# Twitter Sentiment Extraction

Ankit Kumar
*Introduction to Machine Learning*
*EE769 Course Project*
Guide : Prof Amit Sethi

***Abstract*—Given a tweet, find what words in the tweet support a positive, negative, or neutral tweet sentiment..**

## I. Introduction

With all of the tweets circulating every second, it is hard to tell whether the sentiment behind a specific tweet will impact a company, a person's brand for being viral (positive) or devastate profit because it strikes a negative tone. Capturing sentiment in the language is essential when decisions and reactions are created and updated in seconds.There is a need to analyze which words lead to the sentiment description for the tweet.

## II. Approach to the Problem

It can be addressed similar to question answering NLP task. Question answering is a task in information retrieval and Natural Language Processing (NLP) that investigates software that can answer questions asked by humans in natural language. In Extractive Question Answering, a context is provided so that the model can refer to it and make predictions on where the answer lies within the passage. Question answering can be implemented using pre-trained models provided by the Huggingface Transformers library.

## III. Architecture, Input and Output

Pretrained Roberta transformer architecture is selected.III-A Pretrained Roberta-base tokenizer is selected for the tokenization of texts into numbers. Input is the tokenizer of both tweet text and sentiment. Output is two one-hot encoded arrays of input size whose one's position starting and ending position of the selected word in the tweet.

### A. Given data

- Tweet: ' what interview! leave me alone'
- Selected words: 'leave me alone'
- Sentiment: Negative

### B. Input and Output

- Token(tweet):[0,99, 1194, 328, 989, 162, 1937,2]
- Token(selected words):[0,989, 162, 1937,2]
- Token(Sentiment): [0, 2430, 2]
- Intersection position:[4, 5, 6]
- Input :[0, 99, 1194,328, 989, 162, 1937, 2, 2, 2430, 2]
- Output start: [0, 0, 0, 0, 1, 0, 0, 0, 0, 0, 0]
- Output end: [0, 0, 0, 0, 0, 0, 0, 1, 0, 0, 0]

## IV. Challenges

Labeling issues in the training dataset was the main challenge. There was some offset in the labeled selected words of the tweet. The full stop (.) was labeled some time inappropriately.

### A. Offset in a word Issue

- Tweet : I'm not sleeping at all until accepts my appology
- Selected words : I'm not sleeping at all un

### B. (.) issue

- Tweet : I'm sick and sad .... missing out on Martini Lounge tonight
- Selected words : sick and sad ..

## V. Final Model

TABLE I
Architecture

| Second last layer of of Pre-trained Roberta |
|---|
| Dropout layer |
| Conv1D layer |
| Flatten layer |
| softmax layer |

Cross-entropy loss function and Adam optimizer of learning rate 3e-05 were used for the final model.The final model was tested on fivefold cross-validation employing Jaccard score metric .

## References

[1] https://huggingface.co/transformers/glossary.html
[2] https://colab.research.google.com/github/huggingface/transformers/blob/master/noteboo transformers.ipynb
[3] https://nlp.seas.harvard.edu/2018/04/03/attention.htmlencoder-and-decoder-stacks
[4] https://www.kaggle.com/abhishek/roberta-inference-5-folds/notebook?select=pytorch$_{m}odel.bin$