Experimental Report

Muhammad Hamza – [*masterhmhamza.ml@gmail.com*](mailto:masterhmhamza.ml@gmail.com)

[Dataset 2](#_Toc907203034)

[Exploratory data analysis 3](#_Toc2024020698)

[Processing 4](#_Toc907322724)

[Model Training 4](#_Toc361521557)

# Dataset

Two datasets were used in this experiment. Both datasets were taken from Kaggle[[1]](#footnote-17694). Datasets are the collection of reviews given by customers/end-user about a *Restaurant* [[2]](#footnote-15839)and about *movies on IMDB*[[3]](#footnote-29425). Both datasets are annotated manually and given an integer value either 0: negative review or 1: positive review. Datasets have 2 columns:

1. Text – body of review mostly written in English language
2. Label – manually annotated label of review – 0 or 1

## Exploratory data analysis

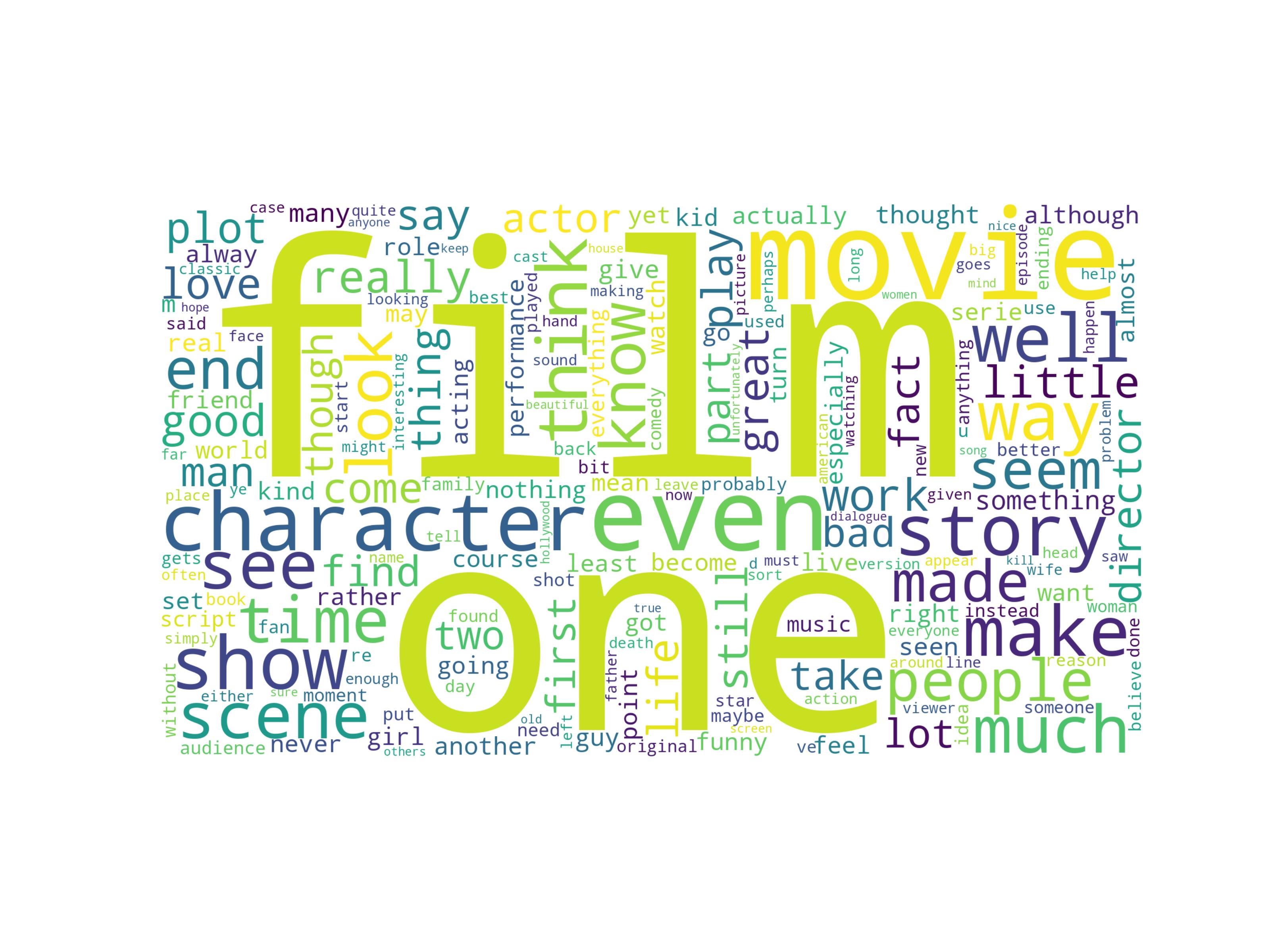
**Class Distribution** in *Restaurant* dataset was equal. 500 positive reviews and 500 negative reviews. In *IMDB* dataset, class distribution imbalance was negligible; 22514 negative reviews and 22486 positive reviews.

