Lab MST Performa (Disruptive Technologies-1)

Section: 22BCB-124 **Group: A**

Title of the Project: **Cyberbullying Classifier**

Names and UIDS of group members:

Ankush Singh	**(22BCT10002)**
Prayrit Dhingra	**(22BCT10015)**
Harsh Saini	**(22BCT10007)**
Moksh Verma	**(22BCT10059)**
Md Karimul Hasan	**(22BCT10001)**

Problem statement:

Often people are targeted and cyberbullied on social media platforms like Discord, Twitter, Reddit and Instagram. We aim to make a text classifier that classifies tweets from twitter as 'not_cyberbullying' or 'cyberbullying' with it's type – racial, age based, gender based, etc.

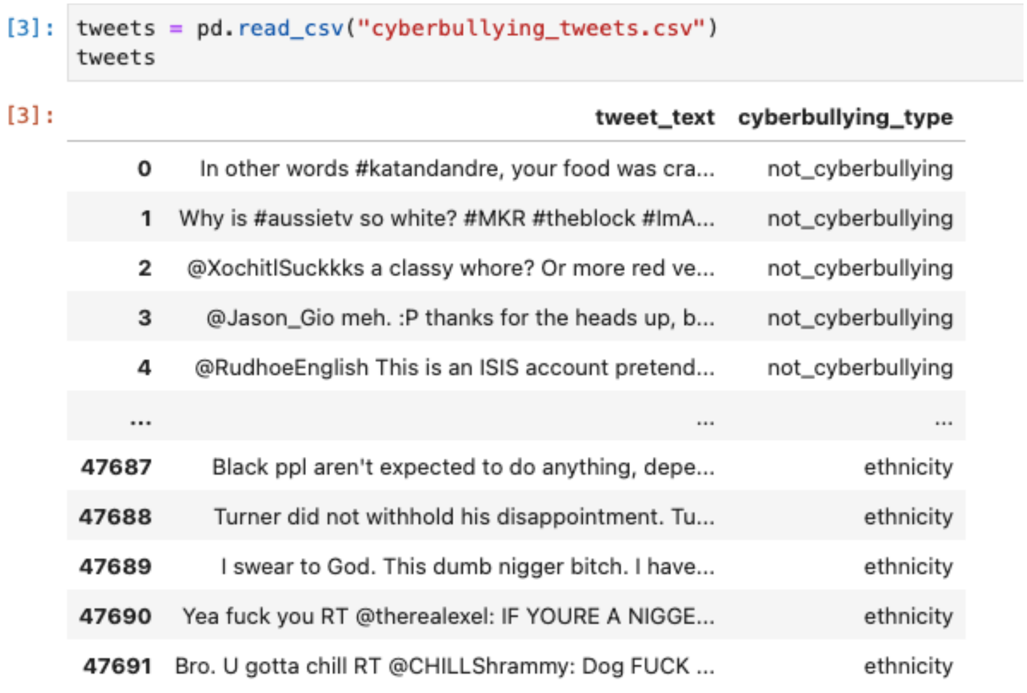
Key Features/Benefits:

- **Attempt to detect Cyberbullying from posts on social media.**
- **Text Classification using SVMs (Support Vector Machines)**
- **Dataset used : 46 thousand labeled tweets from twitter**
- **Detect and prevent Racism, Sexism, Religionism, etc**

Software used:

- **Python3, Google Colab/Jupyter notebook**
- **Github for code collaboration**
- **Pandas, Numpy, Matplotlib**
- **PyCaret, Tensorflow**
- **Kaggle Platform**

Deliverables:

