|  |
| --- |
| CE807 – Assignment 1 - Interim Practical Text Analytics and Report |
|  |
| **Student id: 220222** |
|  |
|  |
|  |
|  |

Abstract

This report will discuss how text classification techniques are used in the modern world and how these techniques are implemented into a machine learning model in order to detect Hate speech.

Review of Generic Text Classification Methods (Task 1)

This section will discuss the results from the articles that were researched. The first article that will be considered is the Study on SVM (Zhijie Liu et al ,2010). This article discusses Support vector machines with other classification models, including Naive Bayes, K-nearest neighbor and Decision trees, this was conducted on two separate datasets. Taking a look at the results, it is clear that SVM was the superior classification method for this dataset, it achieved a total average accuracy of 86.03%, which was the highest out of all the classification techniques. Naive bayes scored a higher accuracy in certain categories when compared to SVM, but it was not as consistent. It would have an accuracy of 95.93% in one category and 73.49% in another, Whereas the lowest scoring category for accuracy for SVM was 86.03%.

Another article that will be discussed will be (Jafar Ababneh, 2019) where the performance of Naïve Bayes, Decision trees and K-Nearest Neighbor will be used on a dataset of movie reviews. The results show that out of the three algorithms, the best one would be Naïve Bayes, as it had the highest recall, F1 score and precision rate out of all three algorithms. Naive Bayes reached an accuracy of 88.1%. K-Nearest Neighbor was the second-best algorithm for this dataset, it had a accuracy of 85.2%. Decision Trees was the worst performing algorithm out of the three, it had an accuracy of 82.3%.

**(**Tomas pranckevičius and Virginijus Marcinkevičiu, 2017) performed an experiment comparing 5 different classifiers to see which would perform the best on the dataset based on customer reviews from amazon. The classifiers used were Naïve Bayes, Random Forest, SVM, Logistic regression and Decision Trees. The experiment also considered the use of unigrams, bigrams and trigrams, so the results of each individual classifier will be different for the distinguished language models. The results show that the highest average classification accuracy was Logistic regression at 58.5% and its lowest of all the category was the lowest average accuracy recorded for a model was Decision Trees, it had a minimum accuracy of 24.1% and a maximum accuracy of 34.58%, This was indeed the worst performing algorithm for these categories and dataset.

(M.Ikonomakis, S. Kotsiantis, V. Tampakas , 2005) approached text classification through the use of machine learning techniques, these are Naïve Bayes, K nearest and Support vector machine. They performed these techniques on the Reuters Corpus Volume I, which contains a total of 800,000 manually categorized news stories.

(Azam M et al,2018) used extraction-based text classification through the use of K-next neighbor as a classification approach. They performed feature extraction to improve their results. They then compared the performance of the model against the Naive bayes classifier. Without the use of feature extractions, K-NN had an accuracy of 78.67, whereas Naïve Bayes had an accuracy of 71.11. Through the use of feature extraction, K-NN achieved an accuracy of 80.00, and Naïve Bayes achieved a accuracy of 76.95.

(Colas, F. and Brazdil, P. 2006.) The purpose of this paper is to compare SVM against other commonly used classification methods, The results show that naïve bayes and K-NN are seen as faster than SVM for this dataset and that SVM as a model is much harder to train than the other algorithms.

* 1. Critical Discussion (Task 1)

This section will be discussing four different text classification algorithms that were used in the articles researched. Text classification is a commonly used technique that involves automatically assigning a document to a predetermined category or label.

Support vector machine is a binary classification algorithm that can be used for classifying data. This algorithm then places the training data to a neutral space which is then used to classify text. This model is designed to find a boundary/hyperplane between the categories. The hyperplane calculation is done by finding the greatest distance between the categories, these categories tend to be something simple such as a positive/ negative category. The way this model classifies text is the algorithm looks at where the text is likely to fall in accordance with the hyperplane and then calculates the distance of the new data point to the hyperplane and then assigns it to the other side of that boundary.

K-nearest neighbor is a popular used method in machine learning that lets you group similar text documents together based on their characteristics. It is a relatively simple and efficient algorithm as the algorithm stores all instances of the training data in its memory and classifies new instances based on their similarity of the training data. It does this by finding the k-nearest neighbor of a given point and assigns the most common class of that neighbor’s point. The KNN algorithm assumes that similar datapoints will tend to be close to one another. The K-nearest neighbor algorithm is typically used to classify documents into different categories, also known as sentiment analysis, where each category can be labelled as positive, negative or neutral, meaning that this algorithm can be used to detect if articles or reviews can be easily categorized into negative or positive.

