# Introduction

Automatic classification of textual content becomes the only practical method for effective data classification and insight. Although the ability to post comments empowers users to discuss news stories in a creative manner, discussion can also become toxic, leading to racist remarks and hate speech. Researchers have worked on sentiment classification on reviews and comments from social media and other news websites using machine learning techniques at the document level. The main objective of this research study is to investigate hate speech on the reddit platform using the lexicon-based approach. Reddit has its own principles and standards, organized around its communities called subreddits. Subreddits differ from each other in many ways, especially in four specific dimensions: topic, audience, moderation, and style. In our approach, the integration of two lexicon dictionaries was established. The two lexicon dictionaries integrated together were the Hatebase lexicons and the Vader lexicons. This integration created a new dictionary which was used to score comments extracted from four different subreddit and the values derived was used to compare the score derived only from Vader sentiment dictionary. Most studies on reddit comments and posts sentiment analysis are based on binary classification where the reviews are classified into “positive” and “negative.” Moreover, even the best systems currently obtain F1-score, precision, accuracy of only about 85%. There has not been as much work on sentiment analysis using lexicon-based techniques at the document level. However, recently there has been progress on building lexicons for sentiment analysis. Comparing the Vader lexicon-based technique and the integration of Hate base-Vader lexicon-based technique was conducted in this research work. The Vader sentiment lexicon is a library of pre-processed texts. It consists of words that have been categorized as either positive, negative, or neutral. Recently, it has been one of the most used lexicon libraries to perform sentiment analysis. The Hate base lexicon is a collaborative repository of multilingual hate words. It was developed to assist companies and research organizations moderate online conversations and use them as a hate speech predictor. In this work, we investigate top subreddit communities where there might be dissemination of toxic speech. We selected the subreddit communities based on a dataset that has been published in journals by different authors. We collected a large corpus of data, consisting of 100k comments for each of the subreddit used. The data collection process was systematically collected at different time intervals during this research. The results derived from the analysis show a significant difference between our integrated dictionary and the Vader lexicon library.

# Literature Summary

The Internet has become the main source for news acquisition and the use of various social media platforms and social networking applications has become an essential part of our daily life. Its growth is facilitated by the continuous use of the internet. Social media is an inspiring platform for online learning, exchanging ideas and sharing opinions. It is also used by researchers, scientists, and industry to conduct research on different fields ranging from social, political, medicine and various other fields. Social media involves individuals sharing opinions and ideas in form of texts, posts, status, and blogs. The process of analyzing each text in a sentence written in the tweet, text, or post to get certain information in the form of opinion is called sentiment analysis.

Ruwandika and Weerasinghe (2018) developed a model to detect hate speech using machine learning techniques. The author utilized both supervised and unsupervised machine learning techniques. The techniques used were Logic regression, Navies bayes, Decision tree and K-means. The author compared the results gotten from the algorithms with lexicon-based approaches, and the Navies Bayes classifier performed best with an F-score of 0.719. The dataset used were comments from News articles posted on the Colombo Telegraph website. 1500 preprocessed comment dataset were used (Ruwandika & Weerasinghe, 2018). Researchers and scientists lack a general understanding on what type of content attracts hateful discussions and the possible effects of social networks on the commenting activity on news articles.

Pereira-Kohatsu, Quijano-Sánchez, Liberatore, and Camacho-Collados (2019) proposed a system called HaterNet, which has taken a gold standard. This system identifies and monitors the evolution of hate speech in twitter. The system is currently utilized by the Spanish National Office Against Hate Crimes of the Spanish State Secretariat for Security. The dataset used was an initial corpus of more than two million tweets. Twitter's API Rest was used to download these tweets. The dataset was manually tagged and carefully filtered to produce both the training and testing datasets. For the Feature extraction, an NLP preprocessing pipeline was followed. This involved tokenization, POS tagging and lemmatization. The author applied filter method-based features for the feature selection combined with a LASSO model (Least Absolute Shrinkage and Selection Operator). After all necessary pre-processing was done, a newly developed dataset consists of 6000 expert-labeled tweets. Supervised machine learning classifiers were used to develop a model in which the dataset was trained. The best approach amongst the classifiers consists of a combination of a LTSM+MLP neural network that takes as input the tweet’s word, emoji, and expression tokens’ embeddings enriched by the tf-idf and obtains an area under the curve (AUC) of 0.828 on our dataset, outperforming previous methods presented in the literature (Pereira-Kohatsu et al., 2019)

