Toxic Comment Classification

Parth Patel — Vishal Kumar — Amish Ranjan

March 2018

1 Introduction

Nowadays almost all websites for social purposes like news, e-commerce, entertainment, knowledge etc have a comment section in some form or other. Knowledge-based websites have discussion forum while e-commerce websites have rating and review section, in some way all of them are providing a space for social interaction on a public platform. It is certainly available for good purposes but the bad can't be removed, we could just rectify it(the sole purpose of this project). People sometimes use toxic language on these platforms that make others uncomfortable and sometimes the toxicity touches extreme, the comments become racist, communal, life- threatening etc. Every now and then the website administrators have to close the comment section just to avoid these things. Recently Quint(a media body) has to close its rating and review section due to extremely toxic comments from a particular section of society. Just to avoid these scenes many scholars have tried to rectify these toxicities in an optimized way to reduce time and calculation. In this article, we would try to have a critical eye on the approaches of these scholars. Later, we would try to pose our own model to solve this issue.

2 Contribution

We all have invested equal amount of time together. So, we have contributed equally in all parts.

3 Baseline Models

As we have already mentioned in our previous reports that we would be using two models as our baseline model:-

- Logistic Regression
- Multi-layer Perceptron
- Deep Neural Network
- RNN with LSTM

4 Why we are using these models??

Logistic regression is probably responsible for the majority of industrial classifiers, with the possible exception of naïve Bayes classifiers. Logistic regression is also known as maximum entropy, neural network classification with a **single neuron**, and others. Logistic regression uses unrestricted feature extraction, which allows for arbitrary observations of the situation to be encoded in the classifier. In our situation, we may sometimes have non-standard sentences, which could be better captured using arbitrary observation of Logistic Regression. As aforementioned Logistic Regression can be viewed as single neuron neural network, it is quite obvious that we should try deep neural network with multiple nodes to get better performance. That is why we are trying our second model **RNN with LSTM**. We are also trying MLP and DNN, just to see the output.

5 Model Description

- **Logistic Regression:-** Logistic Regression is a classification algorithm. It is used to predict a binary outcome (1 / 0, Yes / No, True / False) given a set of independent variables. To represent binary / categorical outcome, we use dummy variables. You can also think of logistic regression as a special case of linear regression when the outcome variable is categorical, where we are using log of odds as dependent variable. In simple words, it predicts the probability of occurrence of an event by fitting data to a logit function. It is easy to minimize loss function:-

  $$\mathrm{l}(\theta) = \sum_{i=1}^{i=n} \mathrm{y}^i \log(\mathrm{h}(x^i))\text{-}(1\text{-}y^i)(1\text{-}\mathrm{h}(x^i)\ )$$

  If there is a little deviation from actual function, there is exponential higher loss for that.

- **Multi Layer Perceptron:-** A multilayer perceptron (MLP) is a class of feedforward artificial neural network. An MLP consists of at least three layers of nodes. Except for the input nodes, each node is a neuron that uses a nonlinear activation function. MLP utilizes a supervised learning technique called backpropagation for training.

- **Deep Neural Network:-**In deep-learning networks, each layer of nodes trains on a distinct set of features based on the previous layer's output. The further you advance into the neural net, the more complex the features your nodes can recognize, since they aggregate and recombine features from the previous layer. Deep-learning networks perform automatic feature extraction without human intervention, unlike most traditional machine-learning algorithms

- **RNN with LSTM:-**Unlike the conventional translation models, where only a finite window of previous words would be considered for conditioning the language model, Recurrent Neural Networks (RNN) are capable of conditioning the model on all previous words in the corpus. But RNN has Vanishing Gradient Gradient Explosion Problems, which are solved by LSTM. That is why we are using RNN with LSTM.