Decision Tree is a commonly used algorithm in text classification, it uses a tree like model to identify and then classify text documents, it makes decisions that are typically based on conditions that have been created by a tree, these conditions are typically if statements on each leaf of the tree, the algorithm will consider the given query and then traverse down each branch of the tree until the query is proven true. For example, this algorithm can be used to filter email spam, the model can make classifications of an email based on certain characteristics of each email in the dataset. For instance, one of the questions in the decision tree could be “if the email contains the word free” or if “the email is from a unknown sender” if both these questions prove to be true, the model can then classify that email as spam.

Naive Bayes is a simple but powerful machine learning algorithm that can be used in text classification. The classification process is done by calculating how often a word has appeared in the dataset and then finding out how frequent it appears in each category. The algorithm is then used to calculate the probability of a certain text belonging to a specific category, the algorithm will then pick the category with the highest probability for the text to appear in. it is an excellent prediction model for datasets that tend to be large, it can also be used for sentimental analysis.

Review of Offensive Language Detection Methods (Task 2)

This section will discuss the literature that was researched for this assignment. The first report discussed will be (M Zampieri et al, 2019.), this paper focuses on using text classification to predict if a post on social media is offensive. It uses various different ML models, such as decision trees, SVM and logistic regression to determine if a post is offensive or not.

The next report that will be discussed is created by (S Abro et al, 2020) where a bunch of machine learning algorithms are used for automatic hate speech discussion. A total of 6 algorithms were used, Naïve Bayes, Random Forest, SVM, K-Nearest neighbor, Decision Tree and Logistic regression. The algorithm that performed the best was the SVM algorithm being used along with bigram and TFIDF FE techniques. The dataset being used contained 14509 tweets and a total of 49% of these tweets were seen as either offensive or hate speech.

The next literature was produced by, (Fauzi, M. and Yuniarti, A, 2018) in which they created an ensemble method for hate speech, particularly on Indonesian twitter. They used Naïve Bayes, K-Nearest Neighbors, Maximum Entropy, Random Forest, and Support Vector Machines to use as classifiers. They proposed two methods to help classify text, Soft voting and hard voting. They essentially combined all the individual classifiers to help make a prediction model, each classifier can vote to determine which category the text can belong to, whichever category has the highest number of votes will then be selected as the category to put the text in. This is known as Hard voting. In soft voting the average category probability of each classifier is determined and is used as a score to determine a vote.

The fourth literature by, (Nabiila Adani Setyadi et al, 2018) uses backpropagation to detect hate speech in Indonesian news articles. It uses a backpropagation neural network which uses supervised learning to classify data, the Model is uses a dataset of tweets that contain hate speech to train and test their model, the model then makes adjustments to its configurations to get better at recognizing hate speech, this technique is called backpropagation.

The fifth literature by, (Zimmerman et al, 2018) uses a deep learning ensemble method to detect hate speech. Ensemble methods is a technique that combines multiple different models together to try and improve the accuracy of the mode, it is similar to literature produced by (Fauzi, M. and Yuniarti, A, 2018). It uses both a convolution neural network and a long short-term memory network as its classifiers, it also combines both these networks to get varied classification results. The results show that a combined model has a overall better result than individually using these models.

* 1. Critical Discussion (Task 2)

The reports that are reviewed show that there are several different methods that can be used to detect hate speech. Advanced methods such as Backpropagation and ensemble deep learning methods seem to be the most sensible when it comes to creating a text classifier, this is due to the fact that it will use both the strengths and weaknesses of each individual classifiers to help determine whether a text is considered hate speech. The researcher is also able to see which algorithms perform badly on the dataset selected, since some of these algorithms work better on a variety of datasets.

