Anh Tran
CS 315
Course Project Report

Cyberbullying Tweets Classification

December 10th 2022

Table of Contents

Introduction 2

Data Mining tasks 3

i. Input data

ii. Motivation & Problem Statement

iii. Questions

Technical Approach 5

Evaluation Methodology 9

Lesson Learned 12

I. Introduction:

With rise of social media coupled with the Covid-19 pandemic, cyberbullying has reached all time highs. Cyberbullying is a kind of bullying that uses electronic methods like mobile phones or the Internet to harass and send rudely, offensive, insulting, and hateful messages or emails to hurt other persons. Cyberbullying is a new kind of bullying that has many different characteristics from traditional bullying. Cyberbullying is characterized by deliberate and sustained peer aggression, as well as a power differential between the victim and the perpetrators or abusers. Cyberbullying can have a variety of consequences on victims, including mental health problems, poor academic performance, a tendency to drop out of work, and even suicidal thoughts. As social media usage becomes increasingly prevalent in every age group, a vast majority of citizens rely on this essential medium for day-to-day communication. Social media's ubiquity means that cyberbullying can effectively impact anyone at any time or anywhere, and the relative anonymity of the internet makes such personal attacks more difficult to stop than traditional bullying. People think they can get away easily with hateful comments on the Internet. Not only children and teenagers can become victims but adults also. It is even more concerning when a group of users, or a whole community joins to attack one individual on social media like Facebook, Instagram, Snapchat, Twitter, etc.

Cyberbullying is an urgent issue to many celebrities. Choi Jin-ri, known as Sulli of girl group f(x), was found dead in her home in Seongnam, South Korea by police, sparking a closer look at her career as a musician and a feminist who was often the target of cruel Internet trolls. After many times of standing up for herself, she could not survive the public attack of being “stood out” and “challenging the traditional norm”.
On April 15th, 2020, UNICEF issued a warning in response to the increased risk of cyberbullying during the COVID-19 pandemic due to widespread school closures, increased screen time, and decreased face-to-face social interaction. The statistics of cyberbullying are outright alarming: 36.5% of middle and high school students have felt bullied on the Internet and 87% have observed cyberbullying, with effects ranging from decreased academic performance to depression to suicidal thoughts.

II. Data Mining Task

Input data

The dataset was downloaded from the Kaggle as a csv file while my research on cyberbullying dataset. It consists of 47692 rows of tweets from Twitter. The tweets are labeled by cyberbullying type: “age”, “gender”, “ethnicity”, “religion”,“other_cyberbullying” or “not_cyberbullying”. The dataset was pulled from six different datasets on Twitter, and grouped by the cyberbullying type together.
I randomly selected 100 rows from the original dataset to keep it for testing. I split the remaining 47592 datasets into 80:20 in order to leave some for validation and accuracy scoring while creating models.

Motivation & Problem Statement

Approximately 90% of cyberbullying incidents go unreported. Hence, an intelligent, functioning, user-generated text detection system is essential. With all that being said, I was motivated to create models that can flag potentially harmful tweets, and break down the patterns of hatred. The main task is to classify if a tweet is cyberbullying. If yes, what type of cyberbullying will it fall into (age, ethnicity, gender or religion). Cyberbullying can be obvious and obscure at the same time. Obvious cyberbullying contains aggressive words in messages. It can be detected by using text analysis to

analyze the features of contents and words. In contrast, obscure cyberbullying text, for instance": your face is like a pig" is harder to classify since it may not contain red flagged bullying words. This indicates one of the challenges in cyberbullying classification based on textual features.

Questions

Some of the initial questions I set out as for the cyberbullying classification are:

- What is text analysis? What is sentiment analysis?
- What steps can be used to preprocess text data?
- What libraries installed within python that can be used for cyberbullying classification?
- What classifier would be optimal for the task classification?
- What are some common pitfalls to be aware of when it comes to cyberbullying classifications?
- Can we adapt neural networks into the dataset for classification since some of the cyberbullying text are obscure?
- What are the common bullying words in texts?

