**Sentiment-Analysis-on-Online-Product-Reviews**

The objective of this project is to classify whether upcoming product will have positive or negative Sentiments.

**Sentiment Analysis**

Sentiment analysis and Opinion mining is the computational study of User opinion to analyze the social, psychological, philosophical, behavior, and perception of an individual or a group of people about a product, policy, services, and specific situations using Machine learning techniques. Sentiment analysis is an important research area that identifies the people’s sentiment underlying a text and helps in decision-making about the product.

**STEPS**

* Data Collection
* Feature Extraction
* Model Selection
* Model Training and Evaluation

**Data Collection**

**Source**: <https://data.world/datafiniti/grammar-and-online-product-reviews>

**Customer Product reviews**

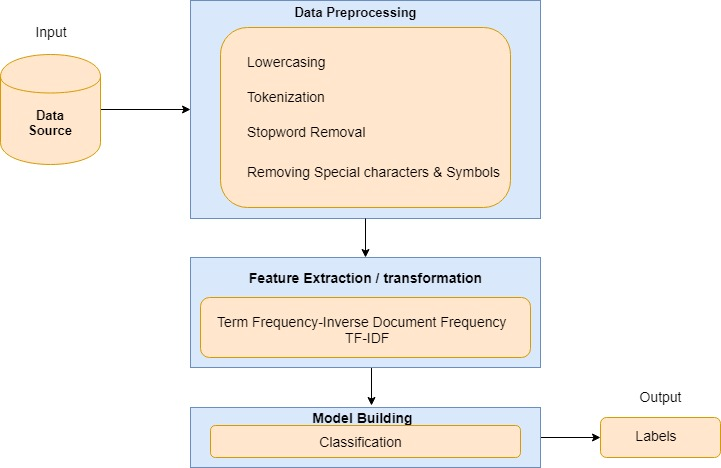
A customer review is a review of a product or service made by a customer who has purchased and used, or had experience with the product or service. Customer reviews are a form of customer feedback on electronic commerce and online shopping sites.

90% of consumers read online reviews before purchasing a product and 88% of consumers trust online reviews as much as personal recommendations.

The Online Product Reviews dataset includes following features:

* Dependent feature: reviews.rating (Unhappy,OK,Happy)
* Independent features: review.text (Reviews)

**Workflow**



1. Data loading.

2. Data Preprocessing

* Remove Punctuations,special symbols and special characters.
* Stopword Removal
* Tokenization
* Stemming

**WordCloud**



**Feature Extraction**

In order to make sense to our machine learning algorithm we have converted each review to a numeric representation which is called 'Vectorization'.

The system uses TF-IDF Vectorizer (Term Frequency-Inverse document frequency) that transforms a count matrix to a normalized frequency representation in float.

Splitting the data into Train and Test set (70-30).

**1. TF-IDF**

1.Count how many times does a word occur in each message (Known as term frequency)

2.Weigh the counts, so that frequent tokens get lower weight (inverse document frequency)

* TF(t) = (Number of times term t appears in a document) / (Total number of terms in the document).
* IDF(t) = log\_e(Total number of documents / Number of documents with term t in it).

Example: Consider a document containing 100 words wherein the word dog appears 3 times. The term frequency for dog is (3 / 100) = 0.03. Assume total of 10 million documents and the word dog appears in one thousand of these. Then, the inverse document frequency is calculated as log(10,000,000 / 1,000) = 4.

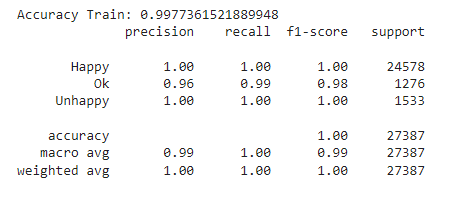
**Model Building**

Building Support Vector Machine (SVM) SGDClassifier on feature vectors.

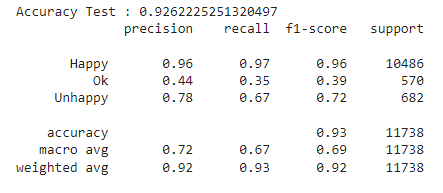
Support Vector Classifier (SVC) and Stochastic Gradient Descent (SGD) classifiers are commonly chosen for sentiment analysis tasks due to their effectiveness in handling high-dimensional feature spaces, flexibility in defining decision boundaries, and efficiency in training large-scale models.

**Results**

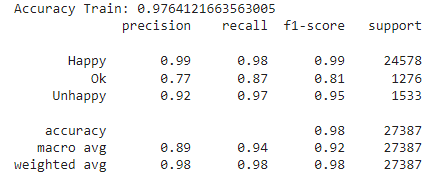
**SVC Train**



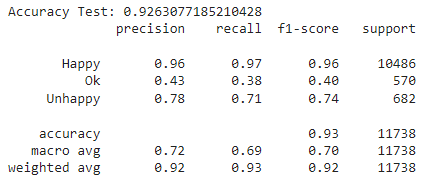
**SVC Test**



**SGD Train**



**SGD Test**



Thank You!

Janani Shree S D