Toxic Comment Classification

# Contents:-

1)Introduction:-

1.1 Business problem

1.2 Understanding metrics

1.2.1 AUC ROC

1.2.2 Confusion matrix

1.3 Data

1.4 Sampling technique

1.4.1 Random sampling

1.5 Mongo DB

2) Methodology:-

2.1 EDA and Statistical analysis

2.1.1 Missing values

2.1.2 EDA

2.1.2.1 Distributions of dependent variables

2.1.2.2 Bivariate analysis

2.1.2.3 Distributions of tags

2.2 Feature engineering

2.2.1 Feature variables

2.2.2 Validation of features

2.2.2.1 Violin Plots for clean text and toxic variables

2.2.2.2 Correlation plot

2.2.2.3 Logistic regression feature importance

2.3 Text preprocessing

2.3.1 Word cloud after cleaning the text

2.4 Vectorization

2.4.1 Top TFIDF unigrams and Bigrams Visualization

2.5 Modeling

2.5.1 Logistic regression

2.5.2 Naïve Bayes regression

2.6 Important features from the models

3) Results and conclusions

3.1 Results

3.1.1 Logistic Regression (auc-roc ,confusion matrix)

3.2.1 Multinomial Naïve bayes (auc-roc ,confusion matrix)

3.2 Conclusion

4) Annexure code

Pdf with the save name.

1 INTRODUCTION

**1.1 Business Problem:-** Build a multi-headed model that’s capable of detecting different types of of toxicity like threats, obscenity, insults, and identity-based hate better than Perspective’s current models. You’ll be using a dataset of comments from Wikipedia’s talk page edits. Improvements to the current model will hopefully help online discussion become more productive and respectful.

**1.2 Understanding the metrics :-**

1.2.1 Area under the curve(ROC curve) :- The ROC curve is created by plotting the [true positive rate](https://en.wikipedia.org/wiki/True_positive_rate) (TPR) against the [false positive rate](https://en.wikipedia.org/wiki/False_positive_rate) (FPR) at various threshold settings. The true-positive rate is also known as [sensitivity](https://en.wikipedia.org/wiki/Sensitivity_(tests)), [recall](https://en.wikipedia.org/wiki/Precision_and_recall#Definition_(classification_context)) or probability of detection in [machine learning](https://en.wikipedia.org/wiki/Machine_learning). The false-positive rate is also known as the [fall-out](https://en.wikipedia.org/wiki/Information_retrieval#Fall-out) or probability of false alarm.

1.2.2 Confusion matrix :- A confusion matrix is a table that is often used to describe the performance of a classification model and has two dimensions actual and predicted.

**1.3 Data**

Text column :- a large number of Wikipedia comments which have been labeled by human raters for toxic behavior. The types of toxicity are:

* toxic
* severe\_toxic
* obscene
* threat
* insult
* identity\_hate

Aim :- create a model which predicts a probability of each type of toxicity for each comment

**1.4 Sampling technique**

1.4.1 Random sampling:- A simple random sample is a subset of a statistical population in which each member of the subset has an equal probability of being chosen, around 30000 points where chosen by random sampling

**1.5 :- Mongo DB**

After random sampling data is stored in a database called data and extracted whenever needed

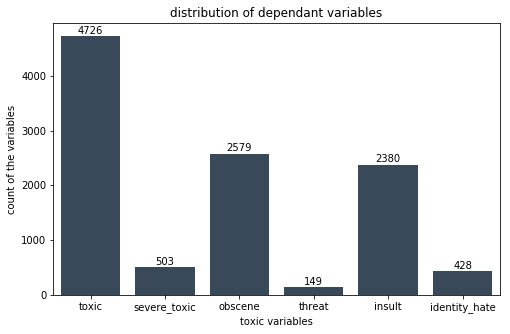
# 2 METHODOLOGY

**2.1 EDA and Statistical analysis**

2.1.1 Missing values :- There are no missing values in the data set

2.1.2 Exploratory data analysis:-

2.1.2.1 Exploring the Distributions of the dependent variables



From the above distribution we can see that toxic variable is the highest followed by obscene then by insult and the least is the threat.

