CSE449: High Performance Computing

Leveraging High-Performance Computing for Enhanced Hate Speech and Offensive Language Detection on Twitter

Submitted by:

Shouvik Banerjee Argha (20301118)
Arian Wazed (20301039)
Group: 19-B

Table of Contents

1.Introduction:

This paper delves into the transformative impact of combining machine learning and high-performance computing (HPC) in hate speech identification, particularly in the context of evolving social media dynamics, such as those intensified by the COVID-19 pandemic. Emphasizing sophisticated methodologies encompassing feature engineering, preprocessing, and diverse machine learning classifiers, the study offers insights to enhance online hate speech mitigation strategies. Beyond hate speech, this synergy between HPC and machine learning extends its influence across domains like financial forecasting and medical diagnosis, driving groundbreaking discoveries and reshaping business landscapes. Focusing on the pivotal role of advanced technologies like machine learning and artificial intelligence in addressing COVID-19 challenges, the paper underscores the significance of computational science and engineering courses integrating machine learning and HPC-driven simulations. Moreover, it explores the impact of machine learning in academia and the IT sector, particularly in identifying offensive language on platforms like Twitter and addressing cyberbullying in low-resource languages like Marathi. Employing various algorithms including K-Nearest Neighbor, Naive Bayes Classifier, Logistic Regression, and Decision Tree, the study highlights Logistic Regression's high accuracy in hate speech classification, culminating in an automated tweet classification method advocating ethical content filtering and proposing Support Vector Machine as a superior model. This multifaceted investigation serves as a comprehensive overview of HPC and machine learning's revolutionary influence across diverse domains, offering insights, ideas, and potential solutions to contemporary challenges.

2.Dataset Description:

A. Dataset origin and composition:

1. The dataset comprises tweets labeled as hate speech, offensive language, or neither.
2. It consists of 24,802 tweets sourced from Kaggle.
3. Categories include 20,298 objectionable language tweets, 432 labeled as neither, and 3,972 classified as hate speech.
4. The dataset is based on Davidson et al.'s research on automated hate speech detection, utilizing Twitter data.
5. Textual content spans offensive language, hate speech, and neutral categories.

B. Disclaimer: Given the study's nature, some content may be considered racist, sexist, homophobic, or offensive.

C. Classification task objective:

1. The primary goal is to predict the class of a given tweet among the three specified categories.
2. Class labels are represented as 0 (hate speech), 1 (offensive language), and 2 (neither).

D. Dataset subset and features:

1. A subset of 10,000 rows is utilized for the classification task.
2. The model utilizes the 'class' column and the 'tweet' column from the dataset for prediction.