Some additional questions did arise as I went through the coding for this project, like why a particular classier returns the highest accuracy? Would the accuracy be higher if we just have a binary classified instead of multi class dataset? How are we suppose to do training and validation with just one dataset?

While some of which are not directly correlated to my project, it offered some insights on how I can follow up to improve my understanding of algorithms implementation.

Tasks Details

The initial goal is to build a working classifier that returns a high accuracy for the dataset. That objective involves comparing different sklearn built in classifiers: SVM, Naives Bayes, and Random Forest.

The first stage of the project was digging into the text classification analysis in the industry, and correlate it to what I have studied so far from the class. I referred to papers of data preprocessing, describing their use of data mining to filter cyberbullying texts,

recommended algorithms of classifiers, and popular train-test-split method to validate the models.

The entire project's codes are divided into these components:

+ Implement data exploration and preprocessing by stripping off noise from the texts
+ Remove outliers with uncommon length of word
+ Figure out what words are most common in cyberbullying tweet by using word cloud
+ Vectorize text into numbers
+ Apply statistical technique SMOTE to increase the number of examples from the minority class, in order to balance the data.
+ Compare classifiers with hyper parameter tuning
+ Visualize the testing accuracy of different classifiers using confusion matrix
+ Choose the best performing classifier and output the predicted labels of the test data

III.Technical Approach

The programming section consists of 3 different scripts I wrote.

Part 1: Preprocessing steps

After learning about the patterns underneath data, I dropped out the "other_cyberbullying" indices because it may cause confusion with other types of cyberbullying. The data shape is now reduced in rows. Next, I renamed the columns for simplicity.

There is a lot of noise within the tweets: URL link, mentions. hashtag, stop words, punctation, duplicates, etc. I moved forward to strip all those entities by using Regex module in Python, along with the NLTK library with corpus module. I applied these functions to the data frame. I proceed the preprocessing method with tokenization. In simple words, I split a sentence of tweets into separate words. At this point, I had to compare the accuracy of the model using Lemmatization versus Stemming for further preprocessing. Lemmatization considers the context and converts the word to its meaningful base form, whereas Stemming leads to incorrect meaning or spelling by removing last few characters from a word. It turned out Lemmatization on tokenized

words returns similar accuracy as Stemming. However, lemmatized preprocessing speeds up the runtime on my machine so I moved forward with lemmatization.
After finished up with preprocessed texts, I figured the majority of tweets are between 6 and 20 words by visualizing a Count plot from seaborn:

