INTERNSHIP: PROJECT REPORT

Internship Project Title	Automate detection of different emotions from textual comments and feedback
Project Title	Emotion Analysis on textual comments and feedback using MultinomialNB and LogisticRegression
Name of the Company	TCS iON
Name of the Industry Mentor	Debashis Roy
Name of the Institute	Alva's Institute of Engineering and Technology

Start Date	End Date	Total Effort (hrs.)	Project Environment	Tools used
29/09/2020	25/10/2020	60	Google Colab	Python 3 scikit-learn Natural Language Toolkit

Project Synopsis:

Introduction

Detecting emotional state of a person by analyzing a text document written by him/her appear challenging but also essential many times due to the fact that most of the times textual expressions are not only direct using emotion words but also result from the interpretation of the meaning of concepts and interaction of concepts which are described in the text document. Recognizing the emotion of the text plays a key role in the human-computer interaction. Emotions may be expressed by a person's speech, face expression and written text known as speech, facial and text-based emotion respectively. Sufficient amount of work has been done regarding to speech and facial emotion recognition but text-based emotion recognition system still needs attraction of researchers. In computational linguistics, the detection of human emotions in text is becoming increasingly important from an applicative point of view.

What is Emotion Analysis on text?

Emotion Detection and Recognition from text is a field of research that is closely related to Sentiment Analysis. Sentiment Analysis aims to detect positive, neutral, or negative feelings from text, whereas Emotion Analysis aims to detect and recognize types of feelings through the expression of texts, such as anger, fear, joy, sadness, love and surprise. Emotion Analysis provides a deeper insight of consumer emotions. Categorizing feedback and analyzing its emotion by picking up words, contexts,

patterns, behaviors. This can be even taken to the level of individual's expressive capability of a particular situation.

Objective

- To develop an algorithm to detect different types of emotions contained in a collection of English sentences or a large paragraph.
- To calculate the attributes such as CV Score, Accuracy, Precision, Recall and F1 Score.

Solution Approach:

- Plot the graph showing different emotions and its count.
- Text cleaning operation is done on the data. Text cleaning includes:
  - ✓ Set all words to lowercase.
  - ✓ Removing mentions.
  - ✓ Removing hashtags.
  - ✓ Removing URL's.
  - ✓ Convert the emojis into one word.
  - ✓ Removing punctuations, digits and stopwords.
  - ✓ Apply the PorterStemmer to keep the stem of the words.
- Apply Text classifiers on the data. Classifiers used are:
  - ✓ CountVectorizer.
  - ✓ TF-IDF Vectorizer.
  - ✓ Word2Vec.
- Use grid-search method for selecting the best parameters for the model.
- Use Text Classification Algorithms:
  - ✓ MultinomialNB.
  - ✓ LogisticRegression.
- Calculate performance metrics which are Accuracy, Precision, Recall and F1 score.

Assumptions:

- Different words in the text represent different emotions. The maximum count of a particular emotion in the text is considered as final one.
- Document focuses on a single object (not true in discussion posts, blogs, etc.) and contain opinion from a single opinion holder.

Project Diagrams:

- Textual comments and feedback are given as input to the Tokenization and Text cleaning process to remove mentions, hashtags, URL's, digits, stopwords, etc.
- Text classifiers such as CountVectorizer, TF-IDF Vectorizer, Word2Vec are used to classify text or count words in the text, URL's, mentions, hashtags, emoji's, capital words, etc.
- Grid-search method is used for selecting the best parameters for the model along with the Text classification algorithm such as MultinomialNB and LogisticRegression.
- Performance metrics will be the output from the model obtained, which are CV score of the model, accuracy, precision, recall and F1 score.