To get the overall summary of the datasets, I drew *wordcloud*. WordCloud highlights the most used words. This can be drawn after preprocessing the dataset, otherwise stop-words and punctuations will dominate the graph suppressing the useful verb/nouns in the dataset.



Figure 1: Word cloud for Restaurant dataset

The highlighted words in cloud also indicate the subject and context of dataset. Similarly, we can draw a word cloud for *IMDB* dataset. Below image shows the word cloud of IMDB dataset, and highlighted words also describe the context of data.



## Processing

Names of columns in datasets were different, for ease-of-access, after loading the dataset, names of columns were set to ‘text’ and ‘label’. *Restaurant* dataset had 1000 reviews. 800 reviews were separated for *training dataset* and other 200 reviews were selected as *test dataset*. *IMDB* dataset had 40,000 reviews as *training dataset* and 5,000 reviews as *test dataset*.

1. Next step was to convert entire dataset into lower case format.
2. Then we removed stop-words and punctuations from dataset.
3. At 3rd preprocessing step, I removed HTML, mentions, trends from dataset.
4. At last, I applied lemmatization on dataset.

# Model Training

I trained 3 models on both datasets. *IMDB* dataset was larger in size, so it gave model larger space to learn than *Restaurant* dataset. This was a classification task, so I used machine learning classifier models from *sklearn* module of Python language. After preprocessing the dataset, the dataset was sent into a *sklearn* pipeline. Pipeline consisted of following items:

1. CountVectorizer – to convert raw text of each review into a form of bag-of-words vectors
2. TfidfTransformer – this accepts a matrix as input, and calculate Tfidf over it
3. SelectKBest – select *k* best features based on *chi* score. *K* varies dataset to dataset and experiment to experiment. This must be tweaked to get best results.
4. Last one was the model class. For each model it was one of:
   1. LinearSVC – SVM model
   2. MultinomialNB - Naïve Bayes model
   3. RandomForestClassifier – Random Forest Model

These models have parameters as well, they needed to be tuned for getting optimal results.

# Model Evaluation

I drew confusion matrices for each model and for each dataset. To get a score for measure model’s performance, I used *Accuracy score, Precision score, Recall score and F1—Score.* Our dataset was almost balanced, so *Accuracy score* can select as base measurement metric to select the most powerful model.

Here is a comparison table of performances of model on *IMDB* dataset.

Note: The performance metrices are ranges between 0-1 (default output way of *sklearn*)

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Model Name | **Accuracy Score** | Precision Score | Recall Score | F1 Score |
| **SVM** | **0.84** | **0.87** | 0.81 | **0.84** |
| **Naïve Bayes** | **0.84** | 0.84 | **0.83** | **0.84** |
| Random-Forest | 0.83 | 0.82 | **0.83** | 0.83 |

Here is a comparison table of performances of models on *Restaurant* dataset.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Model Name | **Accuracy Score** | Precision Score | Recall Score | F1 Score |
| **SVM** | **0.78** | 0.76 | **0.81** | **0.78** |
| Naïve Bayes | 0.75 | 0.75 | 0.76 | 0.75 |
| Random-Forest | 0.74 | **0.85** | 0.60 | 0.70 |

From the comparison tables, this can be concluded that SVM model stands tall among other model tested in this experiment.

1. https://www.kaggle.com/datasets [↑](#footnote-ref-17694)
2. <https://www.kaggle.com/datasets/vigneshwarsofficial/reviews> [↑](#footnote-ref-15839)
3. https://www.kaggle.com/datasets/columbine/imdb-dataset-sentiment-analysis-in-csv-format [↑](#footnote-ref-29425)