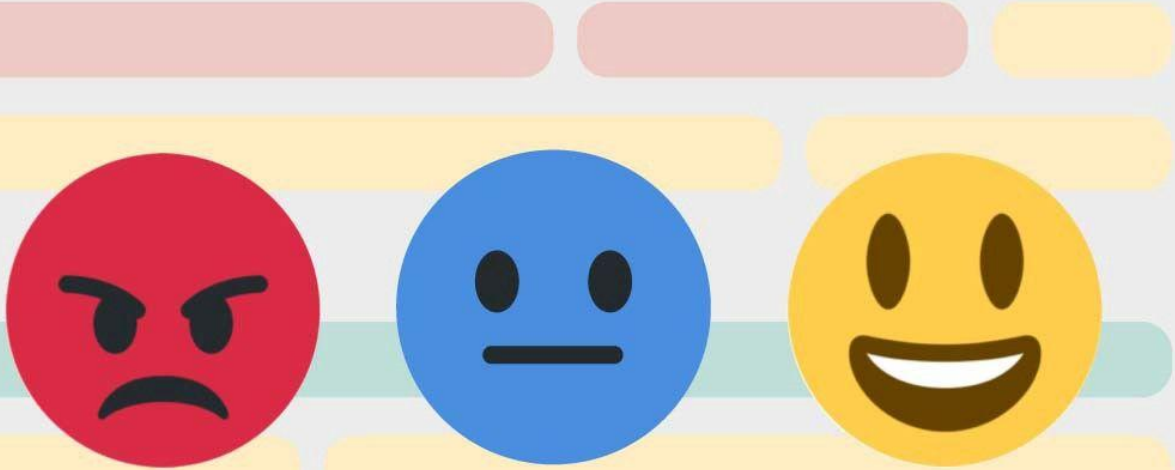
Sentiment Analysis



Overview

1. Problem Statement
2. Data Information
3. Frequency Analysis
4. Machine Learning Modeling (Classifