JES
Journal of Engineering Sciences

Vol 13, Issue 02, FEB /2022

ISSN NO:0377-9254

A SURVEY ON TOXIC AND NON TOXIC WORD DETECTION

Email:jeevanaaembari@gmail.com

Name:Aembari Jeevana

M.tech-IT

JNTUH college of engineering,nachupally,jagtial

Guide:Asha Jyothi

Email: asha4prathap@gmail.com

Name:Asha Jyothi

Professor of IT department,

JNTUH college of engineering,nachupally, jagtial

Name: Suresh Kumar

HOD of IT department,

JNTUH college of engineering,nachupally, jagtial

ABSTRACT

Digitization and communication have experienced a paradigm shift as a result of the rise of social media. Initially, these tools were developed with the goal of bringing individuals from around the world together to exchange ideas and viewpoints. Since the start of the pandemic, a growing number of companies, educational institutions, students, and members of the general public have turned to these websites for information. For a long time, many have expressed concern about the rapid spread of social media sites like Twitter and Facebook. Internet users can also express their opinions on these platforms, which are promptly shared with the rest of the world. When it comes to disputing with those who don't agree with their viewpoints, many of the users on these platforms resort to using unpleasant, aggressive, and hateful language. Hate speech and other undesirable information has grown exponentially in recent years. Because the phrases "profane," "hateful," and "offensive" can all be used interchangeably, this type of content is referred to as "Toxic." A sizable chunk of our data consists of the exchanges between teenagers. This dataset has been analysed using NLP and a variety of machine learning techniques (SVM,RF,KNN,LR&Voting Classifier).

I. INTRODUCTION

Social media services such as Facebook, Twitter, Instagram, Youtube, and Snapchat are used by more than half of the world's population to stay connected and communicate with one another Public affairs and politics, as well as interpersonal

communication across vast distances, have been transformed thanks to the ability to create and share content with a large audience. It all contributed to the inciting of violence and the promotion of divisive views. Many of these social media platforms have a vested interest in garnering attention as part of their business model. As a result, the offensive material is brought to the attention of a wider audience and is therefore easier to hear. Poison spreads when people cannot understand and accept the viewpoints, thoughts, and opinions of those from different origins and races and socioeconomic classes. A large percentage of the population is targeted by anti-Semitism aimed at teenagers. The young people who are a part of these platforms make a huge contribution and receive a significant benefit. This has resulted in the majority of our collection being composed by tweets that are popular among young people.

Slang and abbreviations such as "wtf," "Asap," "Tbh," "Idc" and emoticons, as well as extensive repetitions as a kind of emotional emphasis in a remark, such as "Hate him sooomuchhh," all of this makes it difficult to grasp what the person is saying. Punctuation like "Shut Up You!!!!!" makes it harder to grasp what the person is saying.

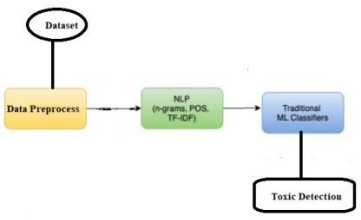
EXISTING SYSTEM:

- An inefficient and imprecise method of filtering data by hand is available. Automated technologies must be employed to complete this task more quickly and efficiently in order to save both time and labour. Machine learning (ML) has made it possible for us to analyse text semantically and make predictions while still being able to grasp the content.
- EXISTING SYSTEM DISADVANTAGES: It is labor-intensive and requires manual intervention.
- There has been a rise in riots and lynchings around the world because of the widespread dissemination of such content.

PROPOSED SYSTEM:

- For the objective of categorising literature into one of three categories: hazardous, nontoxic, or uncertain, we conducted study. To begin, we've joined two separate datasets to create the one we'll be using in this study. The use of ML classifiers for further investigation. LR, SVM, KNN, RF, and Voting are only few of the machine learning algorithms that use preprocessed data in the initial stage.
- The proposed system has a number of advantages:
- • A significant portion of our data set focuses on the most common topics of discourse among teenagers.
- With the use of machine learning, it is possible to identify potentially dangerous substances.

SYSTEM ARCHITECTURE:

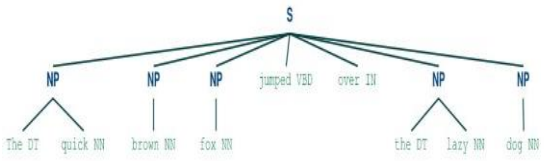


NLTK

A set of modules and applications for statistical and symbolic natural language processing (NLP) for English written in Python is known as the Natural Language Toolkit (NLTK). Steven Bird and Edward Loper of the University of Pennsylvania's Department of Computer and Information Science created it. Graphical demos and sample data are included in NLTK. Along with the toolkit, you'll find an educational text explaining the fundamental ideas behind the language processing jobs the toolkit supports, as well as a recipe book.

Research and education in NLP and related fields, such as empirical linguistics and cognitive science, artificial intelligence, information retrieval and machine learning are supported by NLTK.

[7] Since its inception, the NLTK has served as a teaching tool, an individual study tool, and a platform for prototyping and developing research systems. NLTK is used in classes at 32 colleges across the United States and 25 other countries. Tokenization, stemming, tagging, parsing and semantic reasoning are all supported by the NLTK.



Modules:

Process: Natural Language Processing is a human-machine collaboration that involves detecting and analysing vast amounts of text-based information. To get the most out of the text you're studying, you can turn to NLP. The WordNet corpus has been processed using the NLTK library. Additionally, stopwords have been removed, stemming, lemmatization and Part-Of-Speech tagging have been conducted throughout the data preprocessing procedure. We used the TF-IDF approach over the standard BOW approach to create the count vectorizer in order to get more accurate results.

Logistic Regression (LR): This is a machine learning algorithm. Especially in binary classification, this is a frequent approach to utilise when the dependent variable is categorical. It is the sigmoid function that serves as the basis for the logistic function, which lies at the heart of logistic regression.

Outlier identification, classification, and regression are all common uses of the Support Vector Machine (SVM) technique. By creating an N-dimensional hyperplane, it organises data points (represented by N features).

Classification and regression tasks can both be handled by the supervised technique known as Random Forest (RF). For example, this technique generates a more dense forest of trees, which means better results and less overfitting. Random forest employs MDI (mean reduction in impurity)

or Gini Value to determine the importance of each feature. Each of these algorithms is fed the count vectorizer obtained by the TF-IDF technique and the results are compared.

Algorithms:

i) SVM
ii) Random Forest Classifier
iii) Decision Tree Classifier
iv) KNN classifier
v) Logistic Regression
vi) Voting Classifier(adding Logistic regression, SVM, Decision Tree Classifier)

SCREENSHOTS

1) Importing Libraries

```
import numpy as np
import pandas as pd
import scipy
import matplotlib.pyplot as plt
import seaborn as sns
import re
from wordcloud import WordCloud, STOPWORDS
import nltk
from nltk.corpus import stopwords
from nltk.stem import SnowballStemmer, PorterStemmer
import math
from collections import Counter
from sklearn.feature_extraction.text import TfidfVectorizer, CountVectorizer
from sklearn.model_selection import train_test_split
from sklearn.metrics import roc_auc_score, roc_curve, auc, mean_squared_error
from sklearn.decomposition import TruncatedSVD, PCA
import gensim
import string
```

(**numpy, pandas**In order to preprocess data, SciPy contains modules for optimization, linear algebraics, integrations and interpolations, special functions, FFT and processing of signals and images, as well as ODE solvers and other frequent jobs in science and engineering, The visualisation of data is handled by matplotlib and seaborn, while the functions in this module let you to determine whether a given string matches a certain regular expression (or if a given regular expression matches a particular string, which comes down to the same thing), An example of a wordcloud is one that emphasises words that appear more frequently. In Python, the nltk library and sklearn model-building tool provide a set of tools for performing symbolic and statistical NLP on English text.

2) **Importing Dataset**

```
data = pd.read_csv('data.csv')
data.head()
```

Data Pre-processing

```
data.toxic.value_counts(normalize=True)
```

```
0    0.904156
1    0.095844
Name: toxic, dtype: float64
```

```
data_count=data.iloc[:,2:].sum()
```

(Calculating the toxic values)

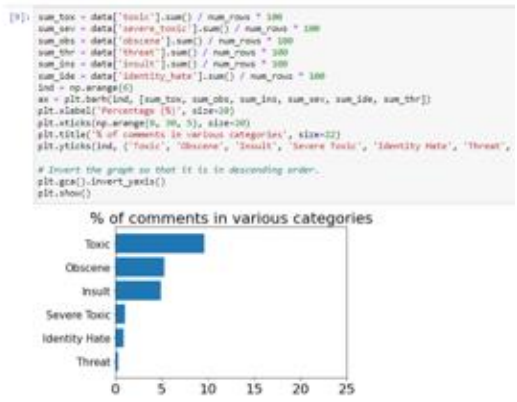
2) **Data Visualization**

(Representing the different comments per class).

Journal of Engineering Sciences

Vol 13, Issue 02, FEB /2022

ISSN NO:0377-9254



(Representing the comments in various categories).