Zampieri et al. (2019) proposed a prediction monitor to target offensive posts on social media. In the paper, the authors compiled an Offensive Language Identification Dataset (OLID), that is a manually curated dataset. The Twitter API was used to extract these comments from tweets. The tweets were annotated for offensive words, this was achieved using a three-layer annotation scheme. The authors carried out a round of trial annotation of 300 instances with six experts using nine keywords. The aim of the annotation was to evaluate the proposed tagset and data retrieval method. It creates a gold standard with instances that was used as test questions to ensure the quality of the annotators for the rest of the data. This new dataset was used to compare with pre-existing datasets for hate speech identification. The authors highlighted the similarities and differences that occur when training both datasets. Further experiment was carried out to compare the performance of different machine learning models on OLID (Zampieri et al., 2019)

Kumaresan and Vidanage (2019) proposed a system that aims at improving the detection of hate speech. The social medium platform used for the experiment was twitter. The authors utilized both ontologies and fuzzy logic approaches combined with sentimental analysis to determine hate speech and deconstruct the ambiguity present. The tweets were classified into hateful, offensive, and neutral. The dataset used was derived from previous research work which has been published. The system achieved an F1-score higher than previous research work. The F1-score achieved was 0.677 for the Hate class and 0.9805 for the offensive classification. The Vader sentiment library was utilized for the sentiment analysis. It was developed to assist companies and research organizations moderate online conversations and use them as a hate speech predictor. Reddit has its own principles and standards, organized around its communities called subreddits. Subreddits differ from each other in many ways, especially in four specific dimensions: topic, audience, moderation, and style (Kumaresan & Vidanage, 2019)

Pitsilis, Ramampiaro, and Langseth (2018) developed a detection scheme using Recurrent Neural Network (RNN) classifiers and incorporated user-related information. These data were fed as input to the above RNN classifier along with the word frequency vectors derived from the textual content. Long Short-Term Memory Network (LSTM) was the RNN classifier used. The model used consisted of four layers. The input layer, the hidden layer, the dense layer and the output layer. In the input layer, the number equals the size to the word vector plus the number of additional features. The word vector dimension was set to 30 so that to be able to encode every word in the vocabulary used. The hidden layer was made up of the sigmoid activation, which is connected to the input layer and the dense layer. The dense layer was just an additional layer used to obtain a more stable result. The ReLU activation function was utilized in this layer. The output layer provided output in the form of probabilities for each of the three classes Neutral, Racism, and Sexism. The softmax activation function was used for this layer. The author’s scheme was evaluated on a publicly available corpus of 16k tweets, and the results demonstrate its effectiveness in comparison to existing state of the art solutions. The results derived from incorporating features related to user’s behavior into the classification has provided a significant increase in the performance vs. using the textual content alone, F = 0.9295 vs. F = 0.9089 (Pitsilis et al., 2018).

Gitari, Zuping, Damien, and Long (2015) utilized a lexicon-based approach to detect hate speech in web blogs and comment sections. The aim of their research was to create a model classifier that uses sentiment analysis techniques and in particular subjectivity detection to not only detect that a given sentence is subjective but also to identify and rate the polarity of sentiment expressions. They started by whittling down the document size by removing objective sentences. Then, using subjectivity and semantic features related to hate speech, we create a lexicon that is employed to build a classifier for hate speech detection. In order to perform subjective sentence detection, they employed a rule-based approach that classified sentences relying on a lexicon of well-established clues. They utilized two known sentiment lexicon resources of (Riloff & Wiebe, 2003) and SentiWordNet package in python. There has not been as much work on sentiment analysis using lexicon-based techniques at the document level. However, the authors of this paper made progress on building lexicons for sentiment analysis (Gitari et al., 2015).

Graumas, David, and Caselli (2019) proposed a method that generated polarized word embeddings using controversial topics on Twitter as proxies for interactions among social media communities that may be liable to use abusive language. The authors obtained their datasets from literature published. They derived three datasets from the literature and used these datasets to train and test the models developed. Two data sets explicitly consisted of words and sentences in the category of hate speech, while the other words consisted of a broader category of offensive language. The results of their experiments, based on simple linear SVM models, showed that the word embeddings, both generic and polarized, outperform n-grams based models across data sets, showing better generalization capabilities, although they fail to outperform such models in the same data distribution scenario(Graumas et al., 2019).

