A COMPARATIVE STUDY OF TOXIC LANGUAGE DETECTION USING NLP ALGORITHM

SHIVAM PANDEY RA1911003010383

MAYANK SINHA RA1911003010386

GUIDE NAME: DR. G RAMYA

TABLE OF CONTENTS

- ABSTRACT
- PROBLEM STATEMENT AND OBJECTIVES
- PROPOSED METHODOLOGY
- BLOCK DIAGRM
- IMPLEMENTATION
- RESULTS
- OUTPUT SCREENSHOTS
- REFERENCES

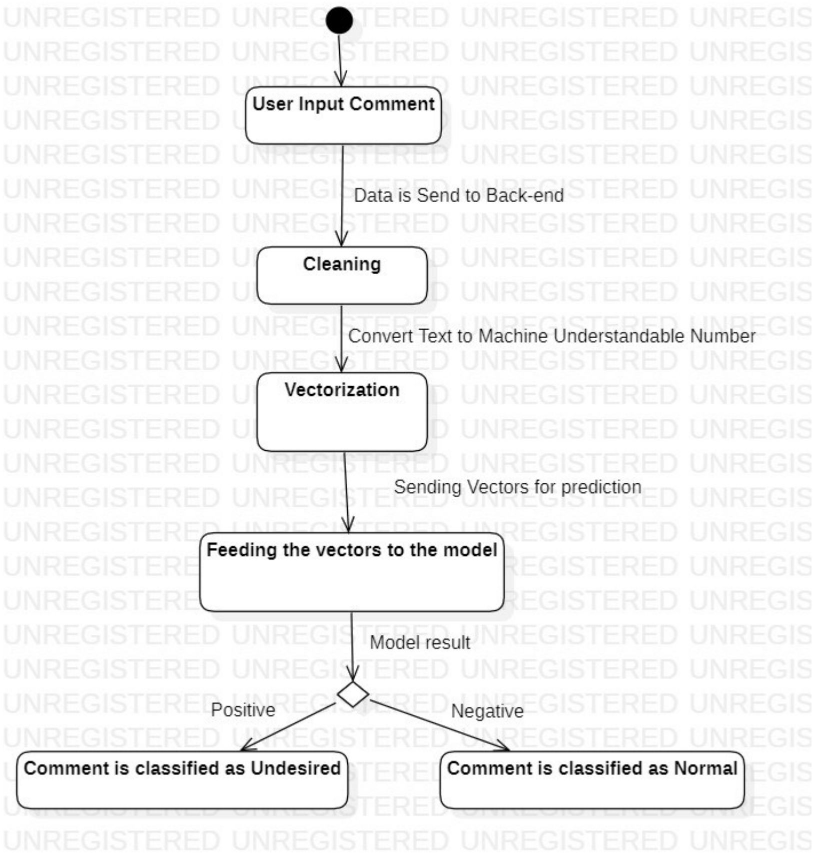
ABSTRACT

- As there are numerous algorithms utilised for toxic language detection in the field of natural language processing now, but they in this algorithm not developed considering production environment.
- The speed of categorization has a significant impact on how quickly an application is used in a production setting where the algorithm is running in real-time.
- The technique is only useful for theoretical applications if it performs classification tasks slowly. In the actual world, in this algorithm need algorithms that operates quickly and with excellent accuracy.
- Through this research, in this algorithm hope to identify an algorithm that works best in the actual world, meaning that there is no compromise in this algorithm speed and accuracy.
- For the classification job, in this algorithm will make use of the toxic language dataset. After classifiers have been trained, they will be put to the test using a variety of criteria, including accuracy, recall, precision, F1 score, and prediction time.
- In this algorithm shall categorise the flaws in different NLP algorithms based on the parameters.