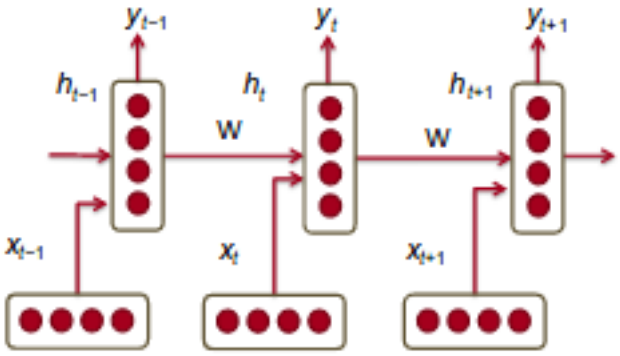

Figure 1: A recurrent neurall network with three time steps shown

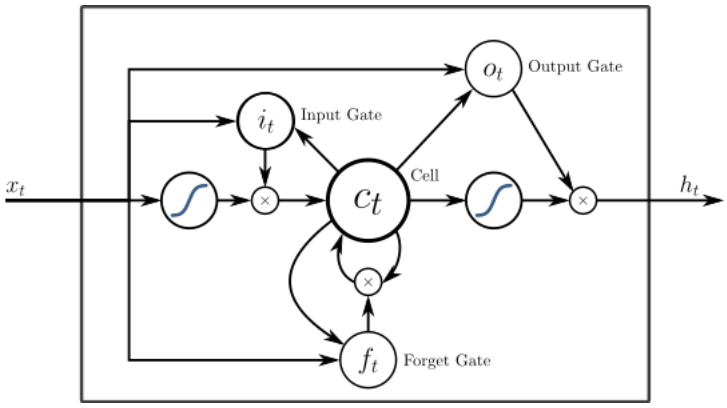

Figure 2: An LSTM with all gates

LSTM model which introduces a new structure called a memory cell (see Figure 1 below). A memory cell is composed of four main elements: an input gate, a neuron with a self-recurrent connection (a connection to itself), a forget gate and an output gate. The self-recurrent connection has a weight of 1.0 and ensures that, barring any outside interference, the state of a memory cell can remain constant from one timestep to another. The gates serve to modulate the

interactions between the memory cell itself and its environment. The input gate can allow incoming signal to alter the state of the memory cell or block it. On the other hand, the output gate can allow the state of the memory cell to have an effect on other neurons or prevent it. Finally, the forget gate can modulate the memory cell's self-recurrent connection, allowing the cell to remember or forget its previous state, as needed.

6 Data Collection

We have referred to Kaggle Competition, for this project. There is a well defined and structured dataset available. The dataset consists of Wikipedia comment made available by Google-Jigsaw in public domain. The training set have:-

id	comment_text	toxic	\
0000997932d777bf	Explanation\nWhy the edits made under my usern...	0	
000103f0d9cfb60f	D'aww! He matches this background colour I'm s...	0	
000113f07ec002fd	Hey man, I'm really not trying to edit war. It...	0	
0001b41b1c6bb37e	"\nMore\nI can't make any real suggestions on ...	0	
0001d958c54c6e35	You, sir, are my hero. Any chance you remember...	0	

severe_toxic	obscene	threat	insult	identity_hate
0	0	0	0	0
0	0	0	0	0
0	0	0	0	0
0	0	0	0	0
0	0	0	0	0

Figure 3: An example of test set

- id
- comment line
- toxic(0,1)
- severe_toxic(0,1)
- obscene(0,1)
- threat(0,1)
- insult(0,1)
- identity_hate(0,1)

7 Data Pre-processing

The data is given in English language, we have parsed it word by word to calculate tf-idf(with n-gram). We have also pre-processed the non-standard english like (a$$hole, l o s e r) to make it useful for our training purpose, otherwise it was being discarded.

8 Result

We are using 2-3rd of the data as training and the rest for test.