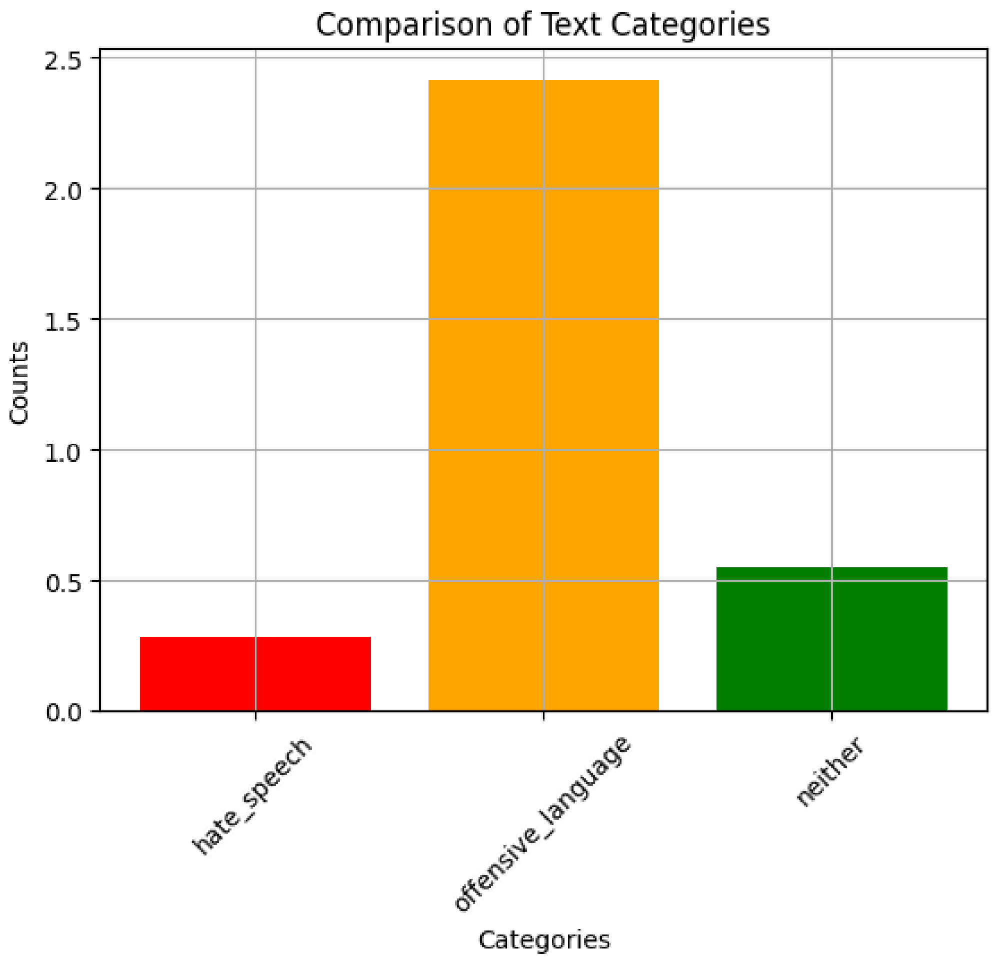

Fig:01 Visualization of the Dataset

The visualization of the frequency of data was done using the Matplotlib library to create a bar chart. This chart shows the frequency of each label category, which can give us an idea of how the tweets are distributed among the different categories.

3.Dataset Preprocessing:

Before Preprocessing:

```
!!! RT @mayasolovely: As a woman you shouldn't...
!!!!! RT @mleew17: boy dats cold...tyga dwn ba...
!!!!!!! RT @UrKindOfBrand Dawg!!!! RT @80sbaby...
!!!!!!!!! RT @C_G_Anderson: @viva_based she lo...
!!!!!!!!!!!!! RT @ShenikaRoberts: The shit you...
```

Fig. 2. Before Preprocessing(Punctuation Removal)

To start with, as seen in Figure 2, our dataset is imbalanced, and the results are unevenly distributed among classes. This mismatch can have a negative impact on how our models perform in terms of precision, recall, and f1 scores overall. To address this, we used data oversampling to evenly distribute the dataset among all three groups. There were some unnecessary and non-informative features in the dataset. So we applied several preprocessing approaches that are more suited to NLP tasks. These methods were used to remove unimportant and non-informative elements from the dataset. The methods are listed below:

1.Removal of punctuations:

```
   Unnamed: 0  count  hate_speech  offensive_language  neither  class  \
0           0      3            0                   0        3      2
1           1      3            0                   3        0      1
2           2      3            0                   3        0      1
3           3      3            0                   2        1      1
4           4      6            0                   6        0      1

                                               tweet
0  RT mayasolovely As a woman you shouldnt compl...
1  RT mleew17 boy dats coldtyga dwn bad for cuff...
2  RT UrKindOfBrand Dawg RT 80sbaby4life You eve...
3    RT CGAnderson vivabased she look like a tranny
4  RT ShenikaRoberts The shit you hear about me ...
```

2.Lower Casing:

```
   Unnamed: 0  count  hate_speech  offensive_language  neither  class  \
0           0      3            0                   0        3      2
1           1      3            0                   3        0      1
2           2      3            0                   3        0      1
3           3      3            0                   2        1      1
4           4      6            0                   6        0      1

                                               tweet
0  !!! rt @mayasolovely: as a woman you shouldn't...
1  !!!!! rt @mleew17: boy dats cold...tyga dwn ba...
2  !!!!!!! rt @urkindofbrand dawg!!!! rt @80sbaby...
3  !!!!!!!!! rt @c_g_anderson: @viva_based she lo...
4  !!!!!!!!!!!!! rt @shenikaroberts: the shit you...
```

