CYBER HARASSMENT DETECTION

CS 334 FINAL PRESENTATION

by Andrew Chung, James Park

OVERVIEW

01 MOTIVATION

Why we chose this topic + work done so far

02 PREPROCESSING

Standardize, binary output, tokenization, word-embedding

03 MODELS + METRICS

Which models we used + how they performed

04 TUNING + FUTURE

KNN and Decision Tree Hyperparameters + Improvements for future work

01

MOTIVATION

Why we chose this topic + work done so far

STATISTICS

37%

FELT CYBERBULLIED

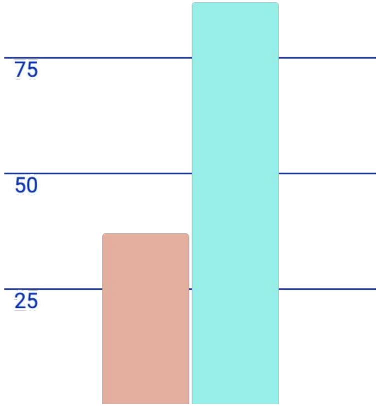
Middle + High schoolers

87%

REPORTED CASES

Have observed cyberbullying

- Decreased academic performance
- Lack of self esteem
- Suicidal thoughts



PREVIOUS WORK DONE

2015

- B. Sri Nandhini and J.I. Sheeba
- FuzZy learning algorithm
- Naïve classifier
- spring.me, myspace.com
- Outdated websites
  - Diction changes over time

2021

- Aditya Desai, Shashank Kalaskar, Omkar Kumbhar, Rashmi Dhumal
- Sentiment analysis
- SVM, Naïve Bayes, BERT
- TF-IDF
- Different approaches
  - Text classification
  - Word-Embedding
  - More transformer models

PREVIOUS WORK DONE

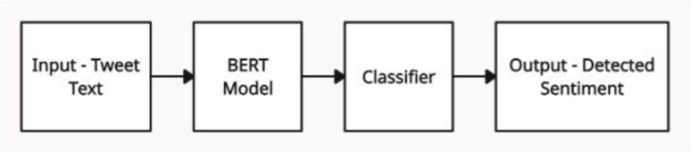


Fig.2. BERT model flow chart based on sentiment analysis

Table 1. Comparison of proposed approach with fuzzy classification rule

Dataset	Accuracy: Fuzzy classification rule	Accuracy: Proposed rule	F – Measure: Fuzzy classification rule	F – Measure: Proposed rule	Recall: Fuzzy classification rule	Recall: Proposed rule
Myspace	.35	.87	.44	.91	.60	.98
Formspring.me	.42	.86	.31	.92	.58	.87

Table 1. Accuracy of SVM and Naive Bayes from [3]

Classifier	Accuracy in percentage
Naïve Bayes Classifier	52.70
Support Vector Machine	71.25

Table 2. Accuracy of BERT Model

Classifier	Accuracy in percentage
Pre-Trained BERT (testing)	70.89
Pre-Trained BERT (training)	91.90

https://www.itm-conferences.org/articles/itmconf/pdf/2021/05/itmconf_icacc2021_03038.pdf

https://www.researchgate.net/publication/277568369_Online_Social_Network_Bullying_Detection_Using_Intelligence_Techniques

02

PREPROCESSING

Standardize, binary output, tokenization, word-embedding

DATA:

- From Kaggle
- **Features:** 47000 tweets labelled according to the class of cyberbullying
  - age, ethnicity, gender, religion, other type, not cyberbullying
  - One file divided into two columns: **text_type, cyberbullying_class**
- **Currently to deal with imbalanced data, we have removed cyberbullying data points to get an even ratio**

PREPROCESSING TASKS

STANDARDIZE

Remove NAs, stop words, special characters + lowercase

BINARY OUTPUT

Change classification from not CB and CB to 0 and 1

TOKENIZATION

Split texts into tokens for easier analysis

W-EMBEDDING

Train word2vec vectors from training

```
# Text preprocessing function
def preprocess_text(text):
    # Convert text to lowercase
    text = text.lower()

    # Tokenization (split text into words)
    # nltk is a package that allows users to acc
    words = nltk.word_tokenize(text)

    # Remove special characters, numbers, and pu
    words = [re.sub(r'[^a-zA-Z]', '', word) for

    # Remove stopwords
    stop_words = set(stopwords.words('english'))
    words = [word for word in words if word not

    return words
```