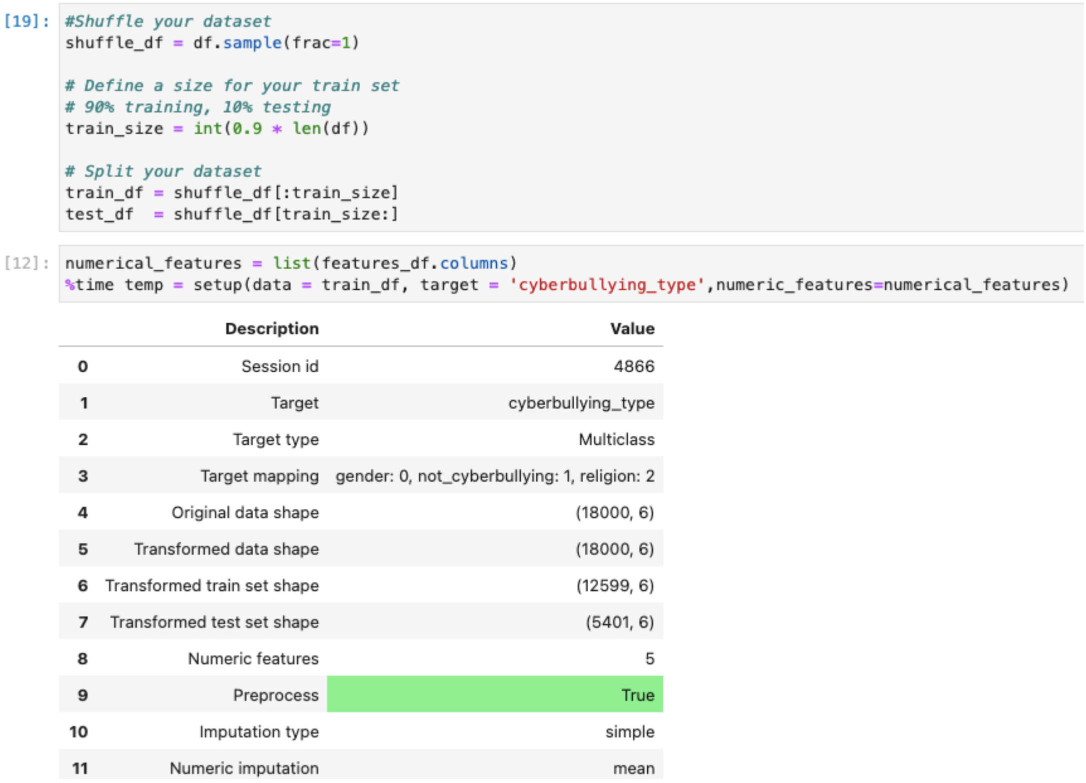


```
[3]: tweets = pd.read_csv("cyberbullying_tweets.csv")
     tweets
```

[3]:

	tweet_text	cyberbullying_type
0	In other words #katandandre, your food was cra...	not_cyberbullying
1	Why is #aussietv so white? #MKR #theblock #ImA...	not_cyberbullying
2	@XochitlSuckkks a classy whore? Or more red ve...	not_cyberbullying
3	@Jason_Gio meh. :P thanks for the heads up, b...	not_cyberbullying
4	@RudhoeEnglish This is an ISIS account pretend...	not_cyberbullying
...	...	...
47687	Black ppl aren't expected to do anything, depe...	ethnicity
47688	Turner did not withhold his disappointment. Tu...	ethnicity
47689	I swear to God. This dumb nigger bitch. I have...	ethnicity
47690	Yea fuck you RT @therealexel: IF YOURE A NIGGE...	ethnicity
47691	Bro. U gotta chill RT @CHILLShrammy: Dog FUCK ...	ethnicity

1: Vulgar Speech Dataset (40thousand + columns)

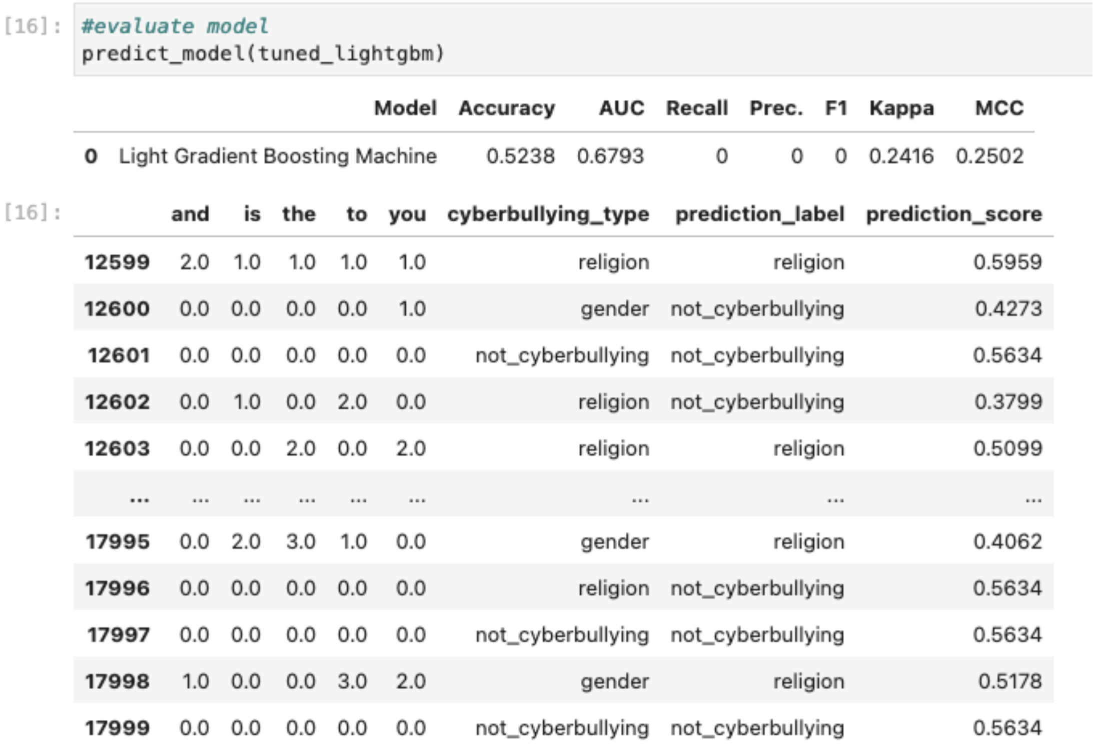


```
[19]: #Shuffle your dataset
      shuffle_df = df.sample(frac=1)

      # Define a size for your train set
      # 90% training, 10% testing
      train_size = int(0.9 * len(df))

      # Split your dataset
      train_df = shuffle_df[:train_size]
      test_df  = shuffle_df[train_size:]
```

```
[12]: numerical_features = list(features_df.columns)
      %time temp = setup(data = train_df, target = 'cyberbullying_type',numeric_features=numerical_features)
```

