**NATURAL LANGUAGE PROCESSING MINI PROJECT**

**SENTIMENT ANALYSIS**

What is Sentiment Analysis?

Sentiment Analysis (or sentiment classification) is a category of text classification where a given phrase or sentence is categorized into negative, positive or neutral attributes. Generally, a classifier does this by labelling the phrases into two attributes- negative and positive- to keep the classification binary.

Since recent times, attributes like, slightly positive, slightly negative are also into considerations.

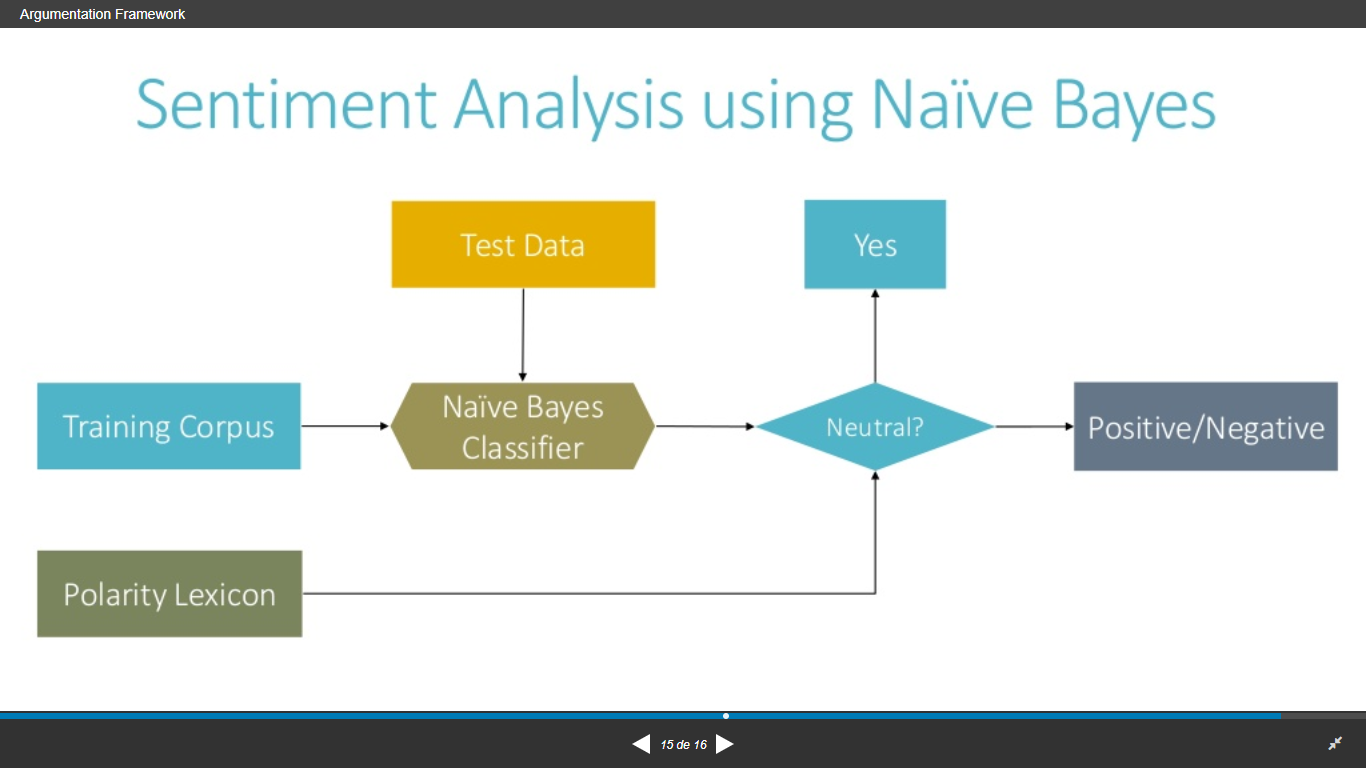


Fig. 01

**Problem Statement**: To train a model to recognize a review as positive or negative using Naive Bayes Classifier. We will be using a binary classification- positive and negative.

**Flowchart:**

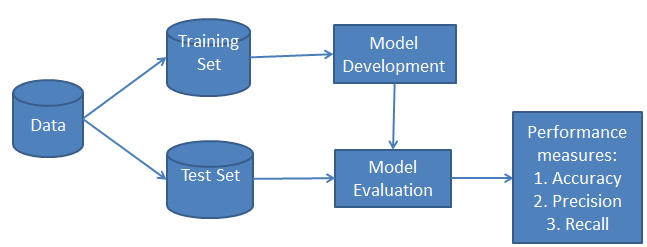


Fig.02

**Method:**

1. Reviews are divided into- 10% testing set & 90% training set.
2. Training set – builds positive words and negative words dictionary.
3. Calculating important probability values,