03

MODELS + METRICS

Which models we used + how they performed

MODELS:

Our general task was binary classification, and we decided to use the following models:

- KNN
- Decision Tree
- Logistic Regression Model
- BERT-base
- RoBERTa-base
- TWHIN-bert-base

METRICS:

- F1 scores
  - Allows for comparison between the two binary classifiers(0 and 1)
- AUC-ROC
  - Good for classification models
  - Shows how effectively the model differentiates between the two classes

CLASSIFICATION REPORT (RAW)

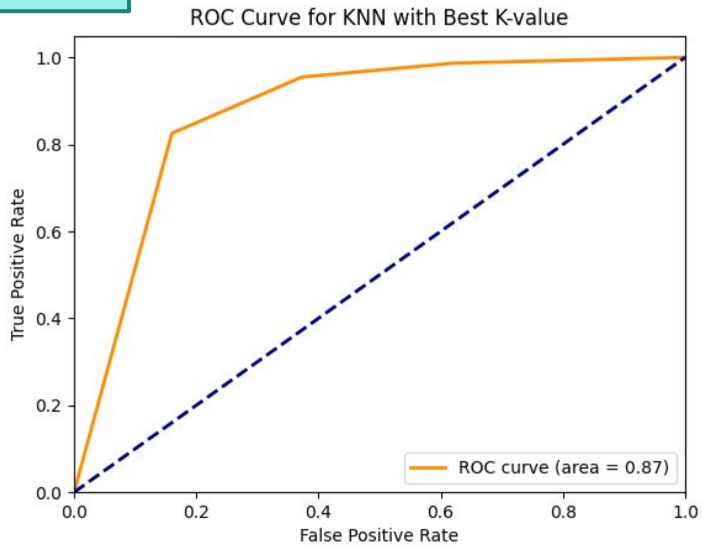
0/1	PRECISION	RECALL	F-1
KNN	.66/.84	.07/.99	.13/.91
DECISION TREE	.68/.88	.33/.97	.44/.92
LOG REGRESSION	.75/.83	.04/1	.08/.91
BERT	.73/.91	.52/.96	.61/.93
ROBERTA	.7/.91	.53/.95	.60/.93
TWHIN-BERT	1/.72	.60/1	.75/.84

CLASSIFICATION REPORT (EDITED)

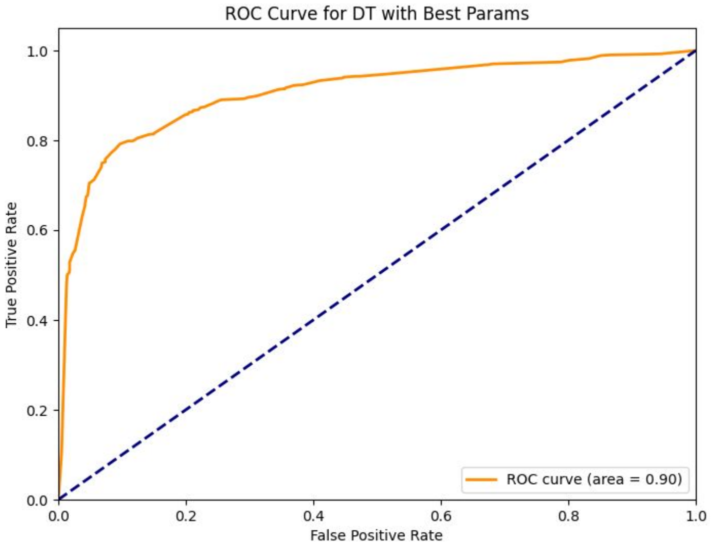
0/1	PRECISION	RECALL	F-1
KNN	.95/.71	.61/.97	.74/.82
DECISION TREE	.81/.89	.9/.97	.85/.84
LOG REGRESSION	.61/.68	.75/.53	.67/.59
BERT	.99/.99	.99/.99	.99/.99
ROBERTA	.99/.99	.99/.99	.99/.99
TWHIN-BERT	.99/.99	.99/.99	.99/.99