From this we can understand that people are identifying comments as toxic, obscene and insult more than the rest further we find class imbalance.

2.1.2.2 Bivariate analysis:- To find the relation between the toxic and other dependent variables I have used cross tabulation.

Cross Tabulation:-

\_toxic severe\_toxic obscene threat insult

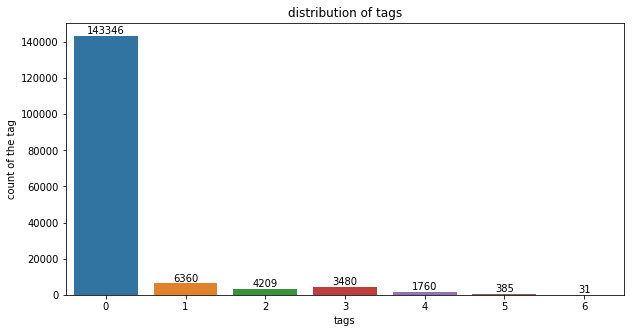
severe\_toxic 0 1 0 1 0 1 0 1 0 1

toxic

|  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **0** | 144277 | 0 | 143754 | 523 | 144248 | 29 | 143744 | 533 | 144174 | 103 |
| **1** | 13699 | 1595 | 7368 | 7926 | 14845 | 449 | 7950 | 7344 | 13992 | 1302 |

from the above we can see that severe\_toxic is total toxic and toxic column has influence on other columns like obscene and threat

2.1.2.3. Distributions of tags:-



Tag with 1 is more followed by tag 2 and 3 least is 6 which is sum of all the tags Here 0 means no tag.

**2.2 Feature Engineering**

Feature engineering is the process of using domain knowledge of the data to create features that make machine learning algorithms work.

2.2.1 Feature variables:-

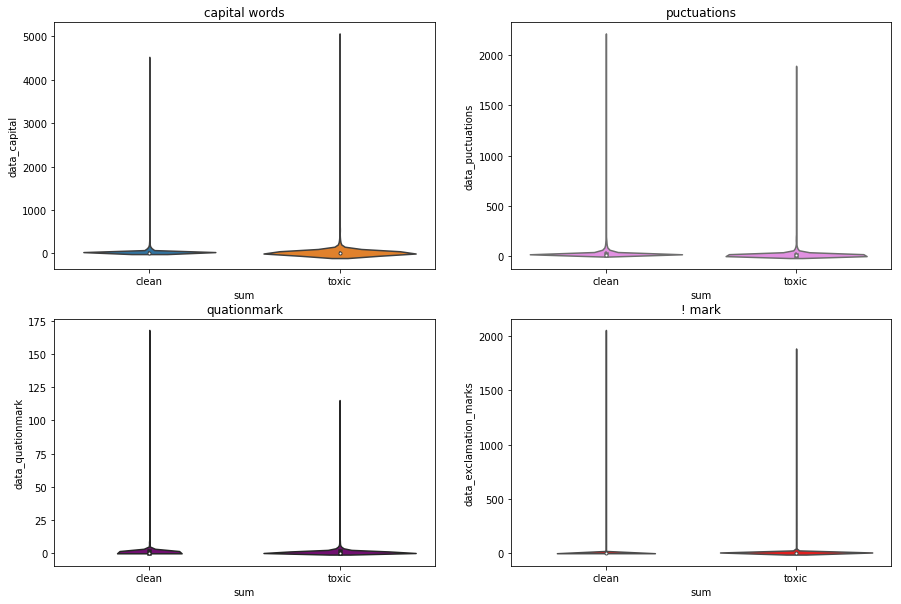
After through investigation of the text columns and the toxic columns I felt that the following variables would be useful for the model.