Tan and Lee (2015) proposed a study which examined three aspects of multi-community engagement in Reddit. There was a sequence of communities whereby users post the language that users employ in those communities, and the feedback that users receive, using longitudinal posting behavior on Reddit platform as the main data source, and DBLP for auxiliary experiments. During the investigation, the authors found that over time, users span more communities every 10 posts, “jump” more, and concentrate less. Fairly-operational users seem consistently less “adventurous” than continuous operational users even, notably, from the very beginning. Curiously, Fairly-operational users imitate continuous operational users in the top activity quartile. The authors demonstrated the effectiveness of features drawn from these aspects in predicting users’ future level of activity (Tan & Lee, 2015).

Aggarwal, Gola, and Sankla (2021) proposed an expert model for hate speech detection works towards overcoming the strong user-bias present in the available annotated datasets. The authors utilized A-Stacking hybrid classifier based on ensemble learning that was used clustering to form weak hypotheses that was systematically integrated using a meta-classifier in a later stage. The model was adaptive with the properties of the dataset which was used to generate the hypotheses used as base-classifiers. The Waseem and Hovy dataset were used and it came with tweet identifiers along with their associated class labels, i.e., sexist, racist and non-hateful. The actual tweets were extracted using any tweet crawler. 16k tweet identifiers constitute the dataset. The results show that the proposed model could adapt to the properties of data and behave accordingly when the test environment is changed. The authors emphasized that the available annotated datasets have a strong bias in them. For the correct assessment of the model, it is necessary to restrict the number of tweets per user (Aggarwal et al., 2021)

Gaydhani, Doma, Kendre, and Bhagwat (2018) proposed an approach that automatically classified tweets on Twitter into three classes: hateful, offensive and clean. They utilized Twitter dataset, to perform experiments considering n-grams as features and passing their term frequency-inverse document frequency (TFIDF) values to multiple machine learning models. They utilized three machine learning algorithms for text classification: Logistic Regression, Naive Bayes and Support Vector Machines.They utilized the Scikit-learn package in python for the implementation. Logistic Regression had the best performance, so it was used to evaluate the test data. The authors observed that the recall value for offensive tweets was 0.93, which signifies that 7% of the offensive tweets were misclassified by the model. The precision for the hateful class is 0.94, which signifies that 6% of both clean and offensive tweets were classified as hateful. On the other hand, the recall for clean class is 0.98, which is significantly better. The authors performed a comparative analysis of the models considering several values of n in n-grams and TFIDF normalization methods. After tuning the model giving the best results, it was achieved with a 95.6% accuracy upon evaluating it on test data(Gaydhani et al., 2018).

Arulmurugan, Sabarmathi, and Anandakumar (2019)utilized an approach to predict sentiment polarity of text toward a specific aspect. Although existing neural network models show promising performances on ABSA, their capabilities can be unsatisfactory in cases where the amount of training data is limited. The authors used multiple dictionaries and knowledge sources to improve the system which they developed. The objective of the system was to perform aspect-based sentiment analysis (ABSA), which utilized multiple sources of text knowledge to predict sentiment. BiLSTM modelling layer was utilized as an attention mechanism that calculates important scores after encoding the contextual information of text into representations of words at each time step. Structure knowledge is extracted via clause recognition and fused into the model through the BiLSTM layer. However, not all words are equally important in terms of expressing sentiment toward a specific aspect. That is why the attention mechanism was infused in this approach. Sentiment knowledge is exploited by means of training a general classification model with the sentiment labels of documents and fused through pretraining specific layers to extract contextual features and predict sentiment polarities more accurately (Arulmurugan et al., 2019).

Melton, Olusanya, Ammar, and Shaban-Nejad (2021)carried out research on the social media platform called reddit. They were investigating if the social media platform had any role in the GameStop short squeeze in 2021. The subreddit used for this research was the r/WallStreetBets. The discussion held during the time of the event was used for the analysis of the American online retailer (GameStop). Over 10.8m comments were extracted. The comments were divided into two categories in order to contain the sentiments. The long messages are threads containing more than 2k comments, while the second group short messages are threads containing less than 2k comments. Sentiment analysis was performed using the lexical approach. The Vader sentiment analyzer was the lexicon dictionary used to assess the sentiment of phrases and sentences in the comments, without the need of looking at anything else. The sentiment analyzer extracts the polarity scores and provides the overall sentiment metrics (compound score) for the comments. The compound score is greater than 0.05, it denotes a positive sentiment. When the compound score is less than -0.05, it denotes a negative sentiment. For the compound score lies between 0.05 and -0.05, it denotes a neutral sentiment. To further assess the connection between Reddit investors sentiments and GME price changes the authors employed the wavelet coherence framework. This method utilized here is powerful for analysis of shorter observation periods and can help to identify time-frequency co-movements between selected variables adding to the regression analysis results. From the results, it shows that both tone and number of comments influence GME intraday returns. Sentiments extracted from longer threads have a greater influence. "Fear" is the dominant sentiment in all comments, while comments that express a "Sad" sentiment show the most significant impact (Melton et al., 2021).

