NLP And Deep Learning Course Final Report

Eviatar Nachshoni, Yosef Danan, Nir Son

December 2021

Abstract

The goal of this project is to detect offensive comments on social media. The data composed from 100,000 label samples of comments from different social media platforms like twitter, Kaggle, YouTube and etc. The focus in this project is on deep learning models.

1 Introduction

1.1 Data Exploration

The Data[1] consist of 100,000 comments from different social media platforms. Each entry is labeled as 'offensive' (1) or 'not offensive' (0).
A key observation is that the data is imbalance - 73.2% of 0 and only 26.8% of 1.

As can be seen in Figures 3 and 4, there is an obvious correlation between "bad" and "good" words to the respective label.

7	5	69705	Block me then you cocksucker - clever people don't give a fuck about editing Wikikpedia? Just because Wikipedia has turned you into its bitch, you can't see that bein	1
8	6	26449	` == Release Date == So what's the deal with this film? When is it being released? Can we get a more accurate date than, ``First Quarter of 2007?`` `	0
9	7	396326	== Kurt Cobain == This is the one Kurt Cobain suicided with	0
10	8	271338	== Vandalism is spreading lies == I think it is you who is a vandal, for you are writing lies about Jami. The pederastic Poetry should be removed, and I doubt that the	0
11	9	321365	Please stop making test edits to Wikipedia. It is considered vandalism, which, under Wikipedia policy, can lead to blocking of editing privileges. If you would like to e	0
12	10	9607	AHAHAHAHAHA Read the whole comment before trying to chime in. There is no needed discussion for templates. This is the JoshuaZ pattern incite then banner. Un	0
13	11	436631	RT @babybrucewayne: every time I see Kat's face I feel like puking #mkr	1
14	12	380196	Your from a family of donkeys.	1
15	13	95091	, 30 November 2012 (UTC) ::The artilce clearly states what happened in February 7th day President nasheed resigned, What proof do you have to say that the source	0
16	14	81302	He can go fuck himself sideways with a spiky spoon.	1
17	15	81986	== Alto Adige == I see you are trying to change the name again to the German point of view. Jan, you really a pathethic nationalistic neo-nazi POS excuse for a huma	1
18	16	137543	==Relevancy== Are you from New York? Are you well steeped in the history and contemporary culture of hardcore music? In my opinion, you should probably sto	1
19	17	236441	"\xa0shut ur nasty ass up u homo ass nigga,u still gona b sticking ur shit in a females asses and i know for a fact u either done it to a dude or thought about it,nasty fu	1
20	18	195057	` : not complete rubbish, it is pretty close to what it means. They didnt translate the ``faka wot`` part, maybe could have said ``Hey, Jerk! Do we have a problem?`` anc	0
21	19	131696	:Spawn bro, please don't tie your fate here to that of a burned out old man. You have MANY friends here...more than I. You're just not counting em all:) I know whe	0
22	20	276360	Your definition of vandalism is wack. Just because you don't like what's being said, it doesn't make it vandalism. You don't know all the facts, and are not an author	1
23	21	62513	` :::Actually Andrew, you have neither the policies nor the consensus to back up such a unilateral decision. I asked you to quote for me the passage in Wiki's rulebook!	0

Figure 1: a sample of the data

A few more statistics on the data:

- average comment length is 74.27398 words
- 3.7% of comments are less then 300 words long
- offensive comments tend to be shorter - the average for 1 is 62.2 words and for 0 is 78.6 words.

