William Sharpe

CISA 4313 Fall 2023 Project Final Report

Introduction

The supervised machine learning project that I have chosen is the Cyberbullying Classification problem. It consists of over 46k tweets that belong to 6 different classes. I will implement techniques used through this class to solve this classification problem. The data of the problem is very interesting from a data analysis point of view, and the topic of cyber bullying is one that most everyone can relate to.

The data is constructed into just two columns. The first column, “tweet_text”, is the actual tweet. The second column, “cyberbullying_type”, is the tweet’s bullying category. Both columns have data that is stored as strings. For cyberbullying_type there are 6 possible classes to categorize the tweet as. These include age, ethnicity, gender, religion, other type of cyberbullying, and not cyberbullying. This is what I find most interesting about the problem. Instead of just classifying the tweets as either bullying or not bullying we are looking deeper and finding out what kind of bullying it is. I look forward to seeing what patterns and words will show up for the identification of each class. The data is only 7.17 MB. The size is very manageable and will make doing more CPU intensive things, like many folds of cross validation, realistic.

There are a few reasons that I chose this topic. I find this data set to be very much in line with the lessons we have had and feel it is a good candidate to test what I have learned. Another reason is how extensive twitter is. The sheer amount of data to look over with this kind of study or really any topic of interest one would have is endless. So, learning to work with this data set, I could see myself branching out and exploring other interests, like political sentiment in the upcoming election year, looking into trends in topics of concern over the environment being expressed through twitter, and so on. Finally, cyberbullying has been around as long as the internet has. With our ever-growing submersion and reliance on social media we have and will see more. The outbreak of Covid-19 has further isolated and intensified cyberbullying (Wang, Fu, & Lu 2020). Cyberbullying can have dire consequences from depression to suicide (Wang 2020). Being able to flag and remove the hate is something we can do to combat it.

Related Work

Sentiment analysis using Twitter data: a comparative application of lexicon- and machine-learning-based approach is a study published February 9th of 2023. In this work the authors also explore data sourced from Twitter. Their focus is the sentiment towards Covid-19 in England. They implement methods utilizing lexicon and machine learning, aiming to classify the tweet into one of three categories. Those being as positive, neutral, and negative. 77,332 unique tweets were analyzed. Many of the same methods and models were used. They also took one approach of making all the characters lowercase, removing punctuation as well as other steps like searching for duplicates, and searching for non-English words. Also, many of the same machine learning models were used and similarly we see some of their outliers as far as top and bottom performers being the same. We see this in the Multinomial Naïve Bayes being the worst performer and Random Forest and Support Vector Machine among the top performers.

Data Exploration and Preprocessing/Preparation

The *cyberbullying_tweets* data was stored as a comma-separated values (.csv) document on the Kaggle website. This data was very well put together and easy to use. The first step was to load the data in using the pandas library. The data was checked for null values in its two columns, *tweet_text* and *cyberbullying_type*. Of the 47,692 rows there was not a single null value found in either column. The dataset was also checked for what data types it contained, both columns were objects, also known as string values. Next the target, *cyberbullying_type* was further explored. It contains six possible categories for the tweets to be classified as. The tweets will only be classified as one value, they cannot be labeled as more than one. Thus, this is multiclass and not multilabel classification. In exploring these six categories the number of occurrences of each was investigated. It is important for performance and building a good model that we have a similar number of each category. The number of each was between 7,823 and 7,998. This is very evenly balanced and will benefit or research.

The data was already cleaned and balanced enough for our initial model building. After the first round of testing, we moved into executing some preprocessing and preparation. This was done for comparison to some of the baseline models to see the effect on performance. All the tweets were manipulated to only lowercase characters and all punctuation was removed. Further into the project the target data was manipulated into numbers instead of strings. That is to say that each of the six string values that the bullying could be classified as was assigned a value of 0-5. This encoding was done for building a Neural Network model.

Modeling

There were a few different approaches taken in building models. All approaches started with the CountVectorizer. This bag of words technique breaks the string of words into tokens that can be used by our models. After vectorizing, the data was split into training and testing. Each time it was done the same, 70% for training and 30% for testing. The integer 132 was used consistently for random state in all splits. All the stop words in the *english* list were removed.

For the first wave of testing six machine learning models were used. This first batch of testing saw the data as is with no data manipulation. The tests and their corresponding accuracy, rounded to 4 decimal places, were as follows:

Logistic Regression: 0.8364

Random Forest: 0.8265

Support Vector Machine: 0.8254

Decision Tree: 0.8079

Multinomial Naïve Bayes: 0.7807

K-nearest Neighbor: 0.6818

For the next set of tests, we look at the top performer, Logistic Regression, a middle performer, decision tree, and the worst performer, K-nearest Neighbor. We will feed these 3 representatives the exact same data as done previously but we will make all data characters lower case and remove any punctuation. One would assume this would help to remove ambiguity and improve performance.