- Logistic Regression :-

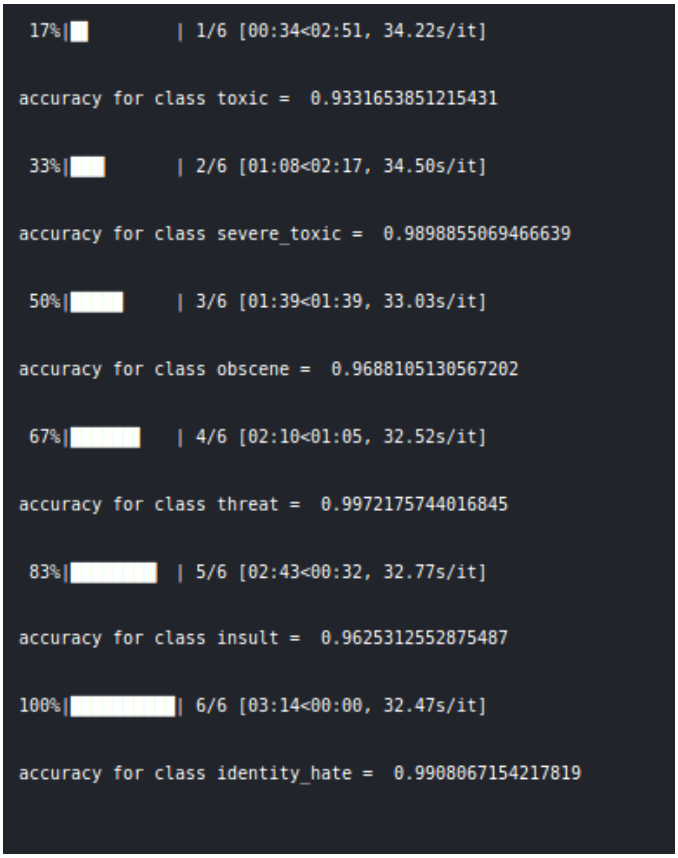

Figure 4: Results for logistic regression

- Multi Layer Perceptron :- We have used five hidden layers for the multi-layer perceptron.