[1] data taken from https://data.mendeley.com/datasets/jf4pzyvnpj/1

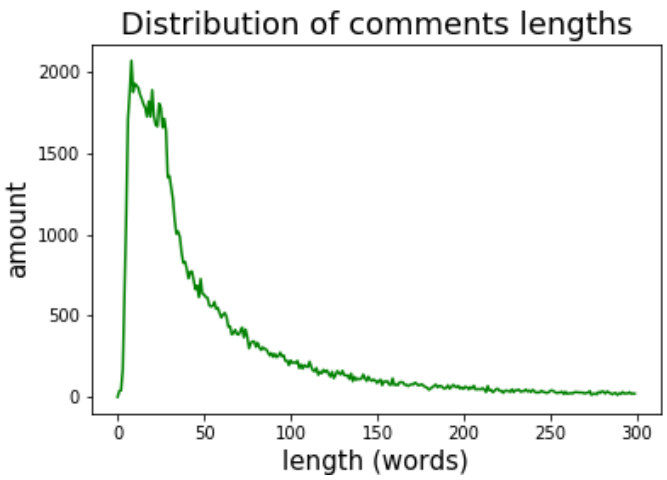


Figure 2: distribution of comments lengths

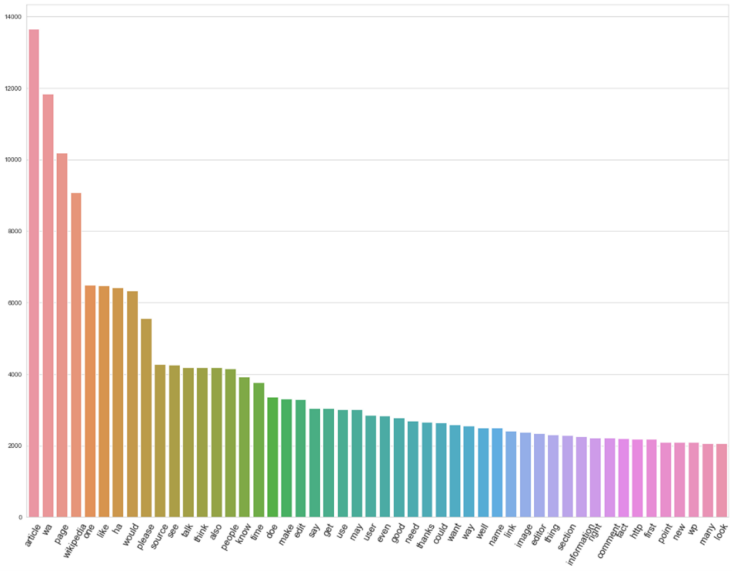


Figure 3: most common words in not offensive comments

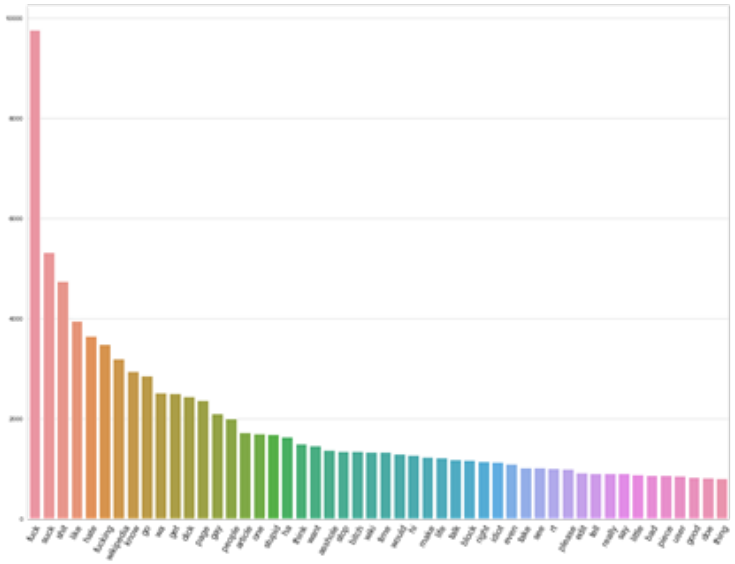


Figure 4: most common words in offensive comments

1.2 Scope

This project will present the results process of our attempt to solve the problem presented using different features extraction and deep learning techniques. The project is not presenting the specifics of the implementation.

2 Required Background and Related Work

The project require knowledge in neutral language processing (as learned in the course) and understanding in deep learning techniques, to the scope of the course.