*P(word|negative) and P(word)*

***P(word|positive)=Nword\_pos /Nall\_pos***

***P(word|negative)=Nword\_neg /Nall\_neg***

***P(word)=Nword /Nall\_word***

Where,

|  |  |  |
| --- | --- | --- |
| ***Nword\_pos -***equals the number of times a word appears in the positive dictionary. | | |
| ***Nall\_pos -*** equals the number of all the positive sentiment words from the training set (words are counted if it appears repetitively.) | | | |
| ***Nword\_neg****-* equals the number of times a word appears in the positive dictionary.  ***Nall\_neg -*** equals the number of all the negative sentiment words from the training set (words are counted if it appears repetitively.) | |
| ***Nword -***equals the number of times a word appears in the positive dictionary.  *-*equals 1, when there is no showing up of the word in the dictionary, which is a smoothing method. |

|  |
| --- |
| ***Nall\_word -*** equals the number of all words.   1. The most useful words (including bigrams) are used for deciding sentiment by looking at the proportion of   ***Power(word)=P(word|positive)/(p(word|positive)+p(word|negative))***  Where the bigger ***Power(word)*** is the more useful a word is.   1. There is a fnction developed to include trigrams. It’s worth noticing that trigramList consists of unigrams, bigrams and trigrams. |

**Conclusion:**

Thus, we achieved the model to classify negative and positive sentiments using Naive Bayes Classifier. So we implemented Sentiment Analysis.

**OUTPUT:**

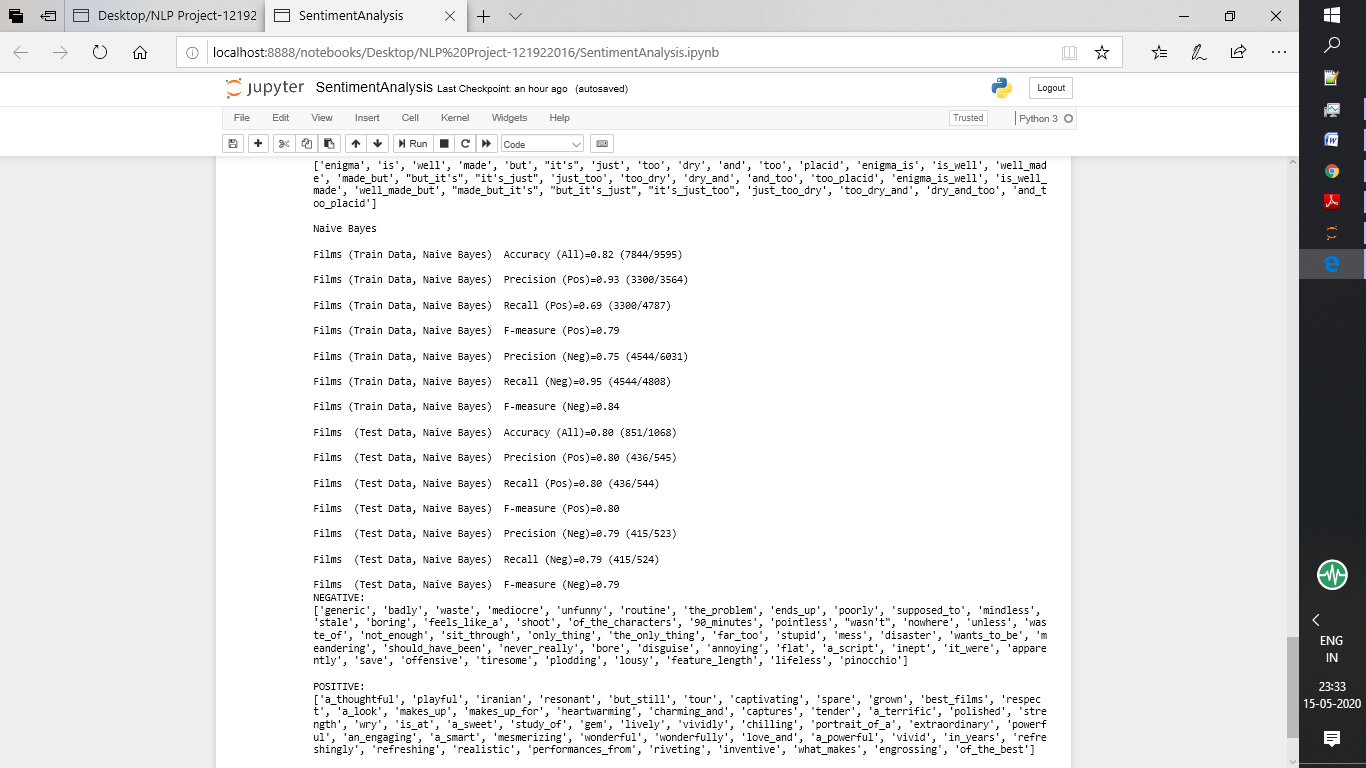


Fig.03