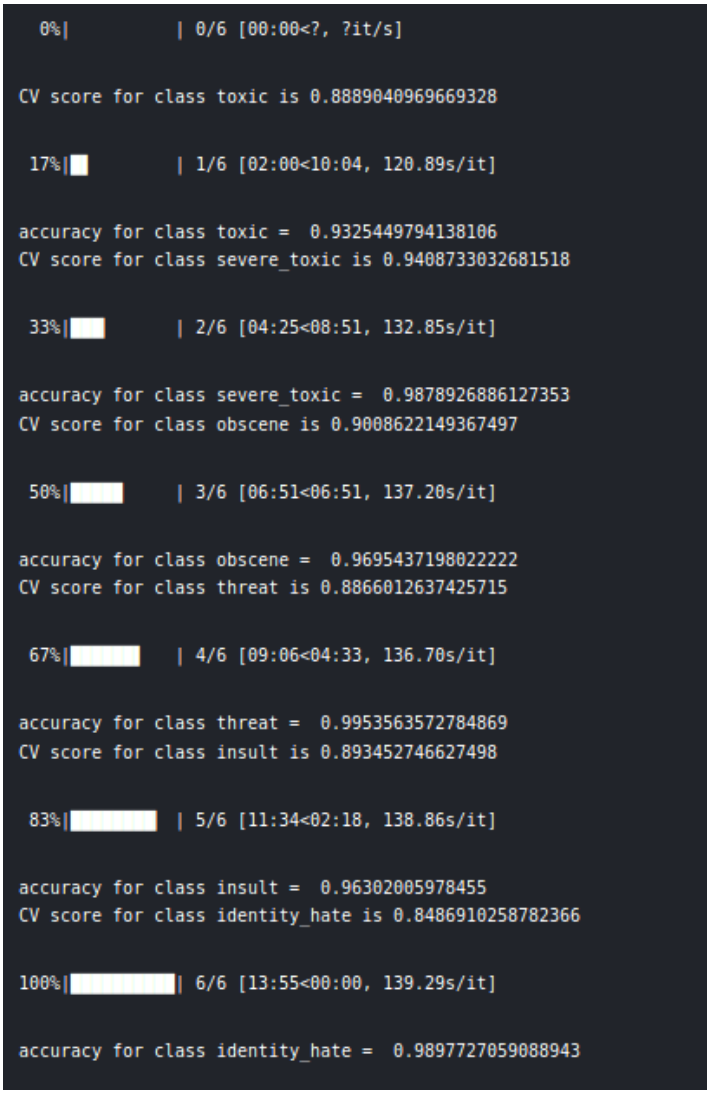

Figure 5: Results for Multi Layer Perceptron

- LSTM In Deep Neural Network :- We have used one layer of LSTM with 300 LSTM cells and one dense layer with one node for output.

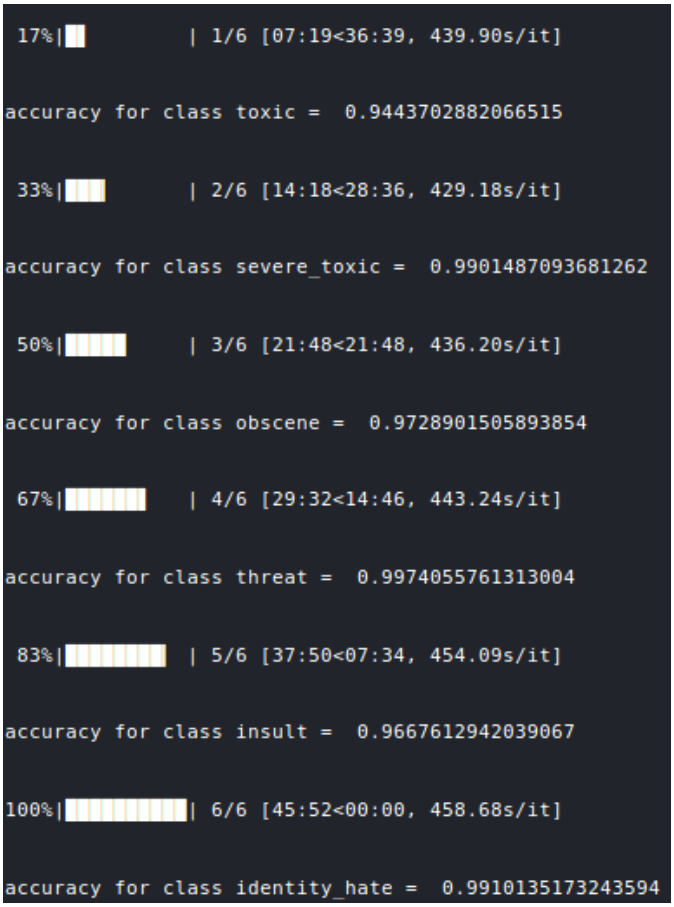

Figure 6: Results for LSTM

- Deep Neural Network :- We have used one Dense layer of 100 nodes with 'relu' activation, next layer is again dense layer with 10 nodes with same 'relu' activation and last layer is again dense layer with one node for output with sigmoid activation.

