

NATURAL LANGUAGE PROCESSING

J COMPONENT

Parental Discretion Model

Slot: E1+TE1

Team Members:

Shivam Yadav(18BCE0908)

Pavitra Kadiyala(18BCE0926)

Lavanya Middha(18BCE0845)

GITHUB LINKS:

1. FRIDAY TASKS:

<https://github.com/ShivamYadav09/18BCE0908-NLP>

<https://github.com/pavitrakadiyala/NLP-TASKS>

https://github.com/LavanyaMiddha/Natural-Language_processing

2. J-COMPONENT:

Classifiers:

<https://github.com/ShivamYadav09/ParentalDiscretionModel>

Logistic Regression:

<https://github.com/pavitrakadiyala/Parental-Discretion-using-Logistic-Regression>

Neural Network:

https://eu-gb.dataplatform.cloud.ibm.com/analytics/notebooks/v2/6e69c75d-9f79-44c5-8bf5-9b0a8c50a2f8/view?access_token=db88b3e1753ecad02a332eed9579f8a013de363802d7c46980240f27246be0cf

3. NATGEO(SAME TEAM):

[https://github.com/pavitrakadiyala/NLP-TASKS/blob/master/nlp-team4-pipeline%20\(2\).ipynb](https://github.com/pavitrakadiyala/NLP-TASKS/blob/master/nlp-team4-pipeline%20(2).ipynb)

OBJECTIVE:

Our Model's objective is to be able to give us the probability which tells us the given sentence is advisable for children (below the age of 8) or if Parental Discretion is needed. We used data mining and added some sentences from our own volition to get 1000 sentences which are decent as well as are violent. The sentences which are advisable for the children as labelled as 0 and which are not are labelled as 1. Then we trained the model on these.

ALGORITHMS:

Methodology-1 (Classifiers- Shivam):

1) Dataset train and Preprocessing:

- removed HTTP tags
- lowered the case
- removed all punctuation and Unicode
- removed stopwords
- lemmatization(converting a word into its root form considering the relevant Part of Speech associated with the word)

2) Got the top 10 and bottom 10 words and plotted them to get a sense of which words have greater impact on the model.

3) Embedding:

- Bow(frequency)
- TF-IDF(weight)

4) Split the data for test and train (15:85)

5) Built 4 classification Models namely:

- Random Forest Classifier,
- Adaboost Classifier,
- Gradient Boosting Classifier,
- Naive Bayes Classifier

6) Evaluated performance of each classifier based on:

- Confusion Matrix
- Accuracy Score
- Precision Score

- Recall Score
- Roc-Auc Score

Result And Error: Comparing the models, Random Forest Classifier performs better than the other 3 with an overall accuracy of 74%. Error was that I initially used the wrong formula for auc value which was giving really poor results despite other evaluation criterias having similar values. Then, I realized the problem and fixed it.

Methodology-2(Logistic Regression-Pavitra):

1) Dataset train and Preprocessing:

- removed html tags
- lowered the case
- remove url and numbers
- Tokenize
- Remove punctuations
- removed stopwords
- lemmatization

2) TF-IDF

3) BOW

4) Split the data for test and train

5) Built the Logistic Regression Model

6) Evaluated performance based on:

- Confusion Matrix
- Accuracy Score
- Precision Score
- Recall Score

Result and Errors: The accuracy of the model was 78% for TF-IDF and 77% for Bag of Words which is more than the Random Forest Classifier. I had a bit of trouble in preprocessing the data of a csv file as i had never done preprocessing of a csv file.

Methodology-3(Neural Network-Lavanya):

1) Dataset train and Preprocessing:

- lowered the case
- remove url and numbers
- tokenize
- removed stopwords
- lemmatization

2) Split the data for test and train

3) Generating tokens for testing and training sentences

4) Converting String tokens to integer sequences

5) Padding the generated sequences with 0s to make all the sequences of the same length

7) Training based on Neural networks

8) Evaluated performance based on:

- Training and Testing Accuracy
- Training and Testing Loss

Result: The accuracy of the model was 80% which is more than the both, Classifier and Logistic Regression Model.

OBSERVATION: We observe that the Neural Network model had the most accuracy in the three models on the dataset we used.

CHALLENGES FACED:

The primary challenge has to be the dataset formation. We had planned on making a project for “Threat Detection”, which would tell if the sentence is a threat (rape, murder, slander) or not. However we quickly found out that the dataset was becoming very monotonous with most of them being slangs and we had to add some really NSFW sentences by myself. So we changed the topic to something which could still make use of the dataset we had made with a much broader field for sentence additions. Thus we went for “Parental Discretion Model”.

Still adding 1000 sentences and manually labelling them is not a small task. We used to keep 1 to 1 and half hours of our night time for adding the sentences. Previously we had no idea of making a dataset from scratch that is why when we were told we can either make one or go for a pre-made one, we were determined to make one as it would broaden our skill set and making datasets is an integral part of being a Machine Learning enthusiast.

We faced quite a bit of problem on making the dataset properly so that it would be imported into the model using pandas. We found out that we don't need to explicitly give serial numbers and thus had to drop that column or make a copy of the dataset without that column.

FUTURE SCOPE OF THE MODEL :

- ❖ We have trained our model on 1000 sentences and the model knows based on the accuracy which is "not suitable" and which is "suitable". Now we can make another really simple model, feed it the sentences that need to be labelled and use the aforementioned pretrained model to classify the new sentences.
- ❖ We have bunched up all the different reasons for which a web page/ article could be deemed as unsuitable for child's viewership. We could divide them in categories (rape/murder/self-derogatory/lowering self-esteem and self-worth/potraying a bleak picture of future) and tell why the page or article is not suitable for viewership.

REFERENCES:

<https://towardsdatascience.com/understanding-auc-roc-curve-68b2303cc9c5>

<https://medium.com/swlh/using-a-pre-trained-toxicity-classifier-to-classify-sentences-354c36417f69>

<https://medium.com/analytics-vidhya/accuracy-precision-and-recall-in-machine-learning-classification-ae84004e86a1>

<https://towardsdatascience.com/nlp-for-beginners-cleaning-preprocessing-text-data-ae8e306bef0f>

<https://www.reddit.com/r/Rarethreats/>

<https://www.foxnews.com/category/us/crime/homicide>