Machova, Mach, and Vasilko (2021) proposed an offensive language detection system to analyze over 50,000 right wing German hate tweets posted between August 2017 and April 2018, at the time of the 2017 German federal elections. The authors used both quantitative and qualitative methods to analyze this large corpus of data. The dataset was divided into 5 categories, First and Second level incitement speech category, insult speech category, First and second level slander category speech. The authors used a combination of qualitative approaches with quantitative techniques from Natural Language Processing (NLP). To evaluate the qualitative approach, they analyzed a random subsample of 2,000 tweets, then compared the results to support quantitative evidence. They focused on a selection of hate speech tweets. They utilized the SentiWS lexicon for German. It assigns scores to words like gut = +0.37, and schlecht = −0.77, which was used to compute an average score for a given text. They computed the average score for all tweets resulting in about 32% of the hate tweets predicted as negative, against 22% of the safe tweets, or a 10% difference. They utilized Character trigrams to efficiently model linguistic variation such as spelling errors, word inflections, function words. In order to evaluate the Perceptron algorithm used, they did a cross evaluation with a hold-out set of the 1,000 most offensive examples in the hate speech dataset (Machova et al., 2021).

# Methodology

In this paper, the most popular microblogging platform Reddit is used. Just like twitter, reddit is a social media platform where users post about several topics. It has sub-categories called subreddit, this is for different discussions held on the social media platform. Some news websites also use social media platforms to pass across news, and users are allowed to comment on them. However, comment moderation remains an issue. The increasing magnitude of user-generated comment repositories and their continuing fast growth makes it very labor intensive to manually monitor and extract sentiment from user-generated content. The first step in any data analytics research is data collection. Data collection follows a certain procedure, this is a systematic process of extracting the right data needed for your analysis.

## Data Collection

In this work, we investigate top subreddit communities where there might be dissemination of toxic speech. We selected the subreddit communities based on a dataset referenced by Tan and Lee (2015). This dataset contains posts and comments from the inception of reddit in January 2006 to December 2018. Some subreddits are topic specific (r/Nickelodeon, r/PBSkids, r/Disney, r/4chan), while others are from a general topic (r/AskReddit). All subreddits have rules regarding what types of content is allowable in that community, the specificity of the rules and the level of moderation differ greatly from subreddit to subreddit. Even when the written standards are similar, reddit communities attract users with different interests and discussions of different nature (Tan & Lee, 2015). Similar data collection procedure was followed by Yan and Liu (2021), in their research the authors extracted the post and comment data from specific subreddit from August to November in 2019 and 2020, respectively, representing the pre-pandemic and pandemic periods, using the Pushshift API. Chen and Sokolova (2021) utilized the API to extract text posts from ‘r/depression’. The subreddit posts conveyed self-expressed contextual aspects of depression and provide a richer context for sentiment analysis than more treatment-oriented posts from r/Anxiety and r/PTSD, the other mental health Reddit subcommunities (Chen & Sokolova, 2021).

