test split:-**

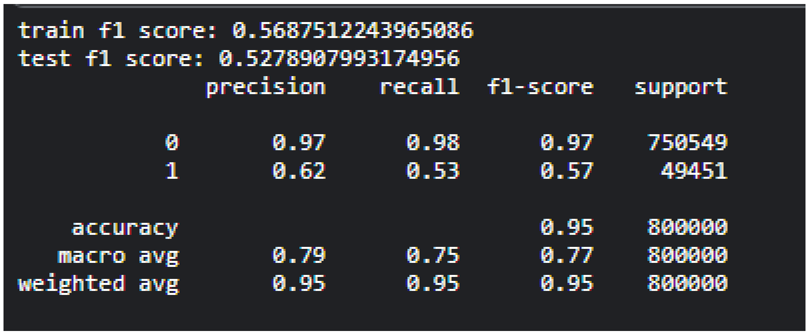
  We are doing the train-test split with test_size=0.2.

- **Storing preprocessed data files**

  Storing preprocessed files in .pickle file format so that we don't have to run all the preprocessing steps before running a model on the dataset.

Binary Classifier Models Used

1. Multinomial Naive Bayes



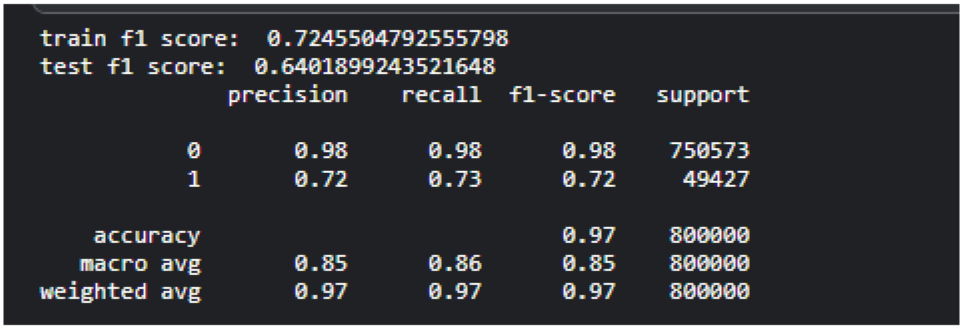
```
train f1 score: 0.5687512243965086
test f1 score: 0.5278907993174956
              precision    recall  f1-score   support

           0       0.97      0.98      0.97    750549
           1       0.62      0.53      0.57     49451

    accuracy                           0.95    800000
   macro avg       0.79      0.75      0.77    800000
weighted avg       0.95      0.95      0.95    800000
```

2. Logistic Regression

This gave the best results as of now. Hence we spent a lot of our time on hypertuning for this particular model. Some of the parameters we explored for this model are :-

a. penalty
b. max_iter

c. solver
d. class_weights
e. Tol
f. C



```
train f1 score:  0.7245504792555798
test f1 score:  0.6401899243521648
              precision    recall  f1-score   support

           0       0.98      0.98      0.98    750573
           1       0.72      0.73      0.72     49427

    accuracy                           0.97    800000
   macro avg       0.85      0.86      0.85    800000
weighted avg       0.97      0.97      0.97    800000
```

