Identifying hate, abusive and racist comments on social media platforms

Project proposal for IS:567 Text Mining semester long project

Abstract

In today's internet-dependent world, social media platforms like Twitter, Instagram, and Facebook have interconnected the globe in a truly fascinating way, enabling instant communication between individuals across geographical boundaries, whether in the United States, Japan, India, or even the most remote locations. However, this increased connectivity has also led to a surge in online cyberbullying and the proliferation of inappropriate racial, sexist, and abusive slurs on these platforms. All these hateful comments are posted on social media in text form. Through this project, I aim to develop an automated identifier that detects such hate comments, which would help us in enabling the blocking of offending users or deletion of harmful content, or both.

1 Objectives

By employing various data pre-processing techniques and text mining algorithms, I intend to categorize my dataset into four primary categories: hate comments, racist comments, sexist comments, and other types of hate comments that do not fall under the aforementioned categories. Furthermore, I plan to utilize traditional text classification algorithms, specifically the Naïve Bayes classifier, to initially determine whether a comment is abusive or not. If deemed abusive, I will then apply a multi-class classification algorithm to identify which of the four categories the comment belongs to. Thus, I aim to implement a two-stage algorithmic approach on my dataset and compare the effectiveness of both algorithms to determine which one yields the most optimal results.

2 Source of my Data

The dataset for this project will be sourced from Kaggle, a renowned and reliable data repository, particularly well-suited for machine learning and text mining applications. I have chosen Kaggle due to its convenient data format; the dataset is already available in a CSV file and structured to meet my project requirements. Some example datasets, along with their online links, are provided below:

Count	tweet
0	!!! RT @mayasolovely: As a woman you shouldn't complain about cleaning up your house. & as a man you should always take the trash out...
1	!!!!! RT @mleew17: boy dats cold...tyga dwn bad for cuffin dat hoe in the 1st place!!
2	!!!!!!! RT @UrKindOfBrand Dawg!!!! RT @80sbaby4life: You ever f*** a b**** and she start to cry? You be confused as shit

Link of the data source :
https://www.kaggle.com/datasets/mrmorj/hate-speech-and-offensive-language-dataset

Apology for the in appropriate text content. This is something which would be occurring on a regular basis since this is the domain of the project working on.

3 Pre-processing Techniques being used

The primary preprocessing techniques I will employ are sentence segmentation and word tokenization, accompanied by the removal of special characters from the tweet column. Additionally, I will separate usernames from tweets and assign them to a new column in the CSV file. Furthermore, I will manually label the data into one of the four predefined categories to prepare it for algorithmic processing. Notably, I will not utilize stop word removal, as research indicates that this technique does not significantly enhance the performance of the Naïve Bayes classification algorithm, particularly in this context.

4 Algorithms

I will employ a two-stage classification approach. Initially, I will utilize *Naïve Bayes* classification to differentiate between hate comments and normal comments. Subsequently, if a comment is identified as hateful, I will apply *multi-class classification using neural networks* to categorize it into one of the specific labels: hate comment, racist comment, sexist comment, or other types of hate comments.

Naïve Bayes Classification

It The Naïve Bayes classifier relies on a simple, probabilistic approach, utilizing the bag-of-words representation to classify text. This representation can take two forms:

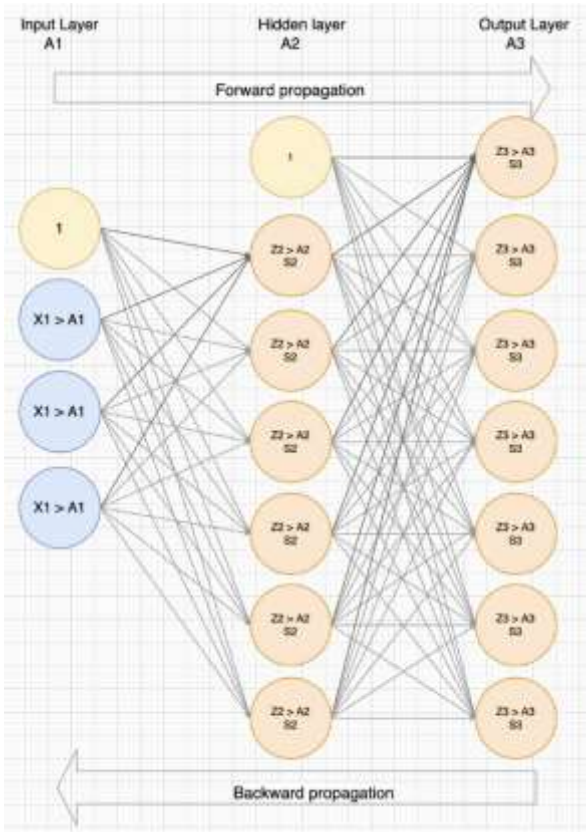
1. *Bag-of-Words (Binary):* It is also called the multivariate Bernouli model which uses simple 0 or 1 to represent if the word is present in the document or not.
2. *Bag-of-Words (Count-Based):* Also called as the multinomial model, here, the count of each word is stored which would help us know the relevance of the data present.

Multi-Class Classification using Neural Networks

Neural networks is an idea inspired from the field of neurobiology where a brain cell or known as the *neuron* is the fundamental unit of learning anything for humans. We take this idea and introduce neural networks which has one input layer, one output layer and at-least one hidden layer. The hidden layer has several activation functions which process the input data with some kind of biases or weights.

Neural networks comes under unsupervised learning category and thus requires no labeled data or training dataset to work with.

The multi-class classification using neural networks will take a statement or document in the input layer and will be classified to one of the labels present in the output layer. The figure below would is a visual representation of how the multi-class classification using NN will look like :



Picture taken from : https://towardsdatascience.com/the-complete-guide-to-neural-networks-multinomial-classification-4fe88bde7839

5 Outcomes of this Project

At the end of this project I intend to learn following things:

- Identify what are some of the most common key words of each label mentioned above and the frequency of that being used in the dataset and overall.
- Observe and note that whether the multi class classification be able to catch the

some of the subtle or indirect hate comments.

- Report the performance and accuracy differences between the using a traditional text classification technique and using an unsupervised learning algorithm both being used to classify text into labels.
- Observe the f1 score, precision, recall and confusion matrix of the two algorithms and report about it.