However, the results showed that the models stayed the same or even performed slightly worse. Results again rounded to four decimals compared to their original values:

Logistic Regression: 0.8299 as compared to previous of: 0.8364

Decision Tree: 0.8014 as compared to previous of: 0.8079

K-Nearest Neighbor: 0.6665 as compared to previous of: 0.6818

For the next approach I am interested in trying to implement n-grams with CountVector. The advantage here is that instead of breaking everything into single word token we can have 1 to many word tokens. Specifically, I am using one approach of only bigrams and one approach consisting of unigrams, bigrams, and trigrams. This might allow more meaning to be retained. The issue I ran into is that the kernel keeps dying because of the sheer amount of data. When looking at the unigram, bigram, and trigram, which will be referred to as n-grams moving forward, there are 884,215 features. I tried increasing the max buffer size in Jupyter notebooks config file as well as using Google Collab Pro and other IDEs to no avail. To resolve my issue and continue with testing I took a random sample of 10,000 from the whole data set. This random sample was checked for even representation of each class and was given a random state of 132. After being vectorized n-grams consisted of 217,279 features and bigrams consisted of 93,392 features. The same 3 from the previous section, K-Nearest Neighbor, Decision Tree and Logistic Regression were chosen for their representation of the good, bad and okay. The results were the following:

Bigram:

Logistic Regression: 0.5907

Decision Tree: 0.6100

K-Nearest Neighbors: 0.2717

N-gram:

Logistic Regression: 0.8223

Decision Tree: 0.7990

K-Nearest Neighbors: 0.3987

For the last set of models, a Neural Network was used. As the data that was altered to all lower case and no punctuation performed worse, the original unaltered data was used for the Neural Network. For this model not only was the data broken into training and testing but a small amount, 2,500 entries, of the training were set aside as validation. This allows for us to use just the training and validation to train our model and then we can evaluate the model with the testing data, which our model has never seen before. This will give us the truest measurement of accuracy.

In the first iteration of training the model 100 epochs were used. It was clear to see that in doing this many epochs we had overtrained the model. This can be seen in the overfitting that is shown by having a 0.9758 training accuracy and a testing accuracy of 0.8228. While not terrible, especially when we look back at previous models, the model could be improved. If we look over the history of fitting the model, we can see that there is a sweet spot of balance between the training accuracy and validation accuracy around 16 epochs. A new model was trained for only 16 epochs and yielded more balanced results. They were the following:

Training Accuracy: 0.9209

Validation Accuracy: 0.8402

Testing Accuracy: 0.8432

This is much better not only in accuracy but also addresses the overtraining and over fitting issues.

Since Logistic Regression was the top-performer, besides Neural Networks, it was checked for overfitting and underfitting using cross validation. Again, there is an issue with the kernel crashing while executing the code. The same idea of taking a random sample of 10,000, with even distribution and random state of 132, from the whole set was employed. Cross validation was set to 10 folds and the accuracy results were the following:

0.809; 0.837; 0.840; 0.832; 0.815; 0.833; 0.823; 0.825; 0.827; 0.833;

Mean of scores: 0.83

	Precision	Recall	F1-Score	Support
Age	0.98	0.98	0.98	1,668
Ethnicity	0.98	0.97	0.98	1,713
Gender	0.91	0.83	0.87	1,634
Not Cyberbullying	0.57	0.56	0.56	1,639
Other	0.59	0.68	0.63	1,642
Religion	0.97	0.94	0.95	1,704
Accuracy			0.83	10,000

Results and Discussions

The Neural Network with 16 epochs and the Logistic Regression were the best performers. They were 0.8432 and 0.8364 respectively. Both were checked for over and underfitting issues. Each is a good performer for the scope of this project. There is always room for tweaking and improvement. Looking at the table of the cross validation results we can see how well the model is performing at the task of identifying 4 of the 6 classes. Those 4 are the more distinguishable, being more precise as to what they are, age, ethnicity, gender, and religion bullying. The other 2 class that my model could only identify at a much lower rate happen to be much broader categories. They are more ambiguous. If we dropped those two classes our metrics would be in the mid to high 90s. While dropping both classes is not entirely helpful for the usefulness of this study, I would say keeping not cyberbullying class as is and then looking more into the other type of bullying class for opportunities to break it refactor it. If we could break the other class into more specific classes, like the other 4 classes, we could improve there and just take the hit in not cyberbullying class. Other classes could be education, sexual, physical appearance, nationality, and politics to name a few.

Beyond these improvements more information on the data would be beneficial. Knowing where these tweets were sourced, the exact time frame and how they were chosen.

References and Citations

J. Wang, K. Fu, C.T. Lu, "SOSNet: A Graph Convolutional Network Approach to Fine-Grained Cyberbullying Detection," Proceedings of the 2020 IEEE International Conference on Big Data (IEEE BigData 2020), December 10-13, 2020.

Qi, Y., & Shabrina, Z. (2023). Sentiment analysis using Twitter data: a comparative application of lexicon- and machine-learning-based approach. *Social network analysis and mining*, *13*(1), 31. https://doi.org/10.1007/s13278-023-01030-x

Link to data

https://www.kaggle.com/datasets/andrewmvd/cyberbullying-classification/data

Link to work by Qi and Shabrina

https://www.ncbi.nlm.nih.gov/pmc/articles/PMC9910766/