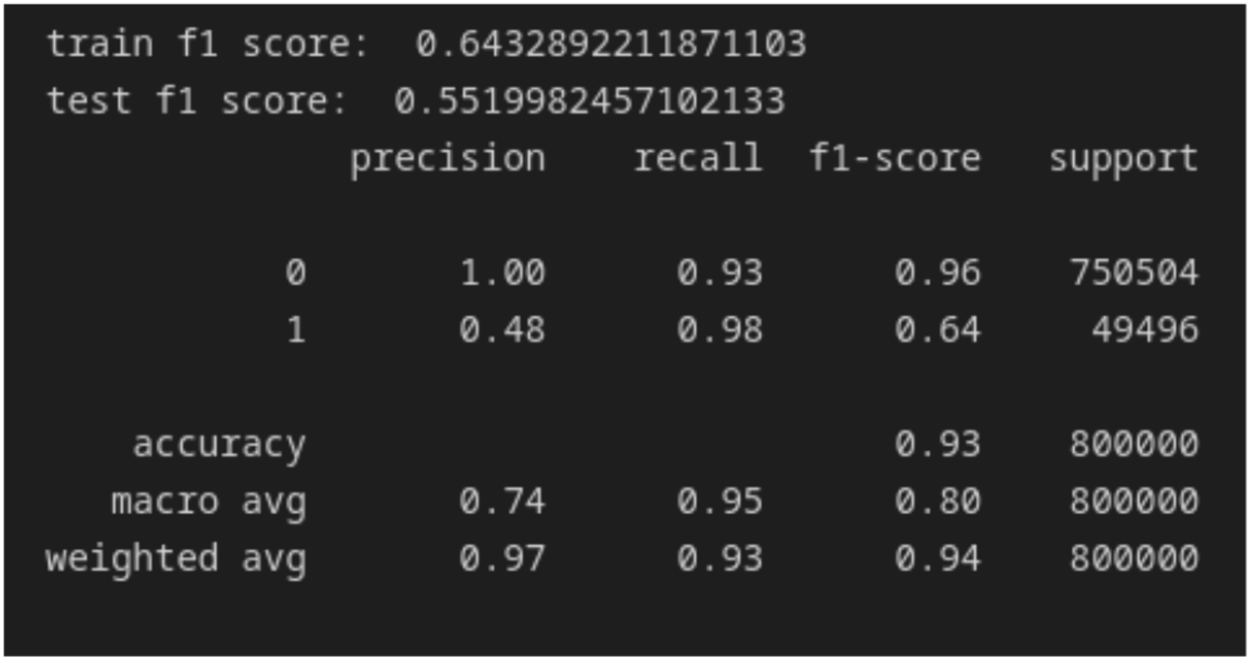
This result is with the parameter class_weight = {0:0.9, 1:2.1} and max_iter =10000

We tested the model with a bunch of class weights which are given below:

- 0: 0.9, 1: 2.1
- 0: 0.9, 1: 2
- 0: 0.9, 1: 1.9
- 0: 0.2, 1: 0.8
- 0: 0.25, 1: 0.75

Ultimately selected 0.9/2.1 as the classweights as they nearly equal precision and recall values which gave it a higher f1-score

3. XGBoost

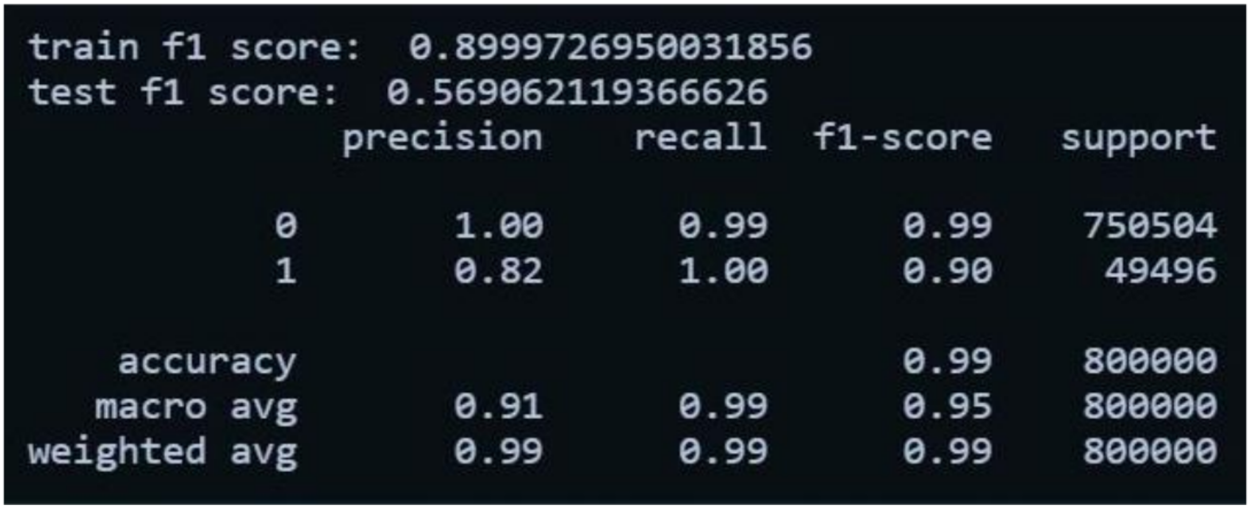


```
train f1 score:  0.6432892211871103
test f1 score:  0.5519982457102133
              precision    recall  f1-score   support

           0       1.00      0.93      0.96    750504
           1       0.48      0.98      0.64     49496

    accuracy                           0.93    800000
   macro avg       0.74      0.95      0.80    800000
weighted avg       0.97      0.93      0.94    800000
```