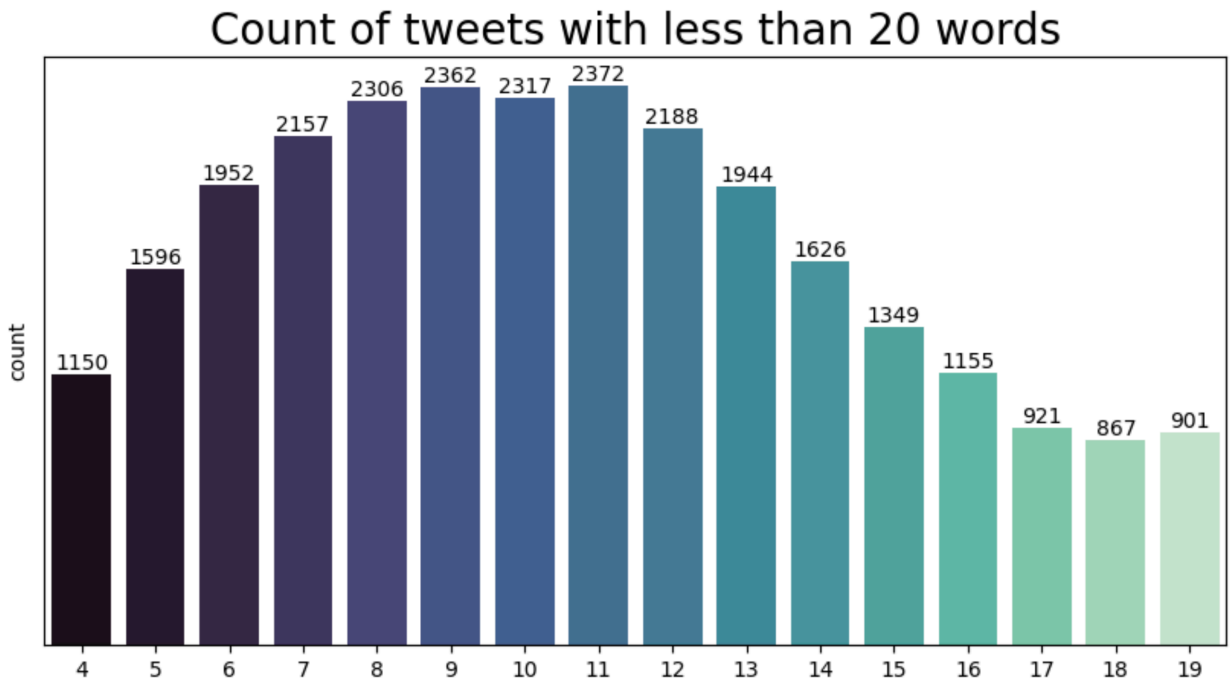


Figure 1

It was observed that some tweets have weird length of greater than 50 words. Those may later become outliers in the models. Therefore, I decided to remove tweets that are less than 4 words and more than 50 words. I created a Word Cloud to show which words are the most frequent among the given cyberbullying tweets (shown in Results section)
I vectorized the clean text into numbers based on the frequency of each word that occurs in the entire text. CountVectorizer function was applied in this case from sklearn.feature_extraction. The clean tweets are now transformed into a sparse count matrix. Next step is to transform the matrix to a normalized tf-idf representation. For my experiment, if a word appears more times in cyberbully tweets and appears less times in normal tweets, this word can represent the cyberbully as a feature.

In order to calculate the TF-IDF, I used the following expression:

$$W_{ij} = \frac{N_{ij}}{\sum_k N_{kj}} \times \lg \frac{D}{1 + S_{tj}} \tag{1}$$

where Wij is the TF-IDF score for a word, Nij is the count of a word appears in the document d and denominator is the count of all words appear in document d. Sij is the amount of document that contains word tj. Considering that the word may not appear in the corpus and denominator will be 0, we use 1 + Sij as the denominator.

One of the most important steps after splitting data into training is application of SMOTE technique from imblearn.over_sampling. Since there are imbalance data within the cyberbullying types, I applied the oversampling method to make duplicates of the data that is least present in the dataset. Below is the balanced data of cyberbullying types.

```
1    6292
0    6292
2    6292
3    6292
4    6292
Name: sentiment, dtype: int64
```

Figure 2. Cyberbullying types after SMOTE

This part answers the “preprocessing steps” in the Questions subsection.

Part 2: Choose classifiers with different hyper-parameters

I chose classifier's algorithms from sklearn documentation and configured different parameters. I then trained them on the training set and recorded the accuracy scores on the test set.

As for the classifiers, I chose:

- Random Forest, default parameters: n_estimators = 50
- Random Forest: n_estimators = 100
- Random Forest: n_estimators = 1000
- Naive - Bayes
- Support Vectors Machine: kernel = linear , C = 1
- Support Vectors Machine: kernel = linear , C = 2

In order to tune the parameters of random forest and SVM, I used multiple sets of parameters to measure the performance. Followed up is the cross validation score with Random forest and SVM(2 highest performers) to verify if models are overfitted. Number of folds are 2.

```
from sklearn.model_selection import cross_val_score

RF_cv_score = cross_val_score(rf_clf,X_train, y_train, cv=2)
SVM_cv_score = cross_val_score(svm_clf,X_train, y_train, cv=2)

print('Cross validation score (Random Forest Classifier):', RF_cv_score.mean())
print('Cross validation score (Support Vectors Machine):', SVM_cv_score.mean())
```

Figure 3. Cross validation accuracy

Part 3: Apply the model with the highest test accuracy to the unseen dataset.