The PushShift Reddit API was used to extract the comments from the necessary subreddits we needed. The reason for using this API is because it has been utilized in previous research work. The PushShift API was designed and created by the /r/datasets mod team to help provide enhanced functionality and search capabilities for searching Reddit comments and submissions. It provides over four billion comments and submissions since the inception of Reddit. (https://github.com/pushshift/api). There are two main endpoints of the API, one of them is for extracting only the comments, while the other is for extracting the submissions. The comment search parameter can be time-based, the reason is that the API allows users to set a time frame of when the comments will be collected. This function can be activated when writing the code, this is done by using an epoch time for the value of the “after” and “before” parameter, it will return all comments with a created\_utc epoch time greater than that value. The comment search can be extracted by hour of the week, this parameter itself can be used directly to limit comments by a specific hour of the week as well. The range is from 0 to 168 with 0 being midnight on Monday and 168 being the 23’rd hour of Sunday night. The comments can be extracted by hour of the day, using this parameter as an aggregation type, it shows when a subreddit or author is active throughout a typical day Baumgartner, Zannettou, Keegan, Squire, and Blackburn (2020).We collected a corpus of 100k comments from each of the subreddit mentioned above. This data came in large files that were put into a csv using panda dataframe in python. These comments extracted for analysis were extracted using the pmaw package in python. This package ensured a large collection of comments at once. We utilized the function after and before for extracting the comments because it was mostly historical comment data. We extracted from the inception year May 2005 to May 2022. The extraction time for each subreddit extraction was about 30minutes.

## Data Pre-processing

In this research study, after collecting a huge corpus of data for each subreddit, the data needs to be arranged and cleaned before further text processing is performed. The data drop function in the panda data frame will be applied to the comment data in the csv. The data drop function applied drops the attribute columns not needed for the analysis. The figure 1 shows source code written python for the execution. Due to the large amount of data the cleaning process took 1min 30sec to complete each corpus. The figure 2 provided below shows the extracted comments for one of the subreddit and the figure 3 shows the source code used to clean the data.



Figure 1: Data drop function

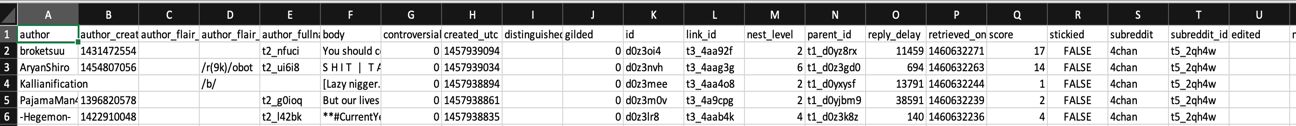


Figure 2: Comments showing attributes of the data.

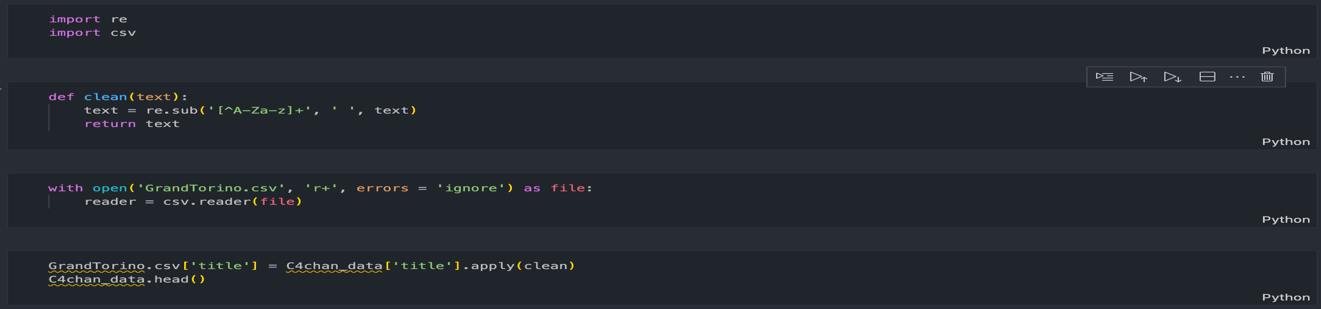


Figure 3: Source code for cleaning comment data

## Vader Lexicon and Scoring Mechanism

Vader sentiment analyzer is a lexicon library which can be installed on python as a module. It is the most common sentiment analysis tool for social media text and text data. This module is based on the lexicon and rule-based methods. We utilized Vader sentiment analyzer because it is fast and accurate, it gives the score of a comment as either “negative”, “positive” or “neutral”. The compound score of Vader is a normalized score between 1 and -1 which is obtained by adding the valence scores of each category score mentioned above and is adjusted depending on the rules. The scoring mechanism of the compound score is explained below

* 0.05< Compound score > -0.05 = Neutral Sentiment.
* Compound Score ≥ 0.05 = Positive Sentiment
* Compound Score ≤ 0.05 = Negative Sentiment

## Vader Lexicon Integration with Hate base Lexicon

In this section, the integration of two dictionaries is the novelty of this research project. Lexical approach to sentiment analysis aims to map words to sentiment by building a ‘dictionary of sentiment’. We used this dictionary to score the sentiment of phrases and sentences, without the need of looking at anything else. The hatebase dictionary contains a corpus of hate words in English that is regularly updated on their website. We utilized extraction packages in python with the Hatebase API to extract these words and their allocated scores into a notepad. Then we created another dictionary which we named “VaderHatebase” dictionary. This dictionary was separate from the original Vader dictionary which contained its own text words and scores.