	Description	Value
0	Session id	4866
1	Target	cyberbullying_type
2	Target type	Multiclass
3	Target mapping	gender: 0, not_cyberbullying: 1, religion: 2
4	Original data shape	(18000, 6)
5	Transformed data shape	(18000, 6)
6	Transformed train set shape	(12599, 6)
7	Transformed test set shape	(5401, 6)
8	Numeric features	5
9	Preprocess	True
10	Imputation type	simple
11	Numeric imputation	mean

2: Splitting dataset into training and testing parts



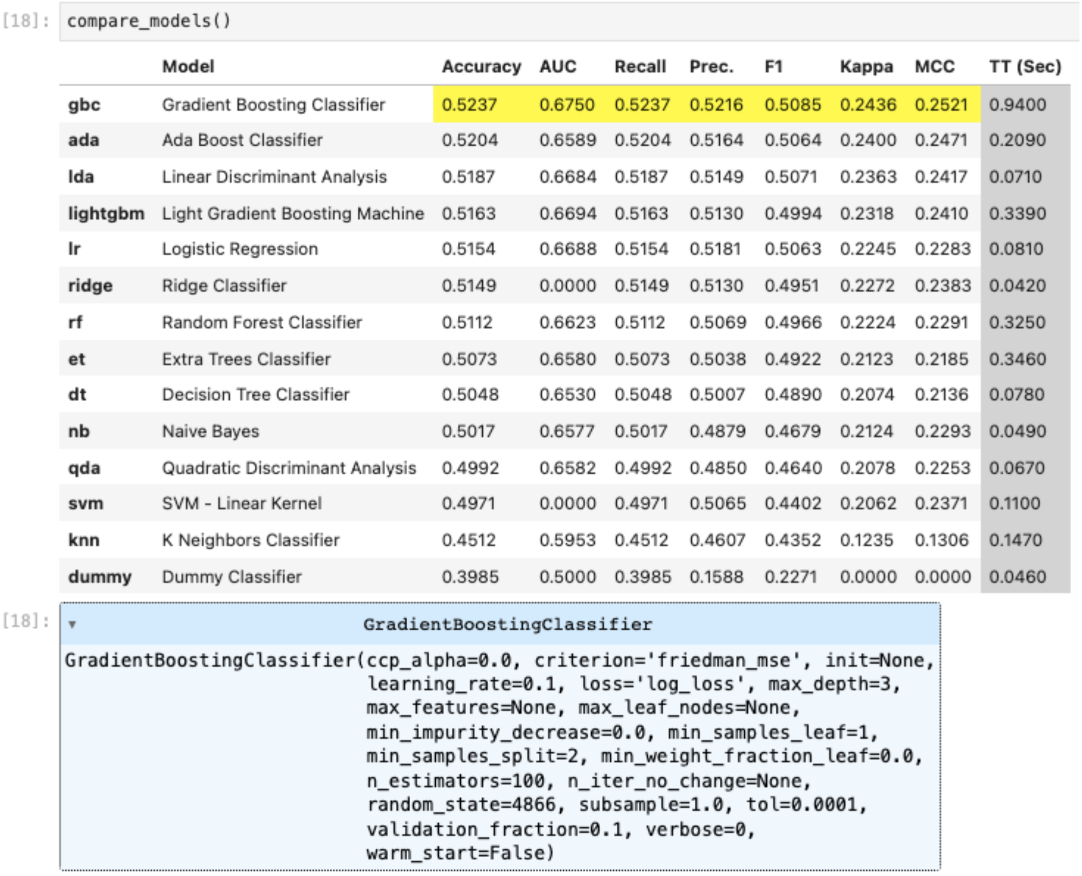
```
[16]: #evaluate model
      predict_model(tuned_lightgbm)
```