4. ADABoost & KNN

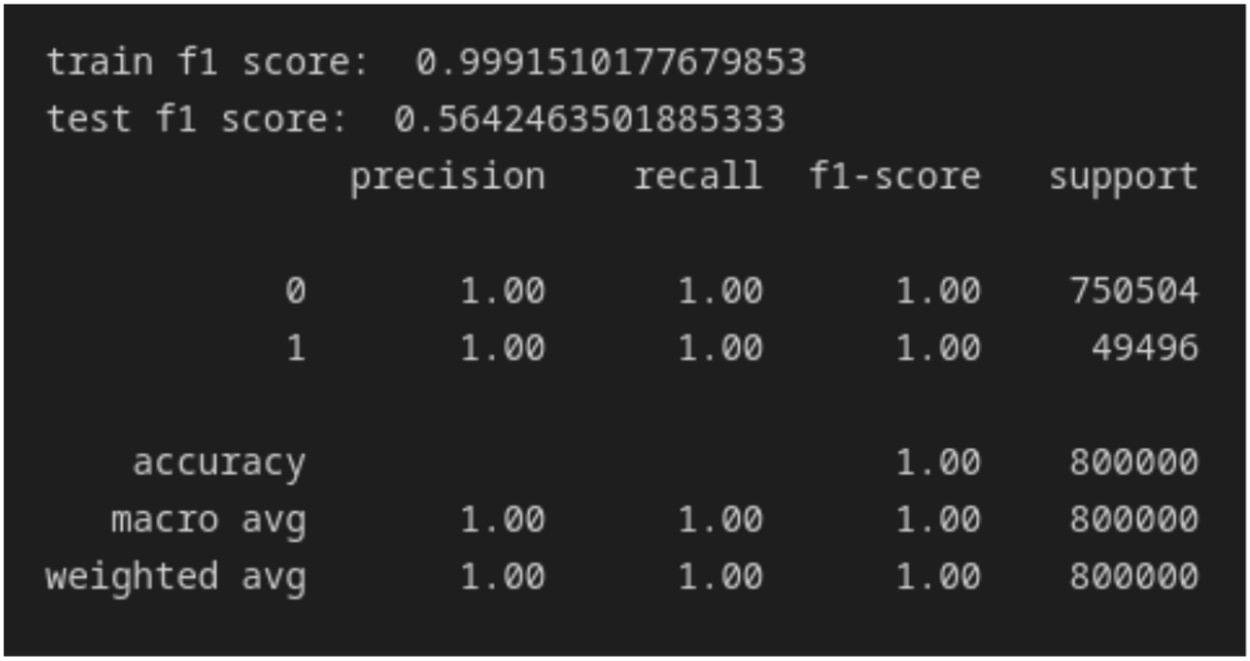
These models took way too long to run. We tried running it for 8-9 hrs after which we had to force stop the model.