## Research Hypothesis

In our effort to contribute to the detection of hate speech on social media platforms, this research study seeks to construct a new sentiment dictionary, which was the integration of both the Vader lexicons and the hatebase lexicons and compare the difference in detection among the subreddit chosen for analysis. To further evaluate our dictionary integration, we scored comment texts from two other blog sources and performed t-test on them. The goal of this research study is to compare the sentiment scores derived from both analyses of the different dictionaries. We are trying to prove there is an extra level of detection of hate speech when using our integrated dictionary. In the method section, the tables there show a t-test of the two dictionaries for each subreddit. The null null hypotheses (H0) and corresponding alternative hypotheses (H1) is shown below:

H0: There is a difference between the Vader lexicon dictionary score mechanism and Vader+Hatebase dictionary score mechanism the vader+hatebase integration will provide a better score

H1: The Vader + hatebase integration is a lower sentiment score

## Preliminary Hypothesis Testing

When comparing large corpus of data such as comparing the difference of the means from two separate subreddit, a good method to evaluate the significance is by undergoing preliminary testing with a smaller size of the dataset. This is also called hypothesis testing (Dietrich, 2015). Forming an assertion and testing it with data is the basic concept of hypothesis testing. The common notion when performing hypothesis testing is that there is no difference between two samples. This hypothesis is used for constructing either the null hypothesis (H0) or the alternative hypothesis (H1). The two possible outcomes of a hypothesis test are either Null or Alternative. After the test is carried out, we could either reject the null hypothesis and accept the alternative hypothesis (which would mean that there is a difference between two samples) or we could accept the null hypothesis, which means there is no difference between samples. The result from the t-test is known as t-statistic. The t-test was utilized in this research to compare a set of subreddit groups as it asserts that the data sets came from different subreddit with unequal variances and it is used to determine whether the two samples are likely to have come from distributions with equal comments means. Two-sample t-test is used when there are distinct subjects in the two samples. The following equation 1 below is used to determine the statistic value t’

(Equation 1):

If each comment is normally distributed with the same variance and with the same mean (μ1 = μ2), then the t-statistic t, in Equation 2, follows a t-distribution with n1 + n2 – 2 degrees of freedom (df).

(Equation 2):

# The t-test for paired two sample means was performed on a sample of comment data extracted from each of the subreddit analyzed in this research. The first sample was the total sentiment scores for Vader lexicon dictionary analysis and the second sample was the total sentiment score for Vader+Hatebase lexicon dictionary.

# Results

This section shows the result of the data analysis process performed in the methods section. The Table 1 below shows the sum and average of the sentiment score after analysis of comments in the Disney subreddit class. Both scores obtained from the Vader scoring mechanism and Vader-Hate Base scoring mechanism were analyzed. The same process was applied for the following subreddit class which are, Nickelodeon comments, 4chan comments, PBSkids comments. To test the difference in both scoring mechanisms, we employed the statistical sum and average of sentiment scores obtained from each class to further prove that the scoring mechanism developed has a good detection rate and significance. From the Tables 2,3,4, we observed that all values for both the sum and averages are positive except for 4chan subreddit. This indicates that there is potential hate speech in this subreddit class.

|  | Vader Scores | VaderHatebase Scores |
| --- | --- | --- |
| Sum | 32370.91 | 29867.67 |
| Average | 0.32 | 0.24 |

**Disney comments. 4chan comments**

|  | Vader Scores | VaderHatebase Scores |
| --- | --- | --- |
| Sum | -2510.48 | -5430.72 |
| Average | -0.02 | -0.05 |

Table 1: Sum and Average of Sentiment Scores Table 2:Sum and Average of Sentiment Scores

**Nickelodeon comments. PBSkids comments**

|  | Vader Scores | VaderHatebase Scores |
| --- | --- | --- |
| Sum | 7395.39 | 5581.61 |
| Average | 0.07 | 0.06 |

|  | Vader Scores | VaderHatebase Scores |
| --- | --- | --- |
| Sum | 8429.54 | 6553.21 |
| Average | 0.08 | 0.06 |