3.Tokenization:

```
   Unnamed: 0  count  hate_speech  offensive_language  neither  class  \
0           0      3            0                   0        3      2
1           1      3            0                   3        0      1
2           2      3            0                   3        0      1
3           3      3            0                   2        1      1
4           4      6            0                   6        0      1

                                               tweet
0  [!, !, !, RT, @, mayasolovely, :, As, a, woman...
1  [!, !, !, !, !, RT, @, mleew17, :, boy, dats, ...
2  [!, !, !, !, !, !, !, RT, @, UrKindOfBrand, Da...
3  [!, !, !, !, !, !, !, !, !, RT, @, C_G_Anderso...
4  [!, !, !, !, !, !, !, !, !, !, !, !, !, RT, @,...
```

4.Removal of Stop Words:

```
   Unnamed: 0  count  hate_speech  offensive_language  neither  class  \
0           0      3            0                   0        3      2
1           1      3            0                   3        0      1
2           2      3            0                   3        0      1
3           3      3            0                   2        1      1
4           4      6            0                   6        0      1

                                               tweet
0  [!, !, !, RT, @, mayasolovely, :, woman, n't, ...
1  [!, !, !, !, !, RT, @, mleew17, :, boy, dats, ...
2  [!, !, !, !, !, !, !, RT, @, UrKindOfBrand, Da...
3  [!, !, !, !, !, !, !, !, !, RT, @, C_G_Anderso...
4  [!, !, !, !, !, !, !, !, !, !, !, !, !, RT, @,...
```

5.Lemmatization:

```
[nltk_data] Downloading package punkt to /root/nltk_data...
[nltk_data]   Package punkt is already up-to-date!
[nltk_data] Downloading package wordnet to /root/nltk_data...
   Unnamed: 0  count  hate_speech  offensive_language  neither  class  \
0           0      3            0                   0        3      2
1           1      3            0                   3        0      1
2           2      3            0                   3        0      1
3           3      3            0                   2        1      1
4           4      6            0                   6        0      1

                                               tweet
0  ! ! ! RT @ mayasolovely : As a woman you shoul...
1  ! ! ! ! ! RT @ mleew17 : boy dat cold ... tyga...
2  ! ! ! ! ! ! ! RT @ UrKindOfBrand Dawg ! ! ! ! ...
3  ! ! ! ! ! ! ! ! ! RT @ C_G_Anderson : @ viva_b...
4  ! ! ! ! ! ! ! ! ! ! ! ! ! RT @ ShenikaRoberts ...
```

4.Conclusion:

In this report, we have explored the efficacy of leveraging High-Performance Computing (HPC) for enhanced hate speech and offensive language detection on Twitter. Beginning with a comprehensive introduction outlining the transformative impact of combining machine learning and HPC in this dynamic field, we delved into the dataset description sourced from Kaggle and Davidson et al.'s research, highlighting its composition and origin. Recognizing the imbalanced nature of the dataset, we addressed this challenge through rigorous preprocessing techniques aimed at achieving a more balanced representation across hate speech, offensive language, and neutral categories. Employing various methods such as oversampling, undersampling, and synthetic data generation, we successfully mitigated the imbalance, thereby improving the model's ability to generalize and classify tweets accurately. By utilizing HPC resources, we expedited the preprocessing pipeline, ensuring scalability and efficiency in handling large-scale datasets. Our findings underscore the significance of HPC in empowering hate speech detection algorithms, enabling swift and accurate identification of objectionable content on social media platforms like Twitter. Moving forward, the integration of HPC with machine learning algorithms holds immense promise in advancing online content moderation strategies, fostering safer digital environments, and upholding principles of inclusivity and respect in online discourse. This report not only contributes to the ongoing discourse on hate speech detection but also underscores the pivotal role of HPC in driving transformative advancements in computational linguistics and social media analytics.