PROPOSED METHODOLOGY

- We will be testing the algorithms in a real world environment, to validate whether these algorithms can solve real world problems when they are deployed in the production environment.
- We trained the algorithms using sklearn python library on toxic language dataset.
- We will compare their training time and then we will use the pickle module to save the model and then we will load the model in the backend of our app and then we will use the model to get predictions in the production environment and then we will check their prediction time and accuracy to understand their viability in solving real world problems.

BLOCK DIAGRAM



User Input Comment

Data is Send to Back-end

Cleaning

Convert Text to Machine Understandable Number

Vectorization

Sending Vectors for prediction

Feeding the vectors to the model

Model result

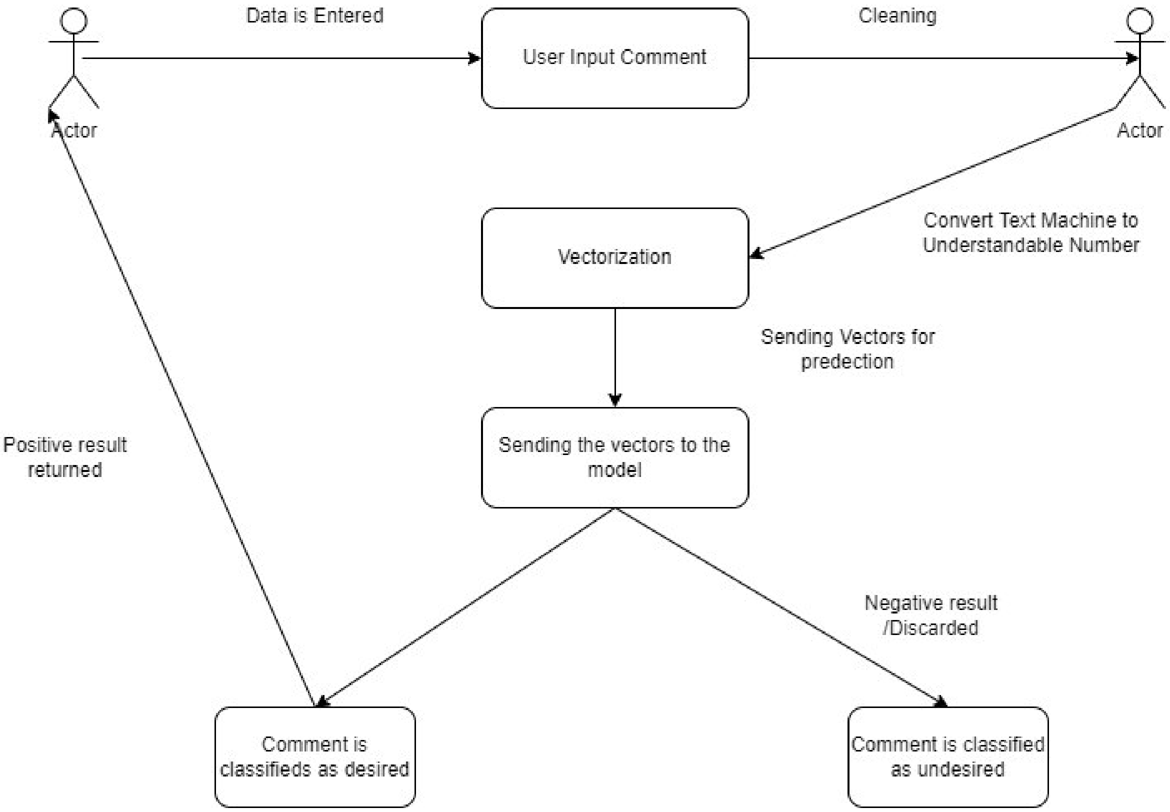
Positive

Negative