Table 3: Sum and Average of Sentiment Scores Table 4: Sum and Average of Sentiment Scores

To determine significance between the Vader scoring mechanism and the VaderHatebase scoring mechanism, we performed statistical analysis of paired sample t-test using SPSS. We utilized the 4chan subreddit class scores for the paired test. The Table 5 below shows the Paired sample t-test, which is testing for both the Vader scores and VaderHatebase score for the 100k comments processed through our approach

| **4chan Paired Samples Test** | | | | | | | | | |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  | | Paired Differences | | | | | t | df | Sig. (1-tailed) |
| Mean | Std. Deviation | Std. Error Mean | 95% Confidence Interval of the Difference | |
| Lower | Upper |
| Pair 1 | VaderHatebase - Vader scores | -0.03 | 0.42 | 0.00 | -0.03 | -0.02 | -27.47 | 166770 | 0.000 |

Table 5:Paired Samples t-test results.

In this study, we utilized only the P(T<=t) one-tailed of the test because in the study carried out, we were analyzing the detection rate of our own curated dictionary and we evaluated the significance of this dictionary with comments extracted from four sources of subreddit. Among these four sources, three of them can be classified as non-hateful comments speech, this is because the subreddit communities were mostly discussion held by users with non-hateful intention in their speech. The last subreddit used contains hateful speech which was where we analyzed the results derived from both our dictionary analysis and found significance, the p-value= p<0.05 for some of the sample sets tested.

To further proof the efficiency of our approach, we extracted two additional comment datasets and analyzed the comments using Vader sentiment analyzer and the VaderHatebase sentiment analyzer. The sum and average of both dictionary evaluation is shown below. The datasets extracted were (1) Quotes from Grand Torino movie. The comments contained both hateful speech and regular sentence, but hate speech was existence in this comment dataset. This dataset contained 118 comments used in the movie. (2) The Wikipedia ethnic slurs comment dataset which was the second comments extracted, contains hate speech words. The dataset contained 558 ethnic slur words. The Table 6 and Table 7 below shows the sum and average of the sentiment scores obtained after analysis. This two comments dataset contains hate speech because the sum and average were negative.

|  | Vader Scores | VaderHatebase Score |
| --- | --- | --- |
| Sum | -11.13 | -14.35 |
| Average | -0.095 | -0.12 |

**Wikipedia ethnic slurs comments Grand Torino Comments**

|  | Vader Scores | VaderHatebase scores |
| --- | --- | --- |
| Sum | -3.76 | -6.12 |
| Average | -0.0067 | -0.27 |

Table 6: Sum and Average of sentiment scores Table 7: Sum and Average of sentiment scores

From the results derived from the sum and average of the Vader sentiment score and VaderHatebase score, we performed statistical analysis to show there is a significant difference in both dictionaries. The t-test for Wikipedia slurs and Grand Torino comments is shown in the tables 8 and 9 respectively below

| **Wikipedia Ethnic Slurs Paired test** | | | | | | | | | |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  | | Paired Differences | | | | | t-stat | df | Sig. (1-tailed) |
| Mean | Variance | Pearson Correlation |  | |
| P(T<=t)one-tail | T critical one-tailed |
| Pair 1 | VaderHatebase - Vader scores | -0.265 | 0.117 | 0.064 | 0.00 | 1.64 | 17.89 | 1116 | 0.000 |

Table 8:Paired sample t-test results

| **Grand Torino Paired test** | | | | | | | | | |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  | | Paired Differences | | | | | t-stat | df | Sig. (1-tailed) |
| Mean | Variance | Pearson Correlation |  | |
| P(T<=t)one-tail | T critical one-tailed |
| Pair 1 | VaderHatebase - Vader scores | -0.03 | 0.02 | 0.94 | 0.01 | 1.65 | 2.21 | 116 | 0.01 |

Table 9: Paired sample t-test results

The results derived from the statistical analysis was evaluated based on the P(T<=t) one-tail because its directional. We were proved that our curated dictionary has a better hate detection mechanism. Just like the 4chan dataset the test was used to show that there was a significant difference in both dictionaries.