Text classification models and offensive language detection methods are quite similar in how they work, they both use labels which are assigned to their text in order to organize the data. However, in Text classification the categories defined can be quite varied, from movie reviews to news articles and so on. But, in Hate speech detection, the focus is on identifying and correctly categorizing text that is deemed to be offensive, so there is usually only typically two categories, offensive or neutral, these labels tend to be binary, so a variety of classification techniques can be used. Another difference with offensive language detection method is that they can use both labelled and unlabeled data, Labelled data is typically used to train the ML model to identify any obscenities where as unlabeled data can be used to identify any new trends of offensive languages. This is crucial to hate speech detection as new words or phrases can be created by humans, some of which will end up being offensive/ not suitable for the internet, so it is key for these models to be able to detect any new patterns of offensive language on the internet.

OLID Dataset Characterization (Task 3)

This section will contain an analysis of the dataset that was produced by Zampieri et al. This dataset is then used to identify and categorize any offensive languages used in social media, in particular Twitter. The creators of this dataset made it publicly available to use for research and academia. This dataset was created in 2019 and has been available on the internet since its release.

The dataset contains a large number of tweets, all of which have gone through three processes of labelling, this is in order to detect if the tweet has any offensive or hateful speech to it. The data was collected by gathering any tweets from Twitter by using a filter that detects any offensive languages in a tweet. This dataset could prove potentially useful in my work as it uses text classification to detect hate speech in an online platform, it will further show how text classification models can be used to reduce the amount of slander that is being produced on the internet. The dataset was originally created for a competition in 2019 and has been used and reviewed several times over the past few years, the dataset has been commonly cited in many academic articles, it has a total of 607 citations (google scholar), it can be considered trustworthy, however, there should always be a consideration of bias when it comes to dataset, as there is a chance that a dataset could be skewed to favor a specific outcome.

1. Summary

Hate speech has been a problem that has impacted the Internet since its beginning, especially with how integral social media is to our lives, so the responsibility of minimizing the amount of hate speech is fallen upon text classification models, since they have the ability to process large amounts of data and organize accordingly, they are the model choice for detecting offensive texts on the internet. These models can also be used for simpler tasks, such as dividing and sorting positive or negative movie reviews, it can also be used to filter spam emails and so on.

One advantage of using text classification models for hate speech is that it is quite cost-effective to implement, the way hate speech was detected before was through the use of moderators on social media platforms who would manually go through thousands of text to see if there is anything offensive on a post, this is ineffective as the speed in which a human can read and categorize text will be significantly slower than a machine programmed to do this.

Another advantage would be the fact that these classification techniques seem to be consistent, if a certain type of word is already labelled by the model as negative or offensive the model will typically place that word in the offensive category, in terms of social media or even in the online gaming industry, this is a great feature to implement as it would reduce the amount of offensive language distributed within their platform. Another advantage of using text classification models for detecting hate speech is that these models can genuinely help in creating a safer online space for individuals, this is crucial as adults may not want their children to be on a website that has offensive language on their platform, so it helps that these models can protect vulnerable users from viewing offensive or obscene text.

A challenge in detecting hate speech is the fact that offensive language as a term is quite broad, a given piece of text can be seen as offensive to one individual and not offensive to another human, there is no global definition of what can be regarded as hate speech or what is offensive to people, since this can be very subjective.

Another challenge for hate speech detection is the fact that most of these algorithms typically require a large and varied dataset in order to train the model. It is quite difficult to gather a dataset needed for hate speech detection, these datasets typically come from social media or news articles. The availability of datasets that aren’t in the English language seem to be rare, As training a model in another language can be difficult since other languages tend to have different grammar and structure. Another factor that also must be considered is selection bias, this refers to the fact that the person training the model might end up using a dataset that is skewed, as in the data being selected might be favored towards one category of hate speech, for example, selecting a dataset that primarily has hate speech towards race, the model will then be excellent at detecting any sort of racism online but will struggle to correctly classify hate speech against sexual orientations or religion. This is due to the fact that Hate speech as a term is extensive and can be hard to be labelled as one type of speech.

Some of the Lessons learnt whilst conducting this report is the importance of having the a high quality dataset and choosing appropriate training data, this is due to the fact it would not matter on which algorithm is picked to perform text classification for the model, if the training data you have selected is irrelevant or skewed it will completely alter the results of the model in a negative way. Another lesson learnt is that Hate speech as a term is subjective, so the model you pick must be trained to consider all forms of hate speech whether that be Race, Gender, Religion, sex and so on, failure to do so will result in the model not detecting all forms of hate speech. Another lesson that has been learnt is that text classification for hate speech is quite a new development, it is a field which has a lot of room to grow, as a lot of these models that were analyzed were consistent and accurate, but they were far from the complete product, a lot of these models that were analyzed typically had an average accuracy of 80%-86% which is a relatively high number but is far from perfect.