Comment is classified as Undesired

Comment is classified as Normal

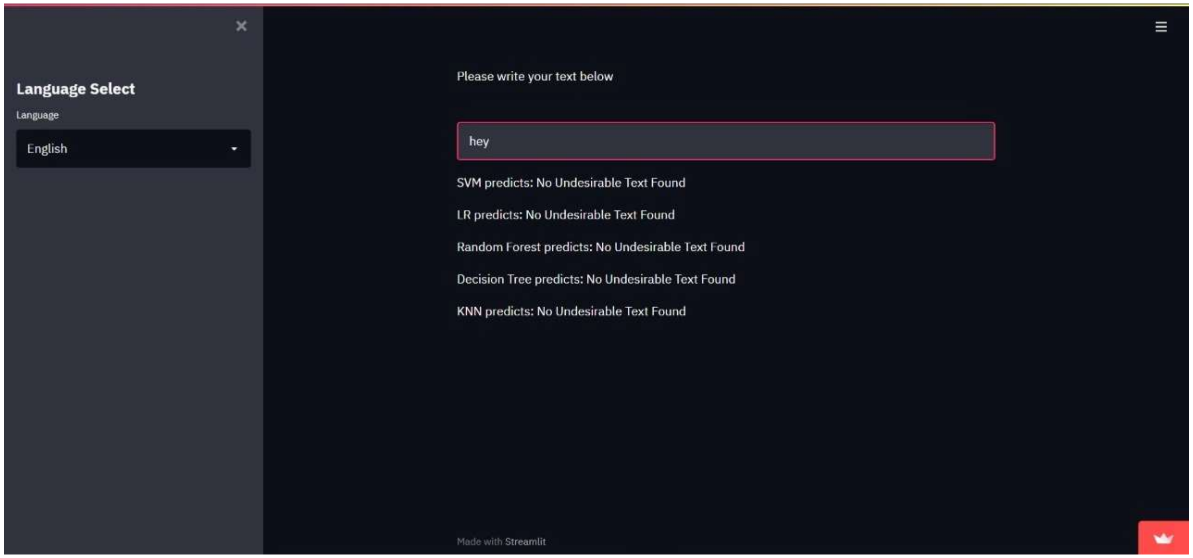
USECASE DIAGRAM


Actor
Data is Entered
User Input Comment
Cleaning
Actor
Convert Text Machine to Understandable Number
Vectorization
Sending Vectors for predection
Sending the vectors to the model
Positive result returned
Negative result /Discarded
Comment is classifieds as desired
Comment is classified as undesired

IMPLEMENTATION(100%)

- We have created the project purely in python programming language.
- We trained various Supervised Machine learning models using Sklearn library on toxic comment dataset, we compared their train time using time library.
- We made our app using the streamlit library, which provides us with the functionality to easily create frontend for our ML app.
- We deployed our app using the streamlit cloud feature and then we tested the time taken to give prediction of our app in the production environment.