* Number of Capital Words.
* Number of punctuation,
* Number of question marks , !
* length of the sentence
* length of the words
* number of symbols
* number of unique words
* proportion of unique words
* Polarity :- By using textblog sentiment analysis I created a variable called polarity which takes in 1-10 Polarity and added the text to the column variable.

2.2.2 Validation of features:-

For validation I have used Violin plots and correlation matrix further decided the variables by Logistic regression feature importance.

2.2.2.1 Violin Plots for clean text and toxic variables text

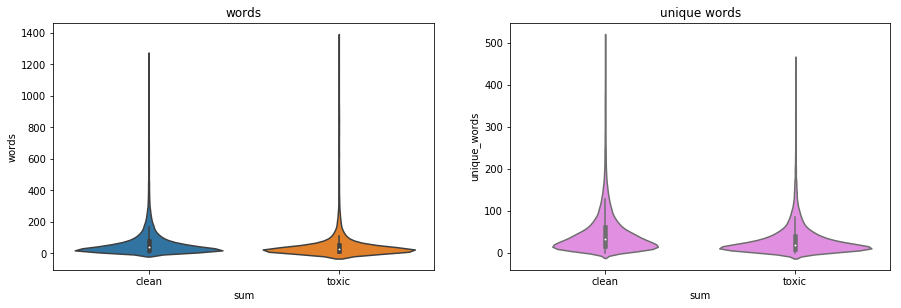


Findings from the above

we can clearly see that there are lot more capital words in toxic and qutionmarks too.

! mark may not be usefull due to the distributions and less punctuation in toxi

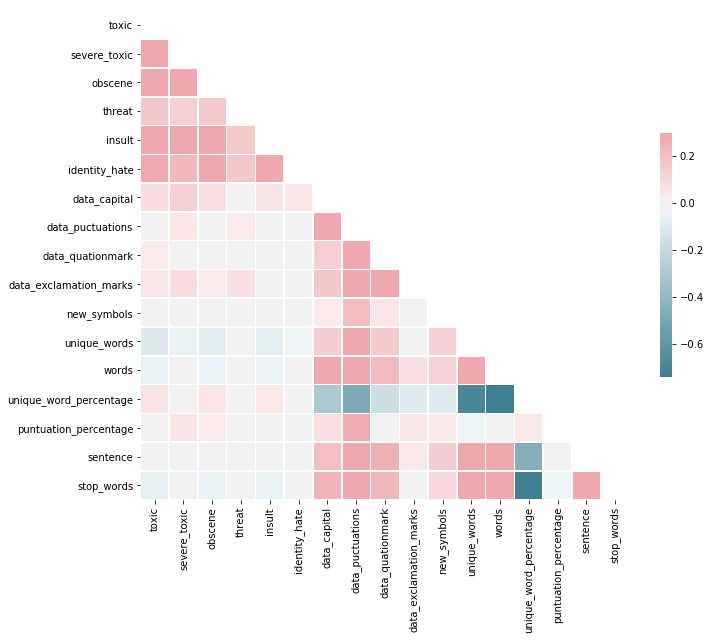
Words vs unique words



Words variables may not be useful as the spread is almost the same further unique words may have effect.