In research and experiments we used the following sources:

- https://keras.io/api/
- "Cyberbullying Detection in Social Networks Using Deep Learning Based Models; A Reproducibility Study", Maral Dadvar and Kai Eckert, Stuttgart Media University, Germany

3 Project Description

3.1 Pre-Process

before training, The Data needed to be cleaned:

- remove not-ascii characters
- remove punctuation
- turn everything to lower case
- stem the words
- remove stop words
- replace digits with text
- Misspell correction

3.2 Feature Extraction

Data represented by a 2829X300 matrix, using google-news-300 encoding for each word, without doc length limiting, after deleting stop-words and punctuation's.

3.3 Model Training

The network has the following architecture: 100 3x300 convolution | dropout | 2825x1 max Pooling | Flatten |fully connected (300-1)
with relu in all activation except the last with sigmoid. Also using L1 & L2 regularization.

3.4 Results

	precision	**recall**	**f − score**
0	0.94	1	0.93
1	0.79	0.83	0.81

The accuracy of the model is 0.89.

4 Experiments

4.1 Features types

When it comes to features extraction from text, there are many ways to represent a text as a vector of numbers. In this project we used some known

way. below are short explanations for each of the ones we used:

name	description	variants
bag of words	count the amount of appearances of each word in the dictionary	- remove words that appear very low amounts of times - remove any word where the variance in the count is very low
word embedding / word2vec	turning each word into vector using word2vec. the text is represented by a matrix.	- set a fix size to the matrix, trimming or padding if needed
summed word2vec	turn the entire text into vector of numbers. in our implementation, this is done by converting each word using word2vec and taking the sum	-

4.2 Models

4.2.1 MLP

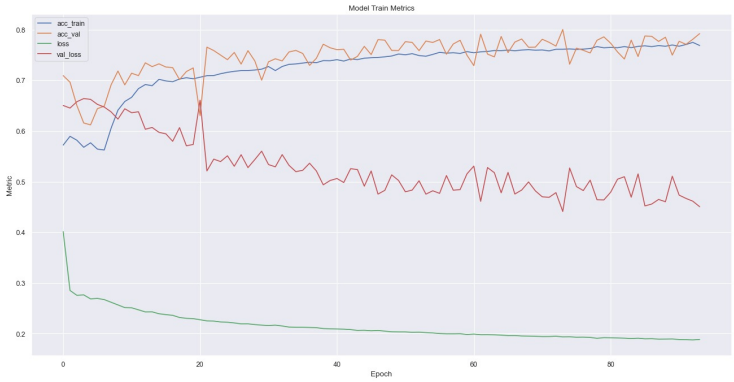
4.2.1.1 bag of words

A multi level perceptron with hidden layers of sizes (100,60,20), weighted loss, and the data represented using bag of words.

	precision	**recall**	**f − score**
0	0.94	0.72	0.81
1	0.53	0.87	0.66

4.2.1.2 summed word2vec

A multi level perceptron with hidden layers of sizes (200, 100, 50), weighted loss, learning rate 0.001 and the data represented using doc2vec.



The loss of the validation decrease very well. The accuracy almost 79

4.2.2 LSTM

With basic LSTM model and data represented using word embedding (Glove-50), the results were not good. Using LSTM with a fully connected network at the end, the results improved slightly, but a problem of over fitting occurred. The final model is bidirectional LSTM with weighted loss, L1 regularization and dropout.

	precision	**recall**	**f − score**
0	0.93	0.87	0.90
1	0.69	0.81	0.75

The accuracy of the model is 0.85.

4.2.3 BERT

BERT fine-tuning for classification, trained with 5000 samples, got 0.8877 accuracy with 3 epochs. further training could not be done due to lack of resources.

5 Conclusions

We notice that all models including complex models such as BERT can't achieved more than 0.9 accuracy score, that's probably because 90,000 words from the data set don't has real word2vec but rather default zero/random vector so there is lack of information in that representation