WORKING OUTPUTS



Please write your text below

hey

SVM predicts: No Undesirable Text Found

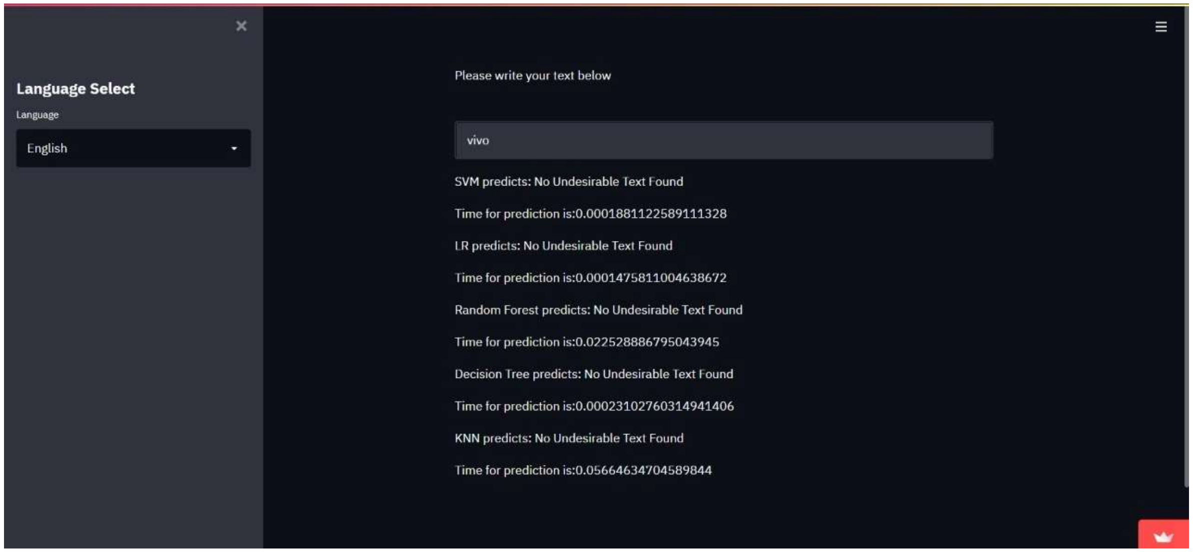
LR predicts: No Undesirable Text Found

Random Forest predicts: No Undesirable Text Found

Decision Tree predicts: No Undesirable Text Found

KNN predicts: No Undesirable Text Found

RESULTS



Language Select

Language

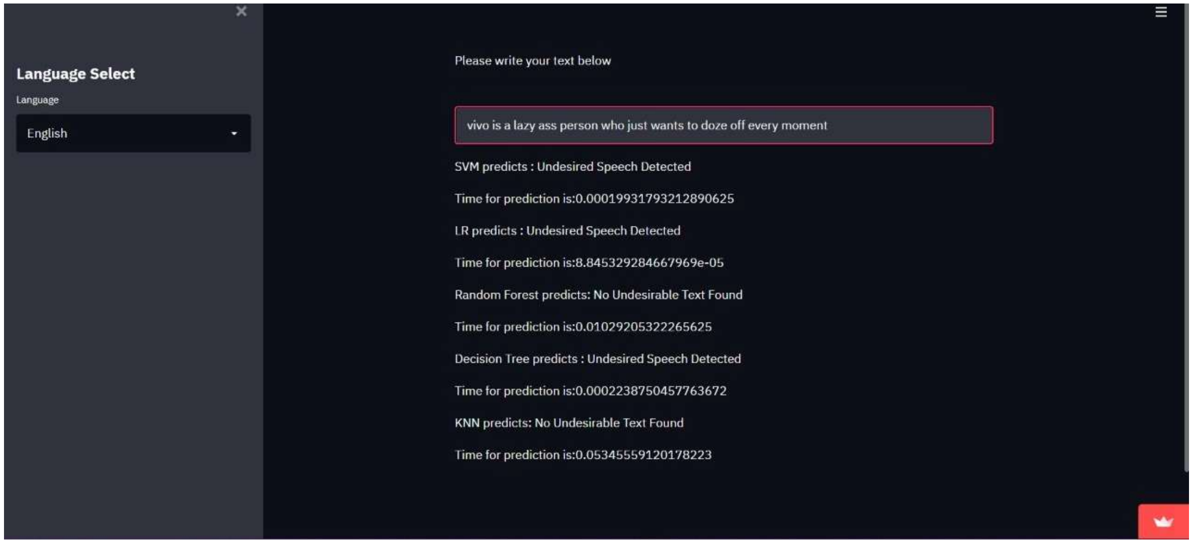
English

Please write your text below

vivo

```
SVM predicts: No Undesirable Text Found

Time for prediction is:0.0001881122589111328

LR predicts: No Undesirable Text Found

Time for prediction is:0.0001475811004638672

Random Forest predicts: No Undesirable Text Found

Time for prediction is:0.022528886795043945

Decision Tree predicts: No Undesirable Text Found

Time for prediction is:0.00023102760314941406

KNN predicts: No Undesirable Text Found

Time for prediction is:0.05664634704589844
```