5. Perceptron



```
train f1 score:  0.8999726950031856
test f1 score:  0.569062119366626
              precision    recall  f1-score   support

           0       1.00      0.99      0.99    750504
           1       0.82      1.00      0.90     49496

    accuracy                           0.99    800000
   macro avg       0.91      0.99      0.95    800000
weighted avg       0.99      0.99      0.99    800000
```

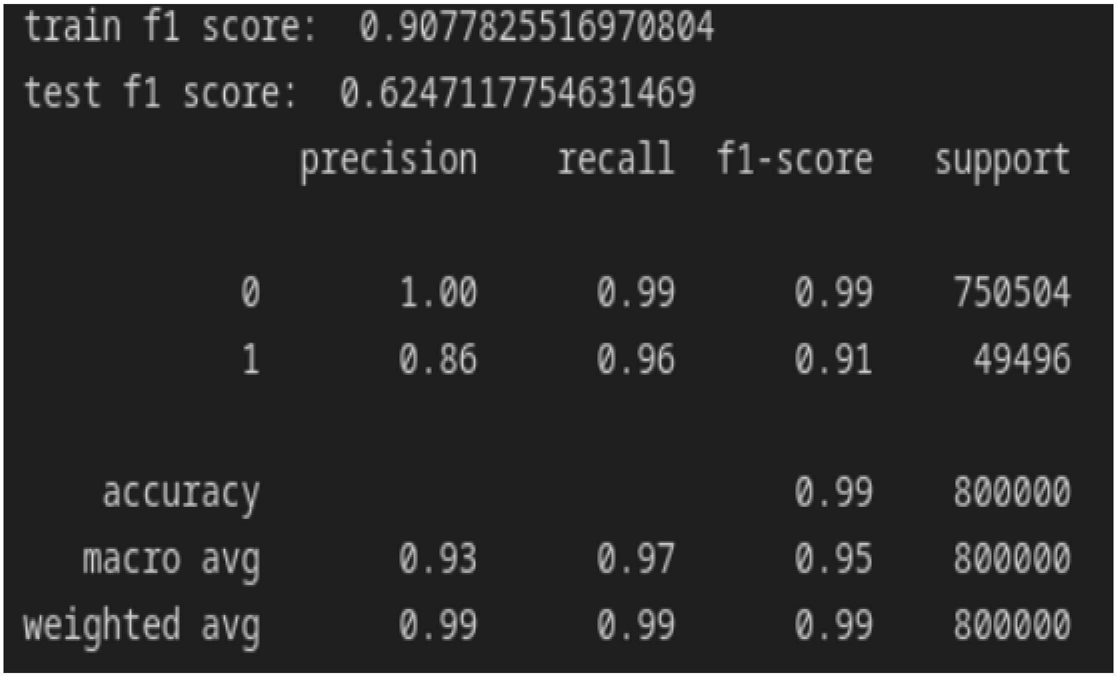
6. SVM



```
train f1 score:  0.9991510177679853
test f1 score:  0.5642463501885333
              precision    recall  f1-score   support

           0       1.00      1.00      1.00    750504
           1       1.00      1.00      1.00     49496

    accuracy                           1.00    800000
   macro avg       1.00      1.00      1.00    800000
weighted avg       1.00      1.00      1.00    800000
```

We used LinearSVC for our dataset since other kernels were taking too long to run. (We waited for 8-10 hrs before stopping it)

7. Bagging Classifier



```
train f1 score:  0.9077825516970804
test f1 score:  0.6247117754631469
              precision    recall  f1-score   support

           0       1.00      0.99      0.99    750504
           1       0.86      0.96      0.91     49496

    accuracy                           0.99    800000
   macro avg       0.93      0.97      0.95    800000
weighted avg       0.99      0.99      0.99    800000
```

8. Stacking Classifier

This ran for way too long(6-7 hours) without giving a result.

Results

Models	f1-score
Multinomial Naive Bayes	0.53
Logistic Regression	0.65
XGBoost	0.55
Perceptron	0.57
SVM	0.56
Bagging Classifier	0.62

Conclusion

We got the best result of 0.64111 on the testing dataset by using the Logistic Ression model on the whole training dataset with parameters class_weights = {0:0.9, 1:2} and max_iter =10000.