3) Text Pre-processing

Text preprocessing

```
0]: import re
    import string

    # remove all numbers with letters attached to them
    alphanumeric = lambda x: re.sub('\w*\d\w*', ' ', x)

    # '[%s]' % re.escape(string.punctuation),' ' - replace punctuatio
    # .lower() - convert all strings to lowercase
    punc_lower = lambda x: re.sub('[%s]' % re.escape(string.punctuati

    # Remove all '\n' in the string and replace it with a space
    remove_n = lambda x: re.sub("\n", " ", x)

    # Remove all non-ascii characters
    remove_non_ascii = lambda x: re.sub(r'[^\x00-\x7f]',r' ', x)

    # Apply all the lambda functions wrote previously through .map on
    data['hate_speech'] = data['hate_speech'].map(alphanumeric).map(p

    data['hate_speech'][0]

0]: 'explanation why the edits made under my username hardcore metall
    n some gas after i voted at new york dolls fac  and please don t
    '
```

(Remove all the numbers, convert the strings to lower case, remove all '\n', and remove all the non-ascii characters)

4) Representing the most hate words

5) Assigning values to x and y, splitting the dataset into test and train

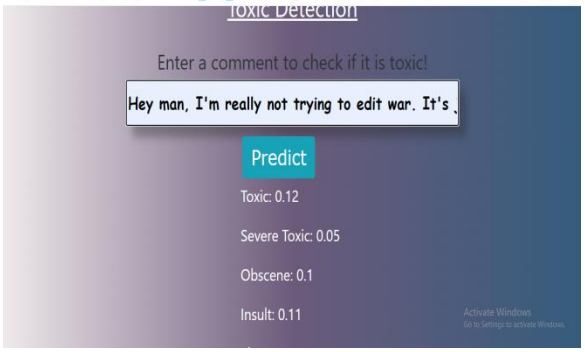
```
X = data_tox_done.hate_speech
y = data_tox_done['toxic']

X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=0.3, random_state=42)

# Initiate a Tfidf vectorizer
tfv = TfidfVectorizer(ngram_range=(1,1), stop_words='english')

X_train_fit = tfv.fit_transform(X_train)  # Convert the X data into a document term matrix dataframe
X_test_fit = tfv.transform(X_test)  # Converts the X_test comments into Vectorized format
```

Assigning hate_speech to x and toxic to y
Training – 70% and testing – 30%.

6) Applying the machine learning algorithms

i) SVM
ii) Random Forest Classifier
iii) Decision Tree Classifier
iv) Knn classifier
v) Logistic Regression
vi) Voting Classifier(adding Logistic regression, SVM, Decision Tree Classifier)

7) Model Comparison



CONCLUSION

Alone relies solely on data from toxic Twitter discussions. Before using LR and SVM, we pre-processed the data using a number of machine learning and ensemble approaches, including TF-IDF, POS tagging, and trigrams.

FUTURE ENHANCEMENT

Deep learning paradigms, such as convolutional and recurrent neural networks, as well as cutting-edge SVM models and feature extraction procedures used for toxic speech identification are all on our to-do list. We believe that the cascading strategy has the potential to succeed. It would be fascinating to improve the model's efficiency and accuracy, despite the fact that we only built a simple version.

REFERENCES

[1] T. Davidson, D. Warmsley, M. Macy, and I. Weber, "Automated hate speech detection and the problem of offensive language," arXiv, no. Icwsm, pp. 512–515, 2017.

[2] F. Del Vigna, A. Cimino, F. Dell'Orletta, M. Petrocchi, and M. Tesconi, "Hate me, hate me not: Hate speech detection on Facebook," CEUR Workshop Proc., vol. 1816, pp. 86–95, 2017.

[3] P. L. Teh, C. Bin Cheng, and W. M. Chee, "Identifying and categorising profane words in hate speech," ACM Int. Conf. Proceeding Ser., pp. 65–69, 2018.

[4] A. Mittos, S. Zannettou, J. Blackburn, and E. De Cristofaro, "'And we will fight for our race!' a measurement study of genetic testing conversations on reddit and 4chan," Proc. 14th Int. AAAI Conf. Web Soc. Media, ICWSM 2020, no. Icwsm, pp. 452–463, 2020.

[5] A. Olteanu, C. Castillo, J. Boy, and K. R. Varshney, "The effect of extremist violence on hateful speech online," arXiv, no. Icwsm, pp. 221–230, 2018. [6] P. Fortuna and S. Nunes, "A survey on automaticdetection of hate speech in text," ACM Comput. Surv., vol. 51, no. 4, 2018.

[7] Z. Waseem, J. Thorne, and J. Bingel, "Bridging the Gaps: Multi Task Learning for Domain Transfer of Hate Speech Detection," Springer International Publishing, 2018.

[8] A. Gaydhani, V. Doma, S. Kendre, and L. Bhagwat, "Detecting hate speech and offensive language on twitter using machine learning: An N-gram and TFIDF based approach," arXiv, 2018.

[9] Z. Zhang, D. Robinson, and J. Tepper, "Detecting Hate Speech on Twitter Using a Convolution-GRU Based Deep Neural Network," vol. 10843 LNCS. Springer International Publishing, 2018.

[10] T. Ranasinghe, M. Zampieri, and H. Hettiarachchi, "BRUMS at HASOC 2019: Deep learning models for multilingual hate speech and offensive language identification," CEUR Workshop Proc., vol. 2517, pp. 199–207, 2019.