RESULTS



Language Select

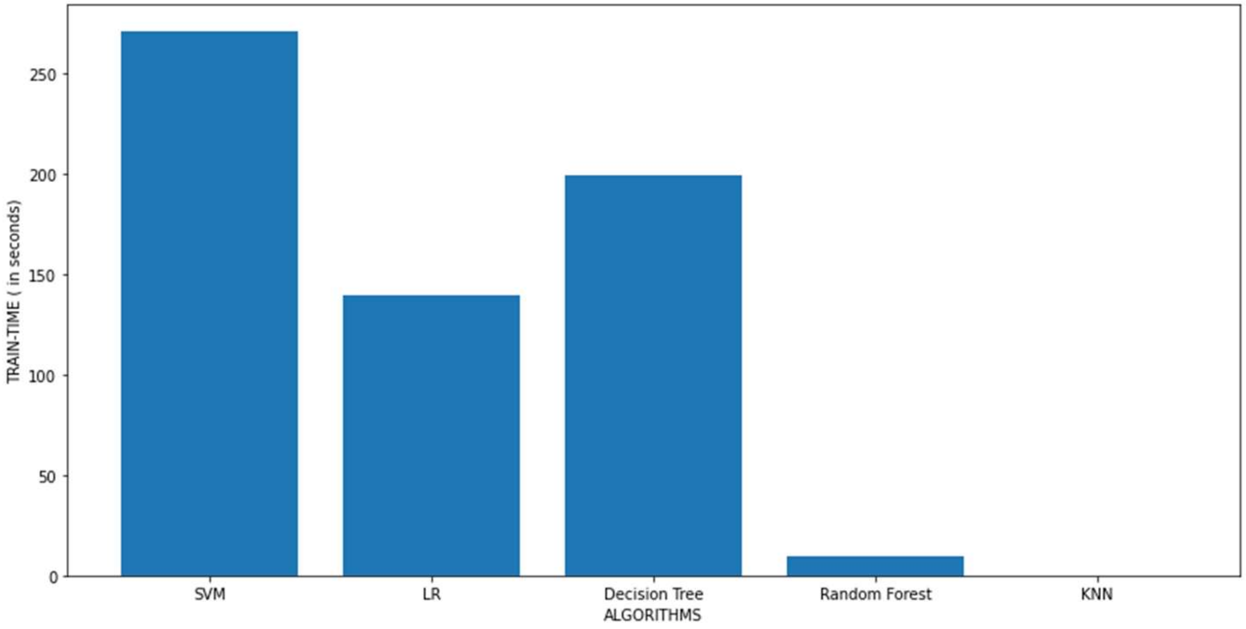
Language

English

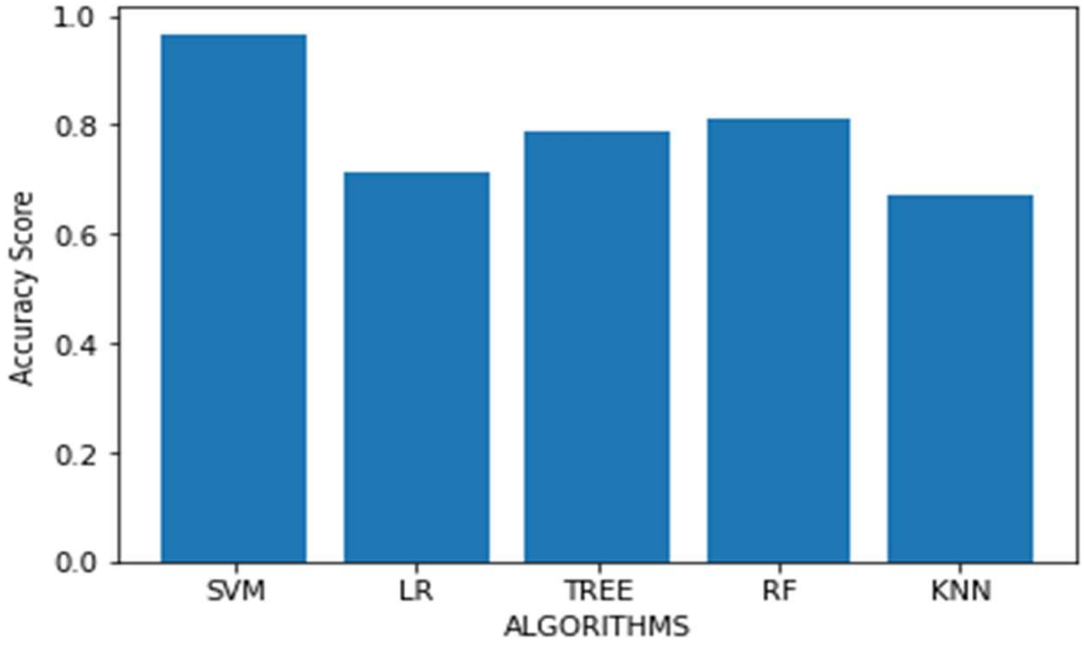
Please write your text below

vivo is a lazy ass person who just wants to doze off every moment

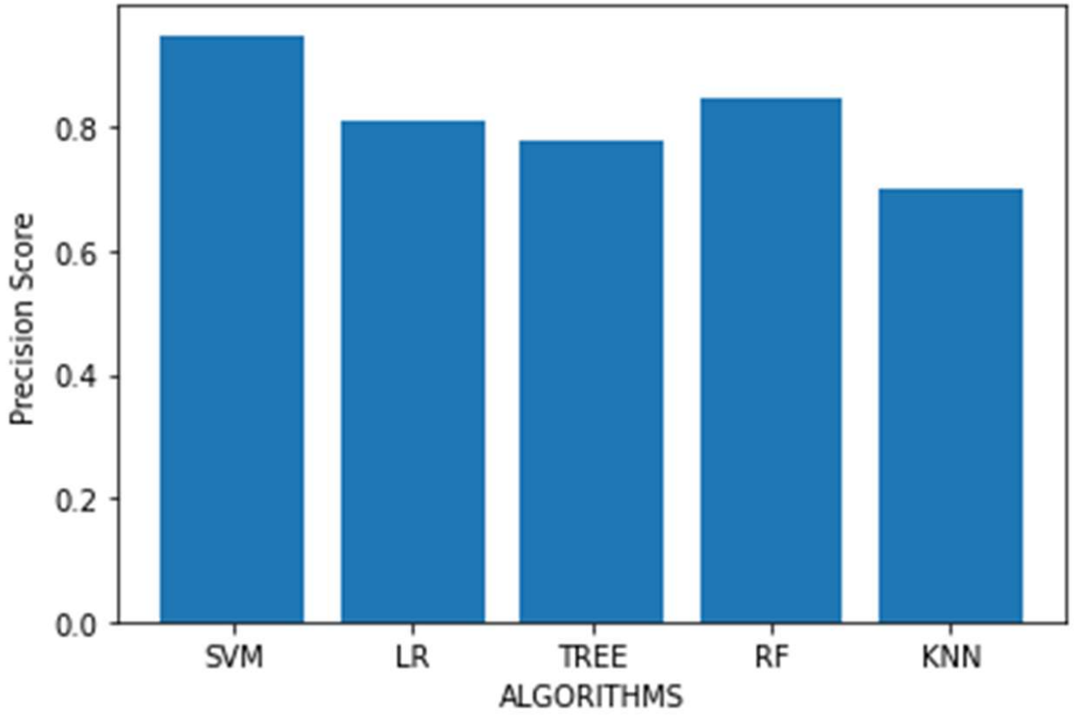
```
SVM predicts : Undesired Speech Detected

Time for prediction is:0.00019931793212890625

LR predicts : Undesired Speech Detected

Time for prediction is:8.845329284667969e-05

Random Forest predicts: No Undesirable Text Found

Time for prediction is:0.01029205322265625

Decision Tree predicts : Undesired Speech Detected

Time for prediction is:0.00022387504577636720

KNN predicts: No Undesirable Text Found

Time for prediction is:0.05345559120178223
```