2.2.2.2 Correlation plot

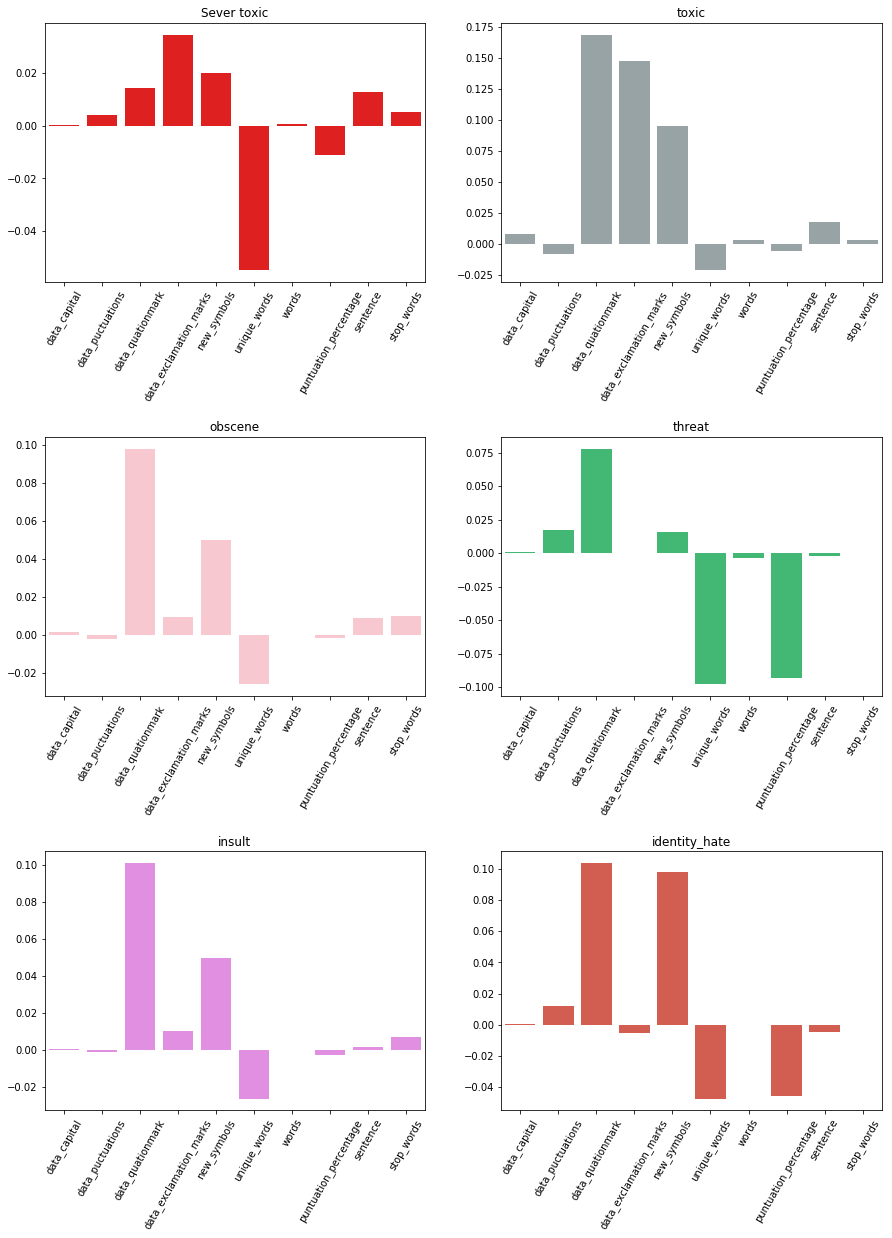
To validate the features further correlation was used.



From the above plot we can see that capital words, number stop words,puntuation %,exclamation marks are helpful on dependent variables.

2.2.2.3 Logistic regression feature importance

To get more better understand the feature engineering variables I have used logistic regression to understand the importance. Feature Importance outputs as follows



With the help the validation methods for feature engineering I have chosen the following

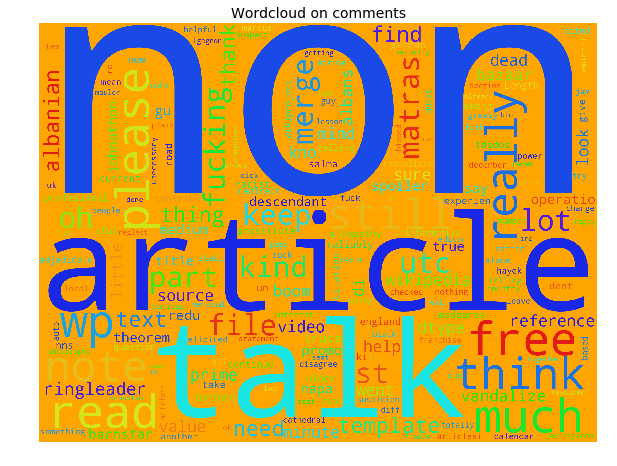
* Number of Capital words, punctuation
* Number of question mark , exclamation marks
* Number of new symbols
* Unique word percentage
* Number of sentence,
* Number of Stop words and polarity – added to text

