**INTERNSHIP: PROJECT REPORT**

---

| | |
|---|---|
| Internship Project Title | Automate Sentiment analysis of textual comments and feedback |
| Project Title | Sentiment analysis of the IMDb dataset. |
| Name of the Company | Tata Consultancy Services |
| Name of the Industry Mentor | Soumyadip Mal |
| Name of the Institute | Delhi Technical Campus ,GGSIPU |

| Start Date | End Date | Total Effort (hrs.) | Project Environment | Tools used |
|---|---|---|---|---|
| 28th May 2020 | 28th June 2020 | 110 hours | Personal/ Home | Spyder, Terminal, Google Colab, Google Chrome |

Project Synopsis:

NLP (Natural Language Processing) is and AI component concerned with the interaction between human language and computers. Our Project's Aim was to analyze the dataset containing 50000 reviews/feedback of different movies out of which 25000 were positive and 25000 were negative and create a model which could then process these reviews and as an outcome could produce a result which would then allow us to decide whether a feedback is positive or negative.

Solution Approach:

We devised multiple models for a better comparison as to which algorithm/ model can achieve the highest accuracy as give better results in the least time possible.

The Very first step was to preprocess the data that was given to us and hence remove any unnecessary words or any special symbols that may have been used in our feedbacks. This is by far the most important step before making any model. We use regular Expressions to identify certain patterns of symbols and removed them and also some stop words (for ex: a, the ,an, is.. etc) which do not contribute in determining the sentiment/ eotion of any sentence.

1. Baseline Model: We used different type of vectorizers such as Count Vectorizer and TF-IDF Vectorizer for the feature extraction of our model and then using Logistic Regression we fit into our model the dataset that was provided to us to get the desired results. We also made use of the n-grams technique wherein we decide the max number of units to be used per sentence.

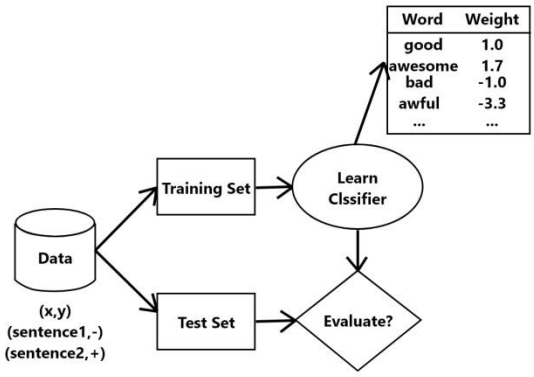
2. Support Vector Machines: We used The Count Vectorizer along with n-grams and instead of fitting our model using LR Algorithm we Used Support Vector Machine Classifier(SVC) to get a slightly better accuracy.

3. Final Model: For our final model we used the sequential model from the Keras inbuilt models to achieve the best accuracy. We made use of different layers including dense layers, Convolution layers and pooling layers to come up with a multilayer perceptron which could predict data at a very fast rate and with almost perfect accuracy.

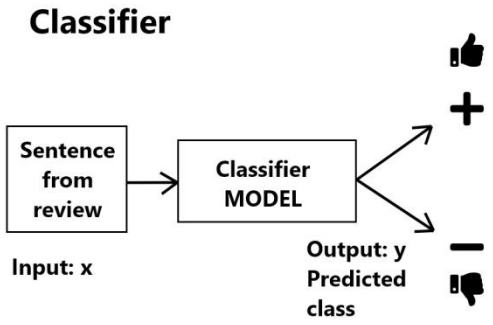
Assumptions:
As the primitive models were created using SVM's we assume the datasets that this model will work on will be small as SVM's give best results on small datasets. Introducing larger datasets may compromise with the learning/prediction time.

Project Diagrams:

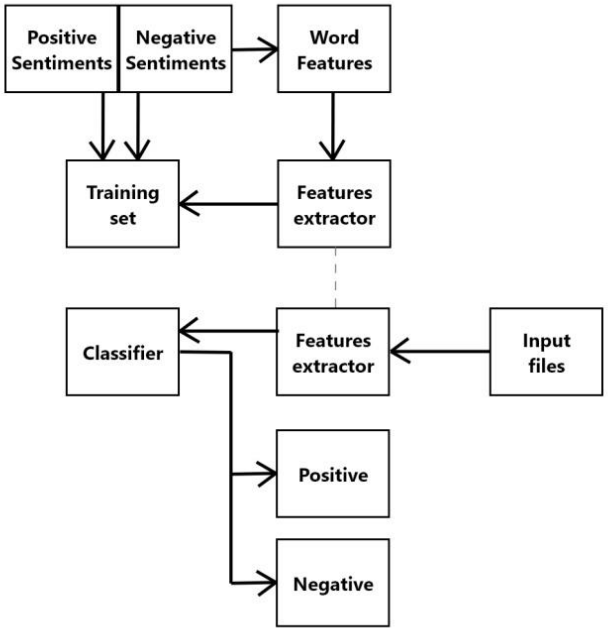
1. Basic Artificial Intelligence Model




2. A Simple classifier Model



3. Our Final model




Algorithms:

The Algorithms that were used in the making of different models were:

1. Vectorization
2. NLP
3. Logistic Regression
4. Support Vector Machines
5. Multilayer Perceptron (MLP)

Outcome:

| S No. | Model | Accuracy (%) | Algorithm Used |
|---|---|---|---|
| 1. | Baseline (Using Count Vectorizer) | 87.18 | Logistic Regression |
| 2. | Baseline (Using TF-IDF Vectorizer) | 88.2 | Logistic Regression |
| 3. | Improvised Model | 90.064 | Support Vector Machine |
| 4. | Final Model | 96.84 | Multilayer Perceptron, Artifical Neural Network |

| |
|---|
| Exceptions considered:<br><br>The dataset that is provided to us contains no neutral reviews which in turn help our model have better prediction accuracy. The Reviews are clearly either positive (>0.7) or negative (<0.3).<br>This is done in order to avoid ambiguity in the decision making process for our model. |
| Enhancement Scope:<br><br>1. The dataset can be preprocessed better.<br>2. Better and more number of stop words can be used.<br>3. Other Algorithms such as LSTM's or Complex ANN can be used.<br>4. While Working with a bigger dataset a dropout layer may also be added to avoid overfitting.<br>5. Hyper Parameter Tuning can also be done. |
| Link to Code and executable file: https://github.com/maheshwarimanvi13/TCS-Summer-InternShip |