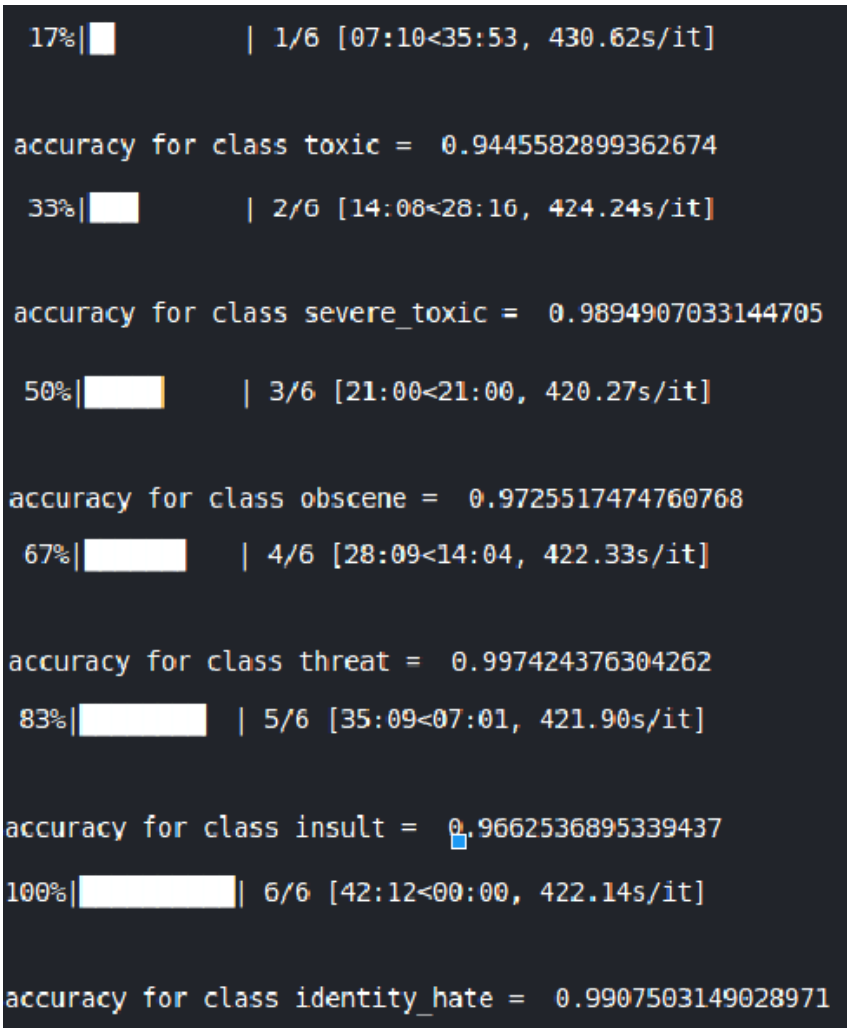

Figure 7: Results for Deep Neural Network

9 Results:-

The results for all the models for different level of toxicity is:-

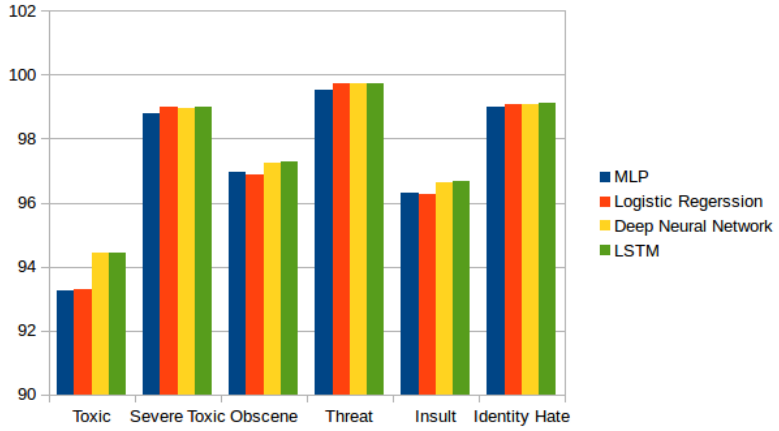

Figure 8: Results for all models

	A	B	C	D	E	F	G
1	Algorithm	Toxic	Severe Toxic	Obscene	Threat	Insult	Identity Hate
2	MLP	93.25	98.79	96.95	99.54	96.3	98.98
3	Logistic Regerssion	93.31	98.98	96.88	99.72	96.25	99.08
4	Deep Neural Network	94.45	98.94	97.25	99.74	96.62	99.07
5	LSTM	94.43	99.01	97.28	99.74	96.67	99.1

Figure 9: Histogram of accuracy for all models

10 Summary:-

We picked one of the hot-shot problem of social networking site, classifying toxic comments as the problem. We started with already available literature and analyzed them, we posed some problems in them and tried to solve them as per our understanding. Right from pre-processing to model-training we tried some already available solutions and then proposed our solution. On the given data our LSTM models tends to perform best among all available models being used for this purpose.