(Recall from the input data section, I randomly left out 100 rows from the original cvs file to test with the optimal classifier model)

IV. Evaluation Methodology

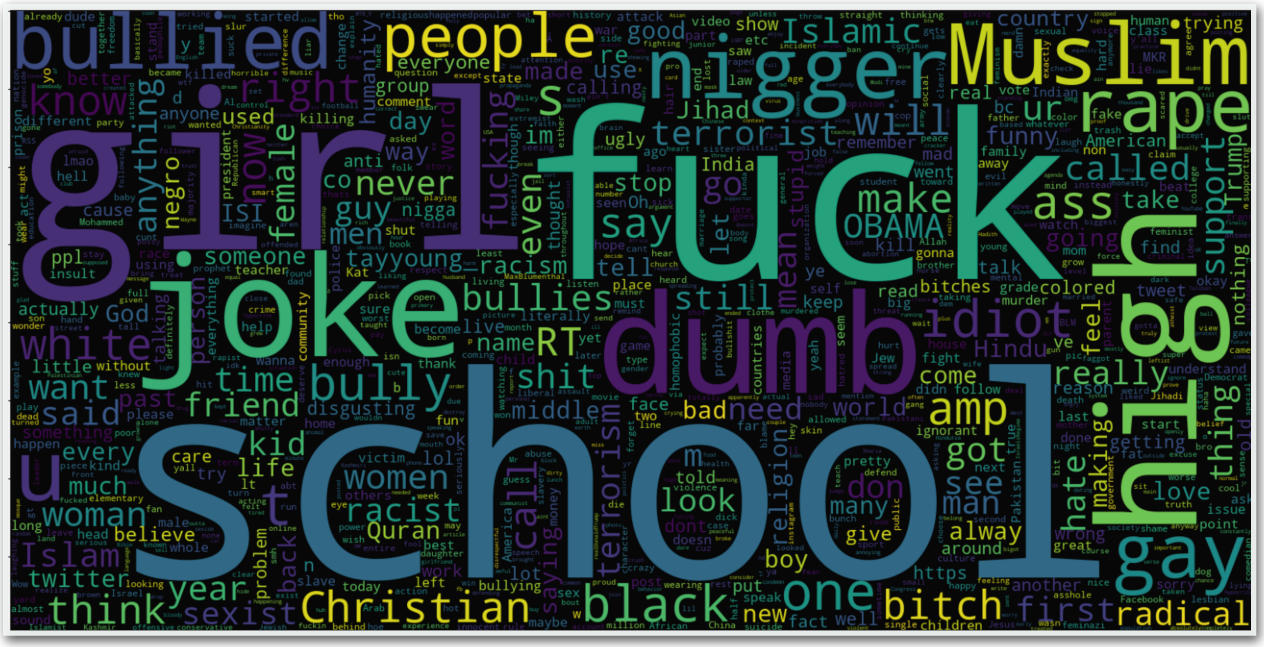
The input data for this project came from the Cyberbullying Classification dataset, published on Kaggle. The entire dataset contains a total 47,692 tweets that are extracted from various sources compiled on Twitter social media. The output of my data mining task would be a data-frame that includes the tweets, original cyberbullying labels, along with the predicted labels.

The optimal model was measured by the test accuracy on the validation set which was obtained from train test split. The confusion matrix represents a great evaluation metric as it implies the number of false positives and reasons behind misclassifications of cyberbullying tweets. I also evaluated my learning experience by finding answers to the problems listed.

The details of some answers I found for my questions are addressed in the Results section.