# Discussion & Conclusion

Different countries have regulations and polices against hate speech in the society. This attention raised the need for automating the detection of hate speech. In this study, we demonstrated the detection of hate speech on Reddit. There has been different hate speech detection mechanism established by researchers. There has not been as much work on sentiment analysis using lexicon-based techniques at the document level. However, recently there has been progress on building lexicons for sentiment analysis. Our main focus was integrating two lexicon dictionaries to improve the detection of hate speech on social media platform. The two libraries that were integrated was the Vader sentiment dictionary and the Hatebase dictionary. We used the Hatebase API to extract the words into a document and integrated it with the words in the Vader dictionary. After the integration, the python code was written to score the comments extracted from different subreddits and two separate comments dataset to prove that our dictionary integration is better at detecting hate speech. The results for “4chan comments” show that our dictionary integration has significant difference in detecting hate speech, while the “Grand Torino” and “Wikipedia slurs” comments was further used to prove our approach. This approach used in this paper can be integrated into social media platforms to help detect hate speech. It could also be integrated into customer review sections on website to detect offensive or hate words from customers when they write a review. It can also be integrated into online workplaces platforms.

# References

Aggarwal, A., Gola, B., & Sankla, T. (2021, 2021//). *Data Mining and Analysis of Reddit User Data.* Paper presented at the Cybernetics, Cognition and Machine Learning Applications, Singapore.

Arulmurugan, R., Sabarmathi, K. R., & Anandakumar, H. (2019). Classification of sentence level sentiment analysis using cloud machine learning techniques. *Cluster Computing, 22*(1), 1199-1209. doi:10.1007/s10586-017-1200-1

Baumgartner, J., Zannettou, S., Keegan, B., Squire, M., & Blackburn, J. (2020). *The pushshift reddit dataset.* Paper presented at the Proceedings of the international AAAI conference on web and social media.

Chen, Z., & Sokolova, M. (2021). Sentiment Analysis of the COVID-related r/Depression Posts. *arXiv preprint arXiv:2108.06215*.

Gaydhani, A., Doma, V., Kendre, S., & Bhagwat, L. (2018). Detecting hate speech and offensive language on twitter using machine learning: An n-gram and tfidf based approach. *arXiv preprint arXiv:1809.08651*.

Gitari, N. D., Zuping, Z., Damien, H., & Long, J. (2015). A lexicon-based approach for hate speech detection. *International Journal of Multimedia and Ubiquitous Engineering, 10*(4), 215-230.

Graumas, L., David, R., & Caselli, T. (2019). *Twitter-based polarised embeddings for abusive language detection.* Paper presented at the 2019 8th International Conference on Affective Computing and Intelligent Interaction Workshops and Demos (ACIIW).

Kumaresan, K., & Vidanage, K. (2019). *Hatesense: Tackling ambiguity in hate speech detection.* Paper presented at the 2019 National Information Technology Conference (NITC).

Machova, K., Mach, M., & Vasilko, M. (2021). Comparison of Machine Learning and Sentiment Analysis in Detection of Suspicious Online Reviewers on Different Type of Data. *Sensors, 22*(1), 155.

Melton, C. A., Olusanya, O. A., Ammar, N., & Shaban-Nejad, A. (2021). Public sentiment analysis and topic modeling regarding COVID-19 vaccines on the Reddit social media platform: A call to action for strengthening vaccine confidence. *Journal of Infection and Public Health, 14*(10), 1505-1512.

Pereira-Kohatsu, J. C., Quijano-Sánchez, L., Liberatore, F., & Camacho-Collados, M. (2019). Detecting and monitoring hate speech in Twitter. *Sensors, 19*(21), 4654.

Pitsilis, G. K., Ramampiaro, H., & Langseth, H. (2018). Detecting offensive language in tweets using deep learning. *arXiv preprint arXiv:1801.04433*.

Ruwandika, N., & Weerasinghe, A. (2018). *Identification of hate speech in social media.* Paper presented at the 2018 18th international conference on advances in ICT for emerging regions (ICTer).

Tan, C., & Lee, L. (2015). *All who wander: On the prevalence and characteristics of multi-community engagement.* Paper presented at the Proceedings of the 24th International Conference on World Wide Web.

Yan, T., & Liu, F. (2021). Sentiment Analysis and Effect of COVID-19 Pandemic using College SubReddit Data. *arXiv preprint arXiv:2112.04351*.

Zampieri, M., Malmasi, S., Nakov, P., Rosenthal, S., Farra, N., & Kumar, R. (2019). Predicting the type and target of offensive posts in social media. *arXiv preprint arXiv:1902.09666*.