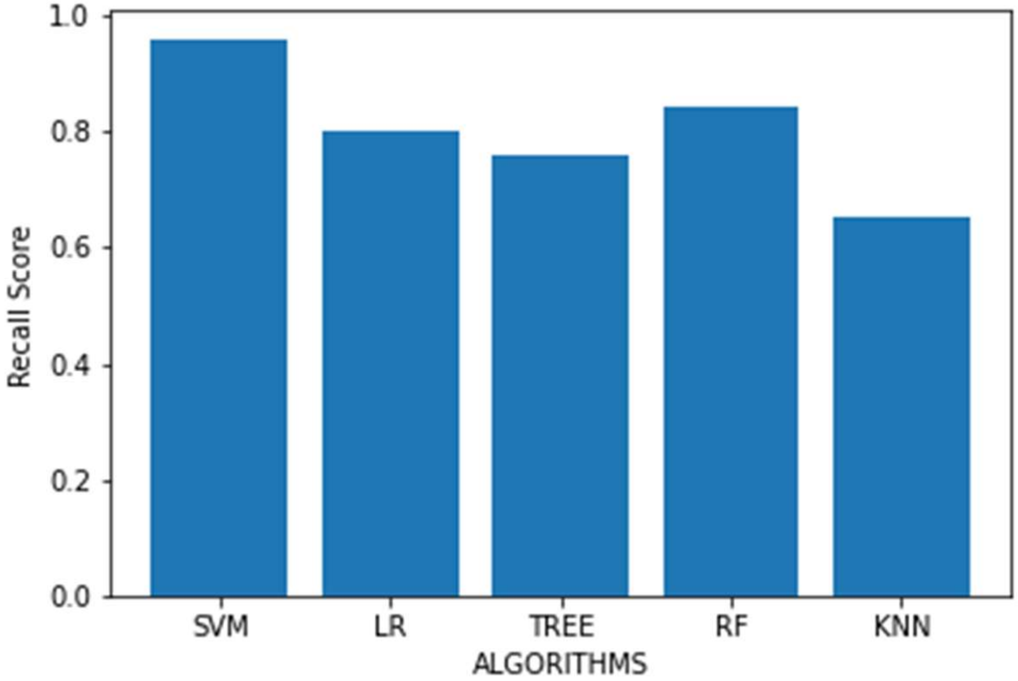
TRAIN TIME COMPARISON



ACCURACY COMPARISON

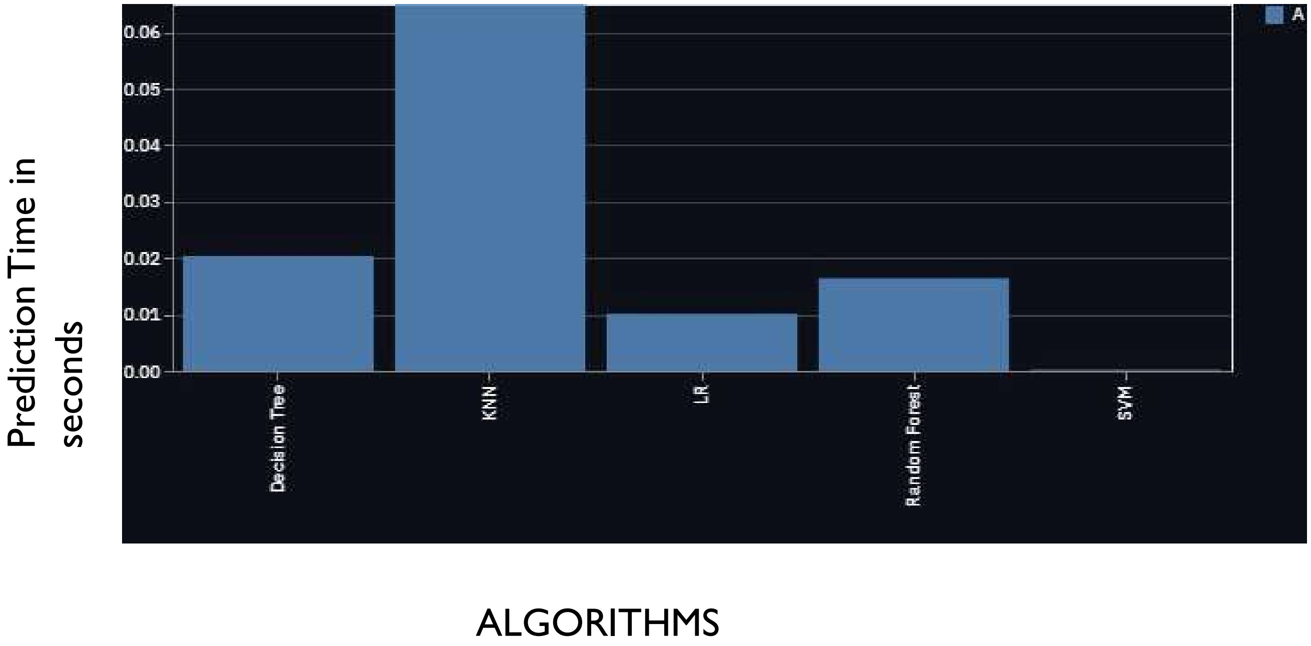


PRECISION COMPARISON



RECALL COMPARISON

PREDICTION TIME IN RUNTIME ENVIRONMENT



OUTPUT/DISCUSSION

Algorithms	Findings
KNN	• Finding the right value of K is a tough task. • Train time for this algorithm was the least. • Prediction time was highest for this algorithm.
SVM	• Train time was the highest for this algorithm. • Prediction time was the lowest for this algorithm. • Hyperparameter tuning was time taking .
Logistic Regression	• The optimizer failed to converge even after 1000 epochs. • Train time was high. • Prediction time was pretty low.
Decision Tree	• Hyperparameter tuning was very time taking. • Training time was pretty low. • Prediction time was pretty low.
Random Forest	• Hyperparameter tuning took a very long time. • Training this algorithm was very tough as the number of parameter was very high. • The predictions given were highly inaccurate.

REFERENCES

- BAHDANAU D, CHO K,BENGIO Y ,A Neural Conversational Model,Google Research paper.
- Jianfeng Gao, Michel Galley, Lihong Li, Neural approaches to conversational AI, Microsoft Research papers 2020.
- Alec Radford, Jeffrey Wu,Rewon Child , Language models are unsupervised multitask learners, OpenAI research 2020.
- Prasadhi Ranasinghe, Nipuni Chandimali, Chaman Wijesiriwardana, Systematic Exploration and Classification of Useful Comments in Stack Overflow IEEE explore,2021
- Aniket L Sulke1 Akash S Varude, Classification of Online Pernicious Comments using Machine ,MIT RESEARCH,2020
- Qing Yu, Ziyin Wang and Kaiwen Jaing,Tianjin, Research on text classification based on BERT-BiGRU MODEL, researchGATE,2021,VOL2.
- Kazi Saeed Alam, Cyberbullying Detection: An Ensemble Based Machine Learning Approach IEEE explore, 2020
- Alee radford,karthik narasimhan,tim salimans,ilya sutskever,Improving language understanding by generative pre-training.ALBERT : a lite bert for self-supervised learning of language,IEEE explore,2021