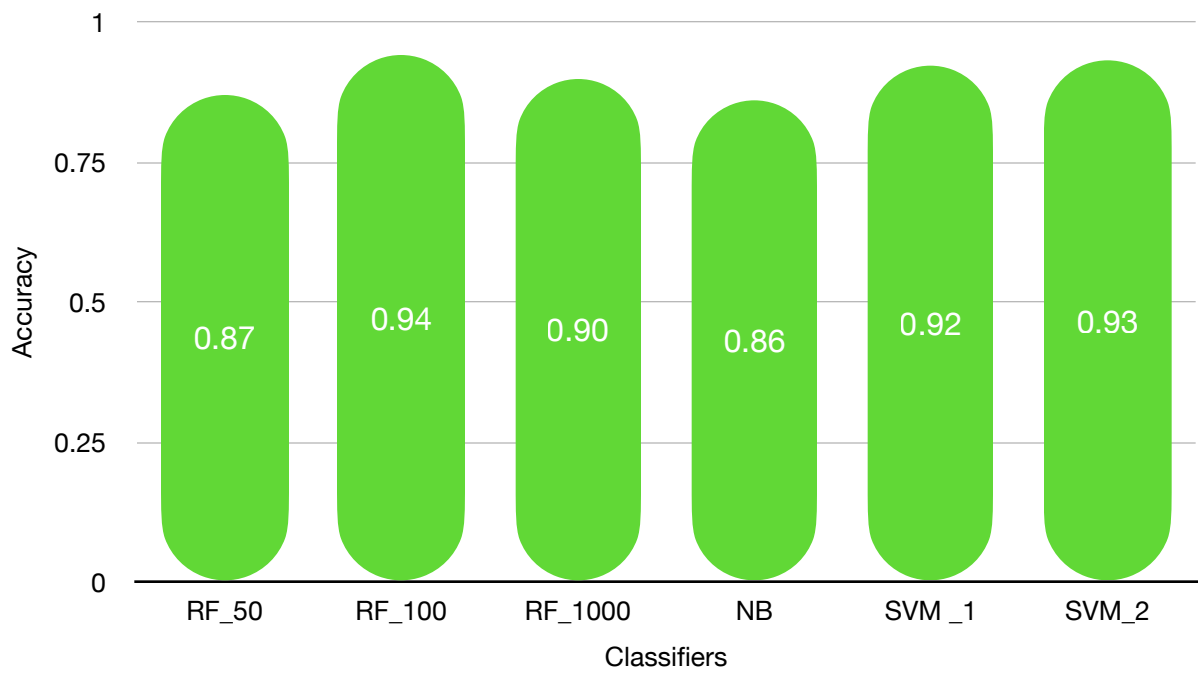
V. Results and Discussion

The results of Word Cloud building are shown below. These are frequent words that appear in age, gender, ethnicity, religion cyberbullying texts from the labeled dataset.


bullied
people
nigger
Muslim
girl
fuck
rape
joke
dumb
school
hate
gay
Christian
black
bitch
radical
bullies
bully
friend
terrorist
Islamic
ass
idiot
women
racist
Quran
Islam
think
sexist
twitter
woman
white
time
want
said
never
guy
men
someone
tayyoung
racism
terrorism
religion
OBAMA
Jihad
India
American
Trump
support
called
make
say
still
stop
go
shit
RT
amp
got
man
see
look
one
new
first
boy
give
many
don
know
anything
female
negro
kid
life
care
much
every
believe
year
problem
back
slave
today
left
bullying
saying
money
work
lot
post
put
wearing
speak
sex
bout
maybe
fact
well
single
children
Hindu
bitches
colored
read
tweet
feel
feeling
hate
making
thing
love
issue
point
truth
wrong
great
course
sorry
another
nice
cool
ask
dog
asshole
Facebook
lesbian
Democrat
Jihadi
Pakistan
Arab
Israel
Kashmir
Muslim
Kat
teacher
prophet
Mohammed
Republican
president
Obama
Hillary
Clinton
Trump
Hitler
Nazi
Jew
Jesus
God
Allah
Christian
Islam
Muslim
Jihadi
Pakistan
Israel
Kashmir
Arab
Hindu
Jew
Jesus
Allah
Hitler
Nazi
Obama
Hillary
Clinton
Trump
Democrat
Republican
President
Mohammed
Prophet
Kat
Teacher
Slave
Today
Left
Bullying
Saying
Money
Work
Lot
Post
Put
Wearing
Speak
Sex
Bout
Maybe
Fact
Well
Single
Children
Suicide
Million
African
China

