# Text Mining

For text mining we have two datasets, one with medicine, condition and reviews given by users with ratings. Second dataset has listed reviews but with no ratings.

By text mining, we will read the reviews and identify patterns between the ‘words’ and ratings. Using this, we will create a prediction model which will identify similar set of words and predict the ratings based on those words.

## Data Preparation

Following steps are taken for data preparation:

1. Normalization:
2. Removing unnecessary characters from the reviews (ex. Html encoding, special characters etc)
3. Removing words with length less than or equal to 2.
4. Removing commonly used words using stopwords. (ex is, a, of etc)
5. Removing words which are different forms of the already present words using lemmatizer (ex: call – called, called would be removed)
6. Removing other columns and only saving normalized cleaned reviews and ratings to a CSV (cleaned\_data.csv)
7. Dividing Normalized Reviews and Ratings as ‘data’ (input) and ‘target’.
8. Using Term Frequency- Inverse Document Frequency weight to score words in normalized reviews and creating matrix of words. Keeping minimum document frequency as 30, ngram range of (1,3). ‘ngram’ means co-occurring words.

## Model Creation and Cross Validation

For this task, we have a rating column with 2 categories, ‘High’ and ‘Low’ as target variable and Text reviews (tokens) as input variables.

We will be creating two classification models (Support Vector and Naïve Bayes) and will be choosing the best model based on the evaluation metrics.

Both models are cross validated using K-folds cross validation. With k=10, meaning the models will be evaluated 10 times and the model with best evaluation score would be final model saved for deployment and predicting the ratings on the other dataset.

## Model Evaluation

The target ‘Rating’ in the dataset is not balanced. There are 17899 ‘High’ and 5803 ‘Low’, hence accuracy can not be used as evaluation metric for these models.

Since the reviews are given by the users of medicine, If a medicine has ‘Low’ as rating and the model predicts as ‘High’ than there is a chance of a selling/purchasing faulty medicine (could have side effects). Hence the best model should have **minimum number of false positives.**

**Using Precision score for evaluation on each model and cross validating it 10 folds,**

**For Support Vector the results are -**

**For Naïve Bayes:**



Taking Average Precision score of all 10 folds for both the models and comparing, clearly the model for predicting minimal false positives for this dataset is **Naïve Bayes.** Saving the final model.



## Model Deployment

Predicting the Ratings on the other dataset, using the Naïve Bayes model created above with following steps:

1. Loading the model created above
2. Reading the vocabulary created above (final normalized reviews’ token)
3. Creating Normalized Reviews for No Rating dataset, Saving these without other columns in a CSV file.
4. Creating TF-IDF matrix for these cleaned reviews
5. Using Model above, predicting ratings for this TF-IDF matrix(cleaned reviews) and saving the new ratings in a CSV.