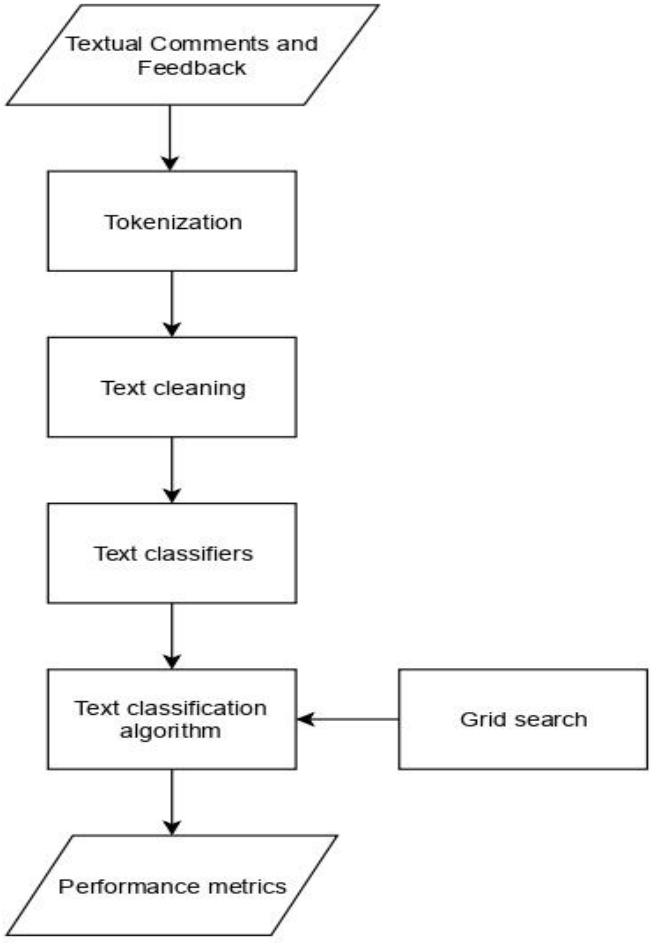

Fig: Flow diagram for the model

Algorithms:

MultinomialNB

The multinomial Naive Bayes classifier is suitable for classification with discrete features (e.g., word counts for text classification). The multinomial distribution normally requires integer feature counts. However, in practice, fractional counts such as tf-idf may also work.

Syntax:

class sklearn.naive_bayes.MultinomialNB(*, alpha=1.0, fit_prior=True, class_prior=None)

Methods

`fit`(X, y[, sample_weight])	Fit Naive Bayes classifier according to X, y
`get_params`([deep])	Get parameters for this estimator.
`partial_fit`(X, y[, classes, sample_weight])	Incremental fit on a batch of samples.
`predict`(X)	Perform classification on an array of test vectors X.
`predict_log_proba`(X)	Return log-probability estimates for the test vector X.
`predict_proba`(X)	Return probability estimates for the test vector X.
`score`(X, y[, sample_weight])	Return the mean accuracy on the given test data and labels.
`set_params`(**params)	Set the parameters of this estimator.

LogisticRegression

Logistic regression is a supervised learning classification algorithm used to predict the probability of a target variable. The nature of target or dependent variable is dichotomous, which means there would be only two possible classes. In simple words, the dependent variable is binary in nature having data coded as either 1 (stands for success/yes) or 0 (stands for failure/no).

Syntax:

class sklearn.linear_model.LogisticRegression(penalty='l2', *, dual=False, tol=0.0001, C=1.0, fit_intercept=True, intercept_scaling=1, class_weight=None, random_state=None, solver='lbfgs', max_iter=100, multi_class='auto', verbose=0, warm_start=False, n_jobs=None, l1_ratio=None)

Methods

`decision_function`(X)	Predict confidence scores for samples.
`densify`()	Convert coefficient matrix to dense array format.
`fit`(X, y[, sample_weight])	Fit the model according to the given training data.
`get_params`([deep])	Get parameters for this estimator.
`predict`(X)	Predict class labels for samples in X.
`predict_log_proba`(X)	Predict logarithm of probability estimates.
`predict_proba`(X)	Probability estimates.
`score`(X, y[, sample_weight])	Return the mean accuracy on the given test data and labels.
`set_params`(**params)	Set the parameters of this estimator.
`sparsify`()	Convert coefficient matrix to sparse format.

