**Project : Movie Review Sentiment Analysis**

1. **Introduction**

We have been provided with a dataset consisting of 50,000 IMDB movie reviews, where each review is labelled as positive or negative. The goal of this project to build a binary classification model to predict the sentiment of a movie review.

1. **Technical Details**

# Vectorization

In order to represent a documents in vector space, we have to create mappings from terms to term IDS. We use word *terms* instead of *words* because they can be arbitrary n-grams not just single words. We represent a set of documents as a sparse matrix, where each row corresponds to a document and each column corresponds to a term.

# Vocabulary-based vectorization

We used create\_vocabulary() function mark each of unique terms with a unique ID.

We use an iterator to create the vocabulary.Below is the details how it works -

We created an iterator over tokens with the itoken() function. Iterator abstraction allows us to hide most of details about input and to process data in memory-friendly chunks.

We built the vocabulary with the create\_vocabulary() function.

**Model:**

Finally we created binary classification model to predict reviews' sentiment.

To create the model we use xgboost function.

XGBoost belongs to a family of boosting algorithms that convert weak learners into strong learners. A weak learner is one which is slightly better than random guessing. it uses gradient boosting (GBM) framework at core.

We will train decision tree model using the following parameters:

* objective = "binary:logistic": we will train a binary classification model .
* eval.metric = auc , we use auc because we need to predict area under cover.It allows us to monitor two new metrics for each round, logloss and error.
* Eta = 0.09 , eta is the learning rate for xgboost. Learning rate (eta) must be set as low as possible. However, as the learning rate (eta) gets lower, you need many more steps (rounds) to get to the optimum
* nrounds = 3000: there will be 3000 passes on the data, it will enhance the model by further reducing the difference between ground truth and prediction.

Benefit of Using xgboost –

* **Parallel Computing:** It is enabled with parallel processing. Because it uses the parallel computing, it is fast.
* **Missing Values:** XGBoost is designed to handle missing values internally.
* **Regularization**:  Regularization is a technique used to avoid overfitting in linear and tree-based models.

**Accuracy**

Mysubmission.txt

In this submission file, we output the probability of a document being positive or negative, which are all decimals. In other words, The prediction is between [0,1]. The result shows accuracy is 0.955 and above.

1. **Conclusion**

We have created an efficient model to predict the weekly sales for these stores by department. Our model is computationally and spatially efficient and can be applied to large scale. Our results are promising.

**Results Obtained from Evaluation**

Area under curve = 95.5%

**Running Time** – 1 mins 40 sec

**System Information** - Windows Laptop, i7 8550 CPU with 1.8 GHz 4 Core Processor, 16 GB RAM