	Model	Accuracy	AUC	Recall	Prec.	F1	Kappa	MCC
0	Light Gradient Boosting Machine	0.5238	0.6793	0	0	0	0.2416	0.2502

[16]:

	and	is	the	to	you	cyberbullying_type	prediction_label	prediction_score
12599	2.0	1.0	1.0	1.0	1.0	religion	religion	0.5959
12600	0.0	0.0	0.0	0.0	1.0	gender	not_cyberbullying	0.4273
12601	0.0	0.0	0.0	0.0	0.0	not_cyberbullying	not_cyberbullying	0.5634
12602	0.0	1.0	0.0	2.0	0.0	religion	not_cyberbullying	0.3799
12603	0.0	0.0	2.0	0.0	2.0	religion	religion	0.5099
...	...	...	...	...	...	...	...	...
17995	0.0	2.0	3.0	1.0	0.0	gender	religion	0.4062
17996	0.0	0.0	0.0	0.0	0.0	religion	not_cyberbullying	0.5634
17997	0.0	0.0	0.0	0.0	0.0	not_cyberbullying	not_cyberbullying	0.5634
17998	1.0	0.0	0.0	3.0	2.0	gender	religion	0.5178
17999	0.0	0.0	0.0	0.0	0.0	not_cyberbullying	not_cyberbullying	0.5634

3: Testing our trained model



```
[18]: compare_models()
```

	Model	Accuracy	AUC	Recall	Prec.	F1	Kappa	MCC	TT (Sec)
gbc	Gradient Boosting Classifier	0.5237	0.6750	0.5237	0.5216	0.5085	0.2436	0.2521	0.9400
ada	Ada Boost Classifier	0.5204	0.6589	0.5204	0.5164	0.5064	0.2400	0.2471	0.2090
lda	Linear Discriminant Analysis	0.5187	0.6684	0.5187	0.5149	0.5071	0.2363	0.2417	0.0710
lightgbm	Light Gradient Boosting Machine	0.5163	0.6694	0.5163	0.5130	0.4994	0.2318	0.2410	0.3390
lr	Logistic Regression	0.5154	0.6688	0.5154	0.5181	0.5063	0.2245	0.2283	0.0810
ridge	Ridge Classifier	0.5149	0.0000	0.5149	0.5130	0.4951	0.2272	0.2383	0.0420
rf	Random Forest Classifier	0.5112	0.6623	0.5112	0.5069	0.4966	0.2224	0.2291	0.3250
et	Extra Trees Classifier	0.5073	0.6580	0.5073	0.5038	0.4922	0.2123	0.2185	0.3460
dt	Decision Tree Classifier	0.5048	0.6530	0.5048	0.5007	0.4890	0.2074	0.2136	0.0780
nb	Naive Bayes	0.5017	0.6577	0.5017	0.4879	0.4679	0.2124	0.2293	0.0490
qda	Quadratic Discriminant Analysis	0.4992	0.6582	0.4992	0.4850	0.4640	0.2078	0.2253	0.0670
svm	SVM - Linear Kernel	0.4971	0.0000	0.4971	0.5065	0.4402	0.2062	0.2371	0.1100
knn	K Neighbors Classifier	0.4512	0.5953	0.4512	0.4607	0.4352	0.1235	0.1306	0.1470
dummy	Dummy Classifier	0.3985	0.5000	0.3985	0.1588	0.2271	0.0000	0.0000	0.0460

```
[18]: ▾ GradientBoostingClassifier
GradientBoostingClassifier(ccp_alpha=0.0, criterion='friedman_mse', init=None,
                           learning_rate=0.1, loss='log_loss', max_depth=3,
                           max_features=None, max_leaf_nodes=None,
                           min_impurity_decrease=0.0, min_samples_leaf=1,
                           min_samples_split=2, min_weight_fraction_leaf=0.0,
                           n_estimators=100, n_iter_no_change=None,
                           random_state=4866, subsample=1.0, tol=0.0001,
                           validation_fraction=0.1, verbose=0,
                           warm_start=False)
```

4: Comparing accuracy with different models