**2.3 Text preprocessing**

For the text preprocessing the following methods where used

* Punctuation Marks
* Numbers
* Case folding
* Stop words
* White spaces
* Lemmatization

2.3.1 Word cloud after cleaning the text

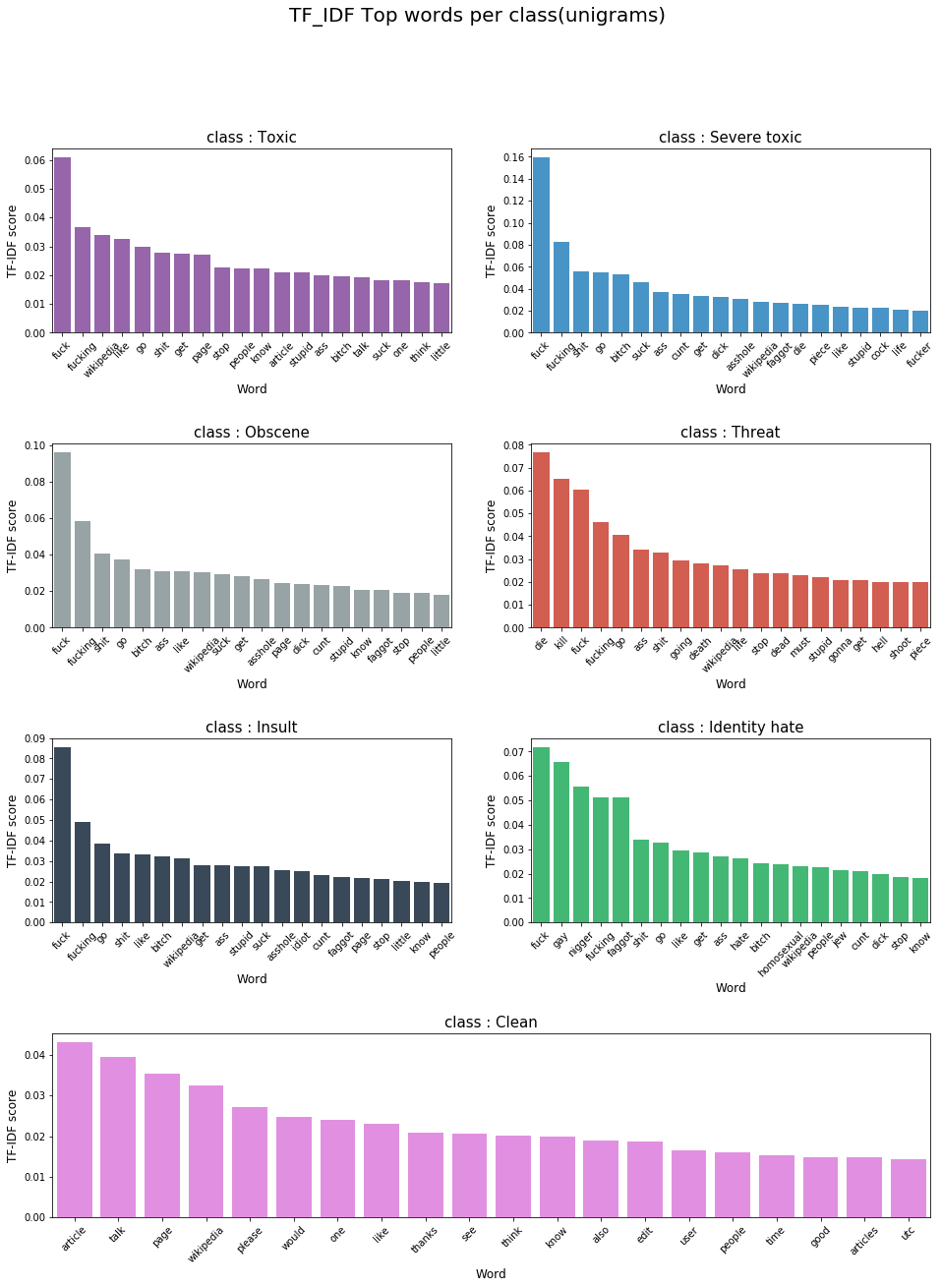


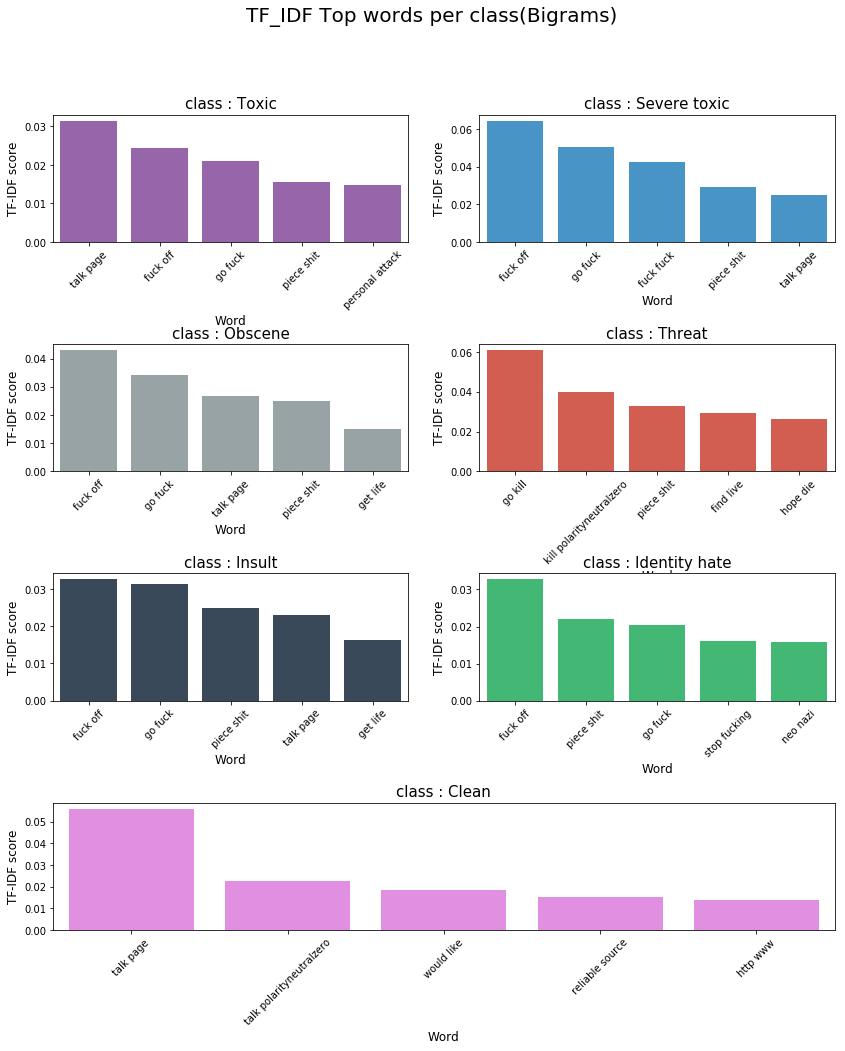
**2.4 Vectorization :-** This is the process of converting the text in to numbers

TFIDF :- A typical weighting is tf-idf weighting:

w = tf \* idf = tf log2 (N / df)

2.4.1 Top TFIDF unigrams and Bigrams Visualization





**2.5 Modeling**

Training and testing sets :- 80 % and 20 %

This is a classification problem so we can use all the classification related models but as we are dealing with text, where the dimensions are large I have chosen Logistic regression and Naïve bayes because the time complexity and run time complexity is low for the above models.

2.5.1 Logistic regression :- Logistic regression is a statistical method for analyzing a dataset in which there are one or more independent variables that determine an outcome.

Hyper turning parameter is alpha which is C = 1/alpha

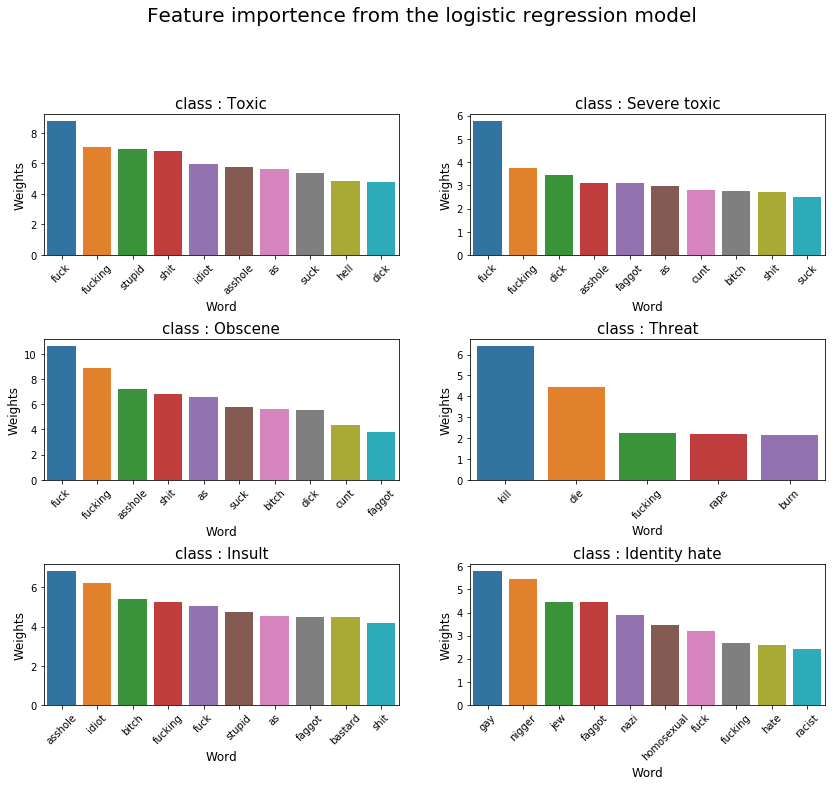
This Controls the over fitting and under fitting. I have chosen C = 1.6 which gave the optimal result for this dataset.

2.5.2 Naïve Bayes regression:- naive Bayes classifiers are a family of simple "[probabilistic classifiers](https://en.wikipedia.org/wiki/Probabilistic_classifier)" based on applying [Bayes' theorem](https://en.wikipedia.org/wiki/Bayes%27_theorem) with strong (naive) [independence](https://en.wikipedia.org/wiki/Statistical_independence) assumptions between the features.

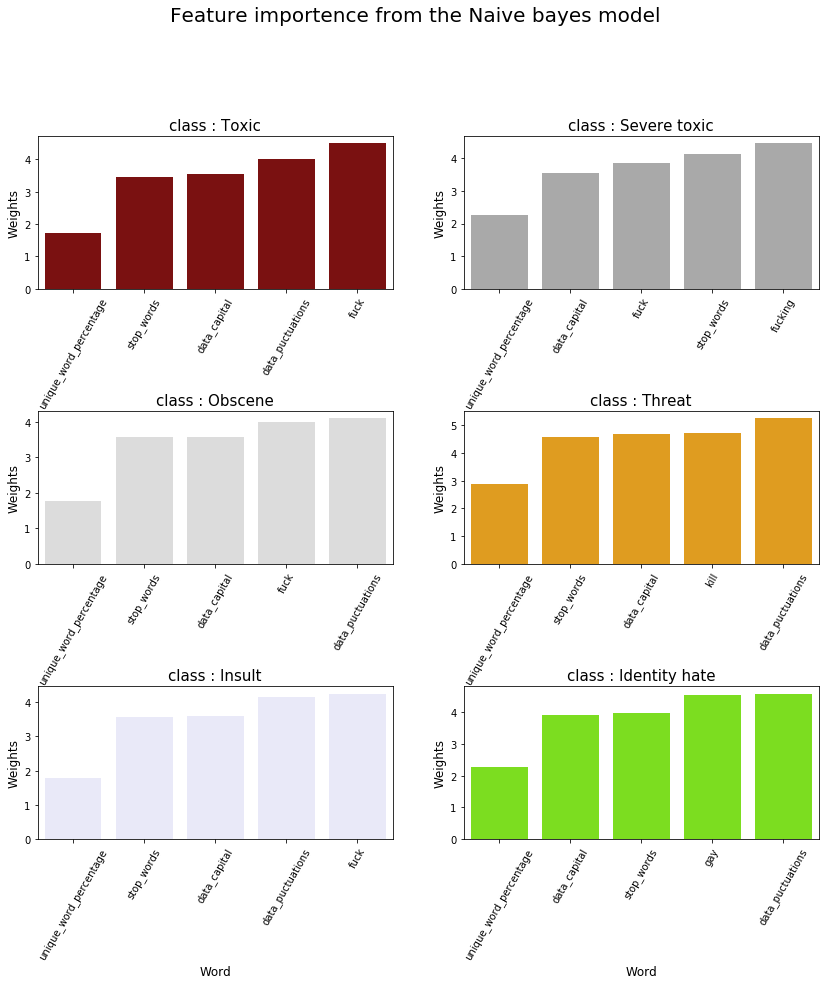
Hyper turning parameter here is alpha, alpha = 0.2 gave me the reasonable result.

**2.6 Feature importance :-**

Interpretability of the model is very important so for this reason, a visualization of features that are important for classification was done and shown below



Naïve bayes



3 Results and Conclusions

**3.1 Results:-**

3.1.1 Logistic regression:-

AUC-ROC :-

Mean score of auc roc curve by Cross validation of 3 on training dataset is 0.965

Score of auc roc on text dataset is 0.96

Confusion matrix:-

Toxic

array([[5421, 27],

[ 235, 317]], dtype=int64)

Sever toxic

array([[5927, 9],

[ 51, 13]], dtype=int64)

Obscene

array([[5661, 11],

[ 137, 191]], dtype=int64)

Threat

array([[5978, 0],

[ 22, 0]], dtype=int64)

Insult

array([[5681, 26],

[ 144, 149]], dtype=int64)

Identity hate

array([[5942, 4],

[ 49, 5]], dtype=int64)

3.1.2 Multinomial NB

AUC-ROC :-

Mean score of auc roc curve by Cross validation of 3 on training dataset is 0.948

Score of auc roc on text dataset is 0.943

Confusion matrix:-

Toxic

array([[5408, 40],

[ 243, 309]], dtype=int64)

Sever toxic

array([[5925, 11],

[ 48, 16]], dtype=int64)

Obscene

array([[5650, 22],

[ 137, 191]], dtype=int64)

Threat

array([[5978, 0],

[ 22, 0]], dtype=int64)

Insult

array([[5668, 39],

[ 136, 157]], dtype=int64)

Identity hate

array([[5941, 5],

[ 50, 4]], dtype=int64)

**3.2 Conclusion** :- From the results we can clearly see that logistic regression has performed well then Multinomial Naïve Bayes model.