Classifer	Accuracy
Random Forest , n_estimator = 50	0.87
Random Forest: n_estimators = 100	0.94
Random Forest: n_estimators = 1000	0.90
Naives - Bayes	0.86
Support Vectors Machine: kernel = linear , C = 1	0.92
Support Vectors Machine: kernel = linear , C = 2	0.93

Figure 5. Classifiers training accuracy graph

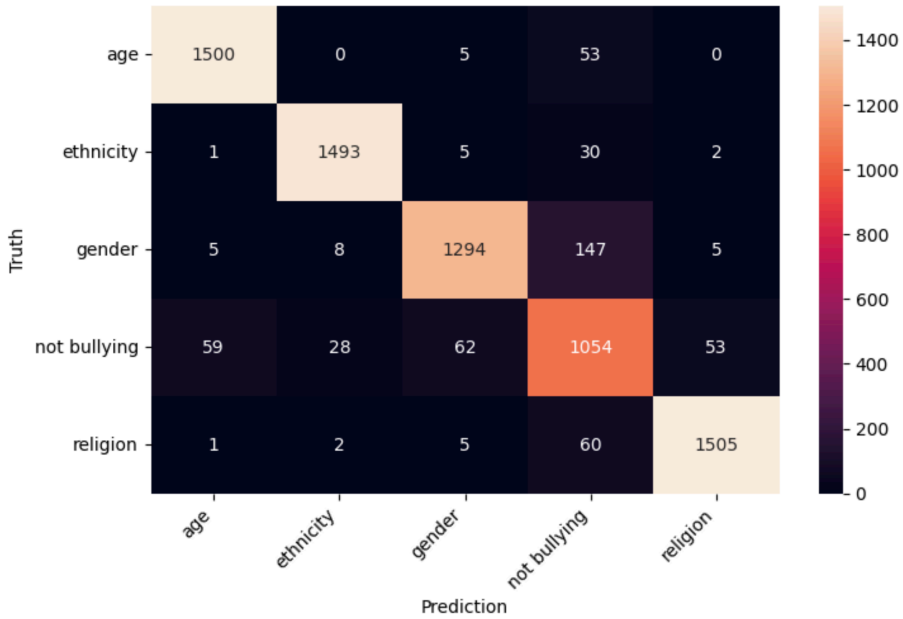


Observation:

+ SVM (C = 2 with linear kernel) and Random forest (n_estimators = 100) came out with high training accuracy of 93% and 94%. Based on the 2-folds cross validation, Random forest outperforms the SVM, thus it is the optimal model for Cyberbullying classification.

+ For the Random Forest, out of a of 7,377 test observations:
    - The number of correctly classified test examples is 6,846
    - The number of misclassified test example is 531

Figure 6. Random Forest Confusion Matrix



+ With regards to 2-folds Cross Validation, the optimal Random forest returns 93.64%, which is close to the test accuracy. It indicates that the training model is not overfitting.

The test accuracy of Random Forest on 100 test observations is shown below:

	text	type	rf_prediction
15	@yasmimcaci @Bferrarii PAREM DE FAZER BULLYING...	not_cyberbullying	not_cyberbullying
243	YES. I LOVE @lenadunham. http://t.co/ycgTBmmdxb	not_cyberbullying	not_cyberbullying
431	@comattwow @sweharris i have no idea what that...	not_cyberbullying	not_cyberbullying
1046	@ShabanaMir1 I saw Armstrong tell five lies in...	not_cyberbullying	not_cyberbullying
1306	Oh. there goes @wadhwa using his position to s...	not_cyberbullying	gender
...	...	...	...
45413	@InkUp4Lyfe: STINKY PUSSY ONCE AH MONTH BLEEDI...	ethnicity	ethnicity
45571	come say that shit to a NIGGERS face? no? shut...	ethnicity	ethnicity
46467	I can't believe them niggers called me dumb fr...	ethnicity	ethnicity
46857	Fuck you dumb ugly nigger name Wayne Positive ...	ethnicity	ethnicity
47560	“@RapeDrew: This is twitter. Imma speak freely...	ethnicity	ethnicity