Overall, text classification methods show the potential to be more reliable/accurate on detecting hate speech, However, as previously discussed, there are still many hurdles for text classifiers to overcome to ensure that these models are as close to perfect when detecting and removing hate speech from online platforms. The classification model that in my opinion was the most common and efficient would be Support Vector Machines.

References

Marcos Zampieri, Shervin Malmasi, Preslav Nakov, 029 Sara Rosenthal, Noura Farra, and Ritesh Kumar. 030 2019. Predicting the type and target of offensive 031 posts in social media. In Proceedings of the 2019 032 Conference of the North American Chapter of the 033 Association for Computational Linguistics: Human 034 Language Technologies, Volume 1 (Long and Short 035 Papers), pages 1415–1420, Minneapolis, Minnesota. 036 Association for Computational Linguistics.

M.IKONOMAKIS, S. KOTSIANTIS, V. TAMPAKAS 2005. *Text classification using Machine Learning Techniques - ResearchGate*. Available at: https://www.researchgate.net/profile/V-Tampakas/publication/228084521\_Text\_Classification\_Using\_Machine\_Learning\_Techniques/links/0c96051ee1dfda0e74000000/Text-Classification-Using-Machine-Learning-Techniques.pdf (Accessed: February 26, 2023).

Azam, M, Tanvir Ahmed, Fahad Sabah, Muhammad Iftikhar Hussain. 2018. *Feature extraction based text classification using K-nearest neighbor ...* Available at: <https://www.researchgate.net/publication/331988179_Feature_Extraction_based_Text_Classification_using_K-Nearest_Neighbor_Algorithm>.

Colas, F. and Brazdil, P. 2006. *Comparison of SVM and some older classification algorithms in text classification tasks*, *SpringerLink*. Springer US. Available at: https://link.springer.com/chapter/10.1007/978-0-387-34747-9\_18

Zhijie Liu, Xueqiang Lv, Kun Liu, Shuicai Shi. 2010. *Study on SVM compared with the other text classification methods* . Available at: https://ieeexplore.ieee.org/document/5459006

Jafar Ababneh. 2019. *Application of naïve Bayes, decision tree, and K-nearest neighbors for ...* . Available at: https://pdfs.semanticscholar.org/9255/8a40d1237e77c65b43e7aaa66a9960fab2a8.pdf

Tomas PRANCKEVIČIUS and Virginijus MARCINKEVIČIU. 2017.*Comparison of naive bayes, Random Forest, decision tree, support vector.*Available at: https://www.researchgate.net/publication/318056374\_Comparison\_of\_Naive\_Bayes\_Random\_Forest\_Decision\_Tree\_Support\_Vector\_Machines\_and\_Logistic\_Regression\_Classifiers\_for\_Text\_Reviews\_Classification

Zampieri, M. *et al.* 2019. *Predicting the type and target of offensive posts in Social Media*, *ACL Anthology*. Available at: https://aclanthology.org/N19-1144/

Abro, S. *et al.* 2020. *Automatic hate speech detection using machine learning: A comparative study*, *International Journal of Advanced Computer Science and Applications (IJACSA)*. The Science and Information (SAI) Organization Limited. Available at: https://thesai.org/Publications/ViewPaper?Volume=11&Issue=8&Code=IJACSA&SerialNo=61

Fauzi, M. and Yuniarti, A. 2018. *[PDF] Ensemble Method for Indonesian twitter hate speech detection: Semantic scholar*, *Indonesian Journal of Electrical Engineering and Computer Science*. Available at: <https://www.semanticscholar.org/paper/Ensemble-Method-for-Indonesian-Twitter-Hate-Speech-Fauzi-Yuniarti/6ead4c60ce0cb335dde8b7d523af34fa81d70fca>

Nabiila Adani Setyadi, Muhammad Nasrun and Casi Setianingsih.2018. *Text analysis for hate speech detection using backpropagation neural ...* (no date). Available at: https://ieeexplore.ieee.org/document/8712109

Zimmerman, S., Kruschwitz, U. and Fox, C. (2018) *Improving hate speech detection with deep learning ensembles*, *ACL Anthology*. Available at: https://aclanthology.org/L18-1404/