Outcome:

- Using Count Vectorizer and MultinomialNB

Emotion	Precision	Recall	F1 score
Anger	0.87	0.77	0.81
Fear	0.80	0.76	0.78
Joy	0.82	0.91	0.86
Love	0.73	0.61	0.66
Sadness	0.86	0.89	0.87
Surprise	0.77	0.39	0.52
Macro Avg	0.81	0.72	0.75
Weighted Avg	0.83	0.83	0.82
Accuracy	0.83		
Best CV Score		0.823	
Test score with best estimator		0.829	

- Using Count Vectorizer and Logistic Regression

Emotion	Precision	Recall	F1 score
Anger	0.85	0.81	0.83
Fear	0.85	0.79	0.82
Joy	0.84	0.89	0.86
Love	0.73	0.68	0.70
Sadness	0.90	0.90	0.90
Surprise	0.73	0.66	0.69
Macro Avg	0.82	0.79	0.80
Weighted Avg	0.85	0.85	0.85
Accuracy	0.85		
Best CV Score		0.839	
Test score with best estimator		0.848	

- Usage of Count vectorizer with Logistic Regression algorithm provides more accuracy than MultinomialNB. Logistic Regression provides more precision while detecting sadness emotion.

- Using TF-IDF Vectorizer and MultinomialNB

Emotion	Precision	Recall	F1 score
Anger	0.96	0.61	0.74
Fear	0.88	0.58	0.70
Joy	0.70	0.97	0.81
Love	0.93	0.25	0.39
Sadness	0.79	0.89	0.89
Surprise	1.00	0.10	0.18
Macro Avg	0.88	0.58	0.61
Weighted Avg	0.81	0.77	0.74
Accuracy	0.77		
Best CV Score		0.761	
Test score with best estimator		0.769	

- Using TF-IDF Vectorizer and Logistic Regression

Emotion	Precision	Recall	F1 score
Anger	0.90	0.46	0.61
Fear	0.78	0.55	0.65
Joy	0.71	0.92	0.80
Love	0.74	0.36	0.49
Sadness	0.74	0.83	0.78
Surprise	0.04	0.03	0.04
Macro Avg	0.65	0.53	0.56
Weighted Avg	0.73	0.72	0.70
Accuracy	0.72		
Best CV Score		0.727	
Test score with best estimator		0.722	

- Usage of TF-IDF vectorizer with MultinomialNB algorithm provides more accuracy than Logistic Regression. MultinomialNB provides very high precision while detecting anger, love and surprise emotion.
- Logistic Regression using TF-IDF vectorizer fails to detect surprise emotion.

Performance metrics:

- **Accuracy**: the percentage of texts that were predicted with the correct tag.
- **Precision**: the percentage of examples the classifier got right out of the total number of examples that it predicted for a given tag.
- **Recall**: the percentage of examples the classifier predicted for a given tag out of the total number of examples it should have predicted for that given tag.
- **F1 Score**: the harmonic mean of precision and recall.
- **CV score**: Cross-validation is a statistical method used to estimate the skill of machine learning models.

Plot showing precisions of different models:

Plot showing Accuracy and Best CV Score of different models:

Enhancement Scope:

- Emotion analysis bases its results on factors that are so inherently humane, it is bound to become one of the major drivers for many business decisions in future. Improved accuracy and consistency in text mining techniques can help overcome some current problems faced in Emotion analysis.
- The cultural affiliations or mother tongue of an individual greatly influence their expressed emotions toward situations. The availability of resources in other languages such as French, Spanish, Hindi, and so on can greatly change the narrative and encourage research in the field of natural language processing.

Link to Code and executable file:

Google Colab link:

https://colab.research.google.com/drive/1djMmlkL6hXNO88cod0HCMCUFU2cLYoo5?usp=sharing

References:

1. https://www.tutorialspoint.com/machine_learning_with_python/machine_learning_with_python_classification_algorithms_logistic_regression.htm#:~:text=Logistic%20regression%20is%20a%20supervised,be%20only%20two%20possible%20classes.&text=Mathematically%2C%20a%20logistic%20regression%20model,as%20a%20function%20of%20X.
2. https://onlinelibrary.wiley.com/doi/full/10.1002/eng2.12189
3. https://colab.research.google.com/github/jakevdp/PythonDataScienceHandbook/blob/master/notebooks/04.11-Settings-and-Stylesheets.ipynb
4. https://www.kaggle.com/c/sa-emotions
5. https://www.youtube.com/watch?v=dyN_WtjdfpA&list=PLhTjy8cBISEoOtB5_nwykvB9wfEDscuEo
6. https://scikit-learn.org/stable/tutorial/index.html
7. https://ieeexplore.ieee.org/abstract/document/7877424
8. https://www.scipy.org/docs.html
9. https://github.com/topics/emotion-analysis
10. https://pandas.pydata.org/pandas-docs/version/0.15/tutorials.html