87 rows × 3 columns

Figure 7. Sample prediction of random forest on test data

The trained Random Forest classifier yields 89.66% accuracy score, considering the fact these 100 test observations are split from the original dataset. Perhaps, test data might not be too different from the training data that the model was trained on.

Some of the questions that I set out to after the research:

1. Why is Random Forest most suited for text classification? Why better than SVM?

Our data contains multi classification problem, so SVM with linear kernel would not be as well suited as Random forest. SVM is intrinsically two-class, while Random Forest is intrinsically suited for multi-class problems. Random Forest builds the Bagging integration based on the decision tree and randomly select attributes in the training process of the decision tree. Compared to decision tree, it reduced overfitting problem.

2. What other techniques could be applied to the cyberbullying filter to increase the classification accuracy ?

I have done research on Google to expand my domain knowledge about this classification problem, and there were recommended many different algorithms. Since cyberbullying data can be obscure, as stated in the introduction, textual features may not be enough for classifying posts as bullying or normal ones. The study by Mitushi, et al. (2022) suggests using CNN-BiLSTM model with stacked word embedding. Hybrid deep learning architecture consists of convolutional neural networks integrated with bidirectional long short-term memory networks to classify social media posts into several bullying types related to gender, religion, ethnicity, age, aggression, and non-cyberbullying. Some studies also suggested using Google Tensorflows to perform text classification.

With regards to resampling data based on the minority, we can utilize the under-sampling method to test if it boots the accuracy.

VI. Lesson learned

I have learned so much about the power of machine learning into real life issues. Sentiment analysis plays an important application of machine learning. It is the use of natural language processing, text analysis, computational linguistics, and biometrics to

systematically identify, extract, quantify, and study affective states and subjective information.

This big project and other assignments in this class have offered me a great chance to reflect on what I have studied about data mining and data science, which relate very much to my summer Data Scientist Internship. For a senior student who barely gets the chance to carry out hands-on projects, I realized that asking the right questions and implementing series of testing to get a valid answer during exploratory steps are super important. Data mining, and Data science are all about experiment! It requires so much trials and errors before landing on a conclusion. I deeply learned how hard it was to implement the guidelines, a program from scratch to finish line, and more.

Overall, I appreciate the freedom that I was given during this project. I got carried away experimenting with the functionality of Regular Expression, as well as nltk toolkits. Those are all useful tools available in Python. In hindsight, I wish I could work more on developing a GUI to get users' input and implement cyberbullying classification.

Bibliography

Kasulis, Kelly. “Suicide of K-Pop Star Sulli Puts Spotlight on Cyberbullying.” *UPI*, UPI, 7 Nov. 2019, https://www.upi.com/Top_News/World-News/2019/11/07/Suicide-of-K-pop-star-Sulli-puts-spotlight-on-cyberbullying/4371573147309/.

“Internet Safety for Kids: Cyberbullying and Cyberharassment.” GCFGlobal.org, https://edu.gcfglobal.org/en/internetsafetyforkids/cyberbullying-and-cyberharassment/1/.

Raj, Mitushi et al. “An Application to Detect Cyberbullying Using Machine Learning and Deep Learning Techniques.” SN computer science vol. 3,5 (2022): 401. doi:10.1007/s42979-022-01308-5

“Sentiment Analysis.” Wikipedia, Wikimedia Foundation, 12 Nov. 2022, https://en.wikipedia.org/wiki/Sentiment_analysis.