AUROC CURVE

KNN

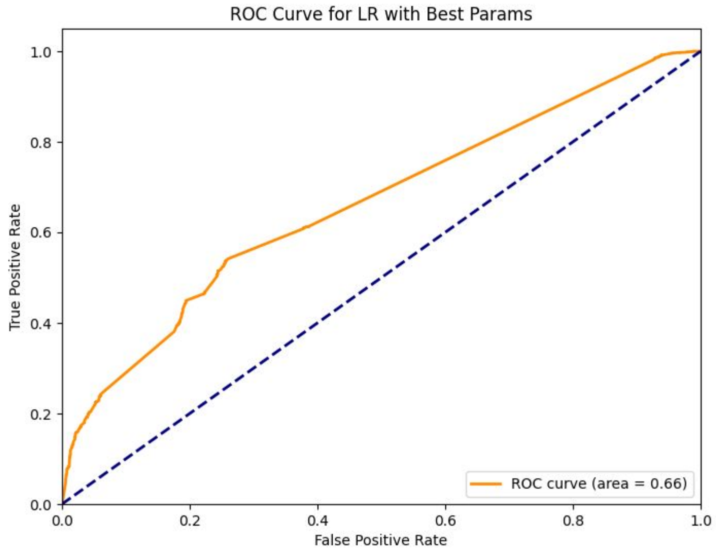


DECISION TREE

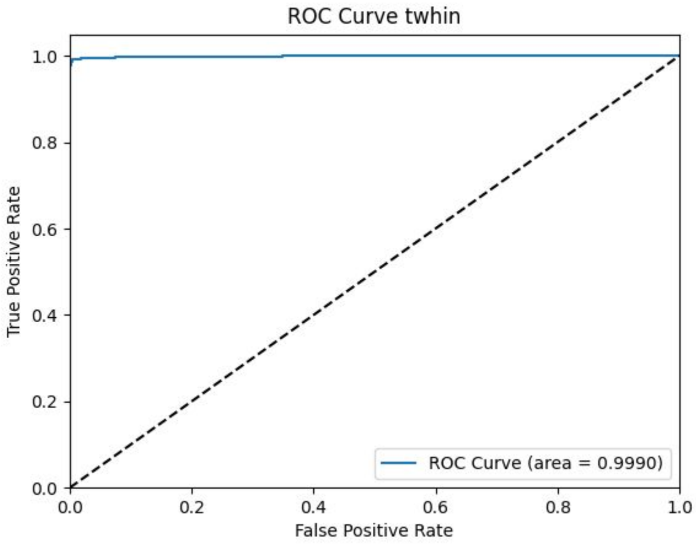


AUROC CURVE

LOGISTIC REGRESSION

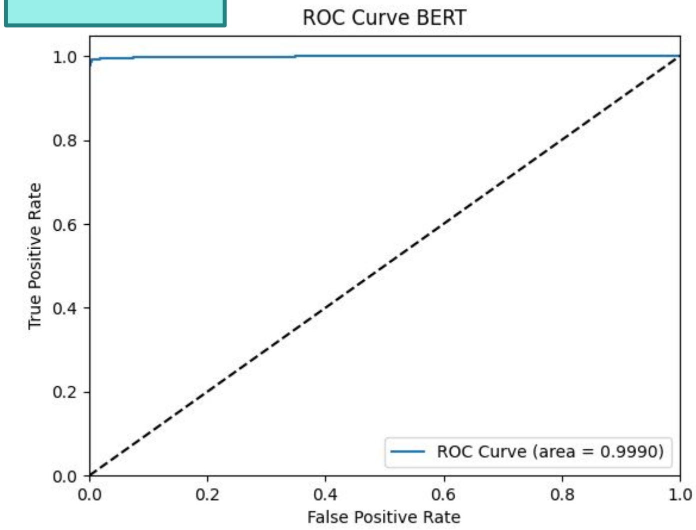


TWHIN

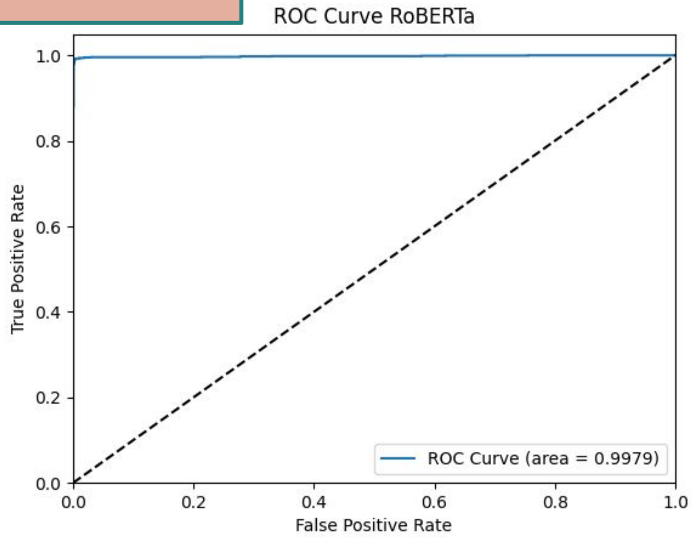


AUROC CURVE

BERT



RoBERTa



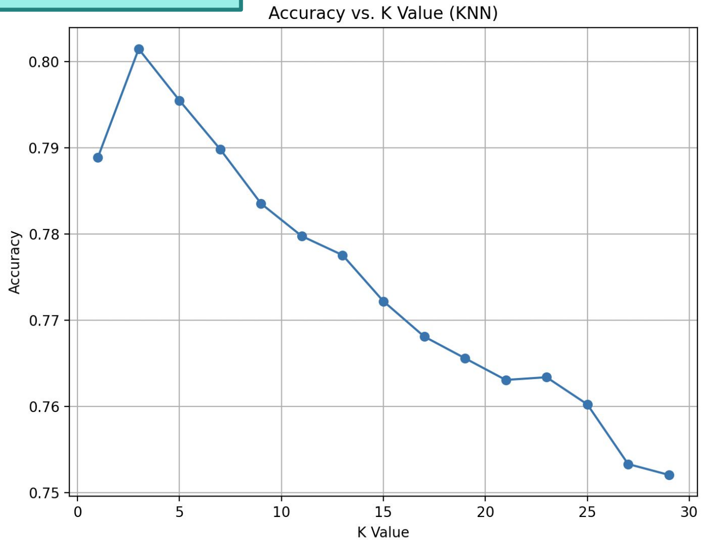
04

TUNING + FUTURE

KNN and Decision Tree Hyperparameters + Improvements for future work

KNN (EDITED)

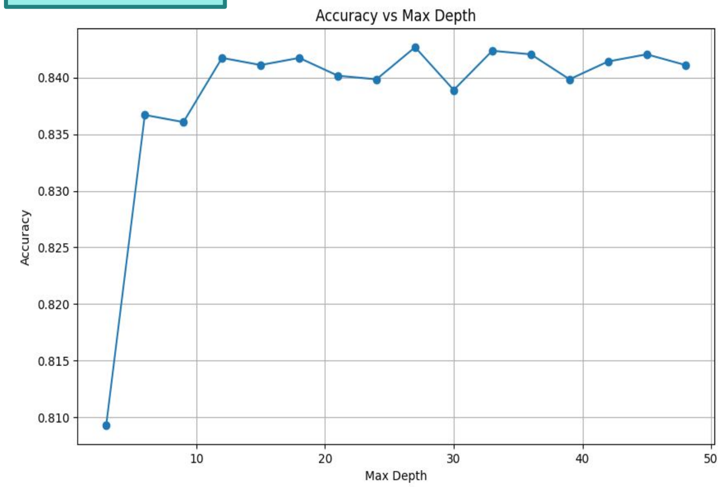
K-value



DECISION TREE (EDITED)

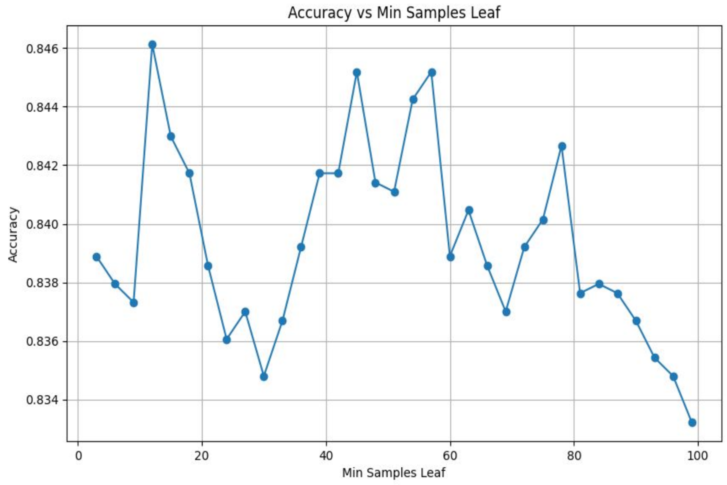
Depth

Min Leaf = 12



Min Leaf

Max Depth = 27



FUTURE WORK

01

Data Balancing

02

Hyperparameter Tuning

03

Transformer Model Verification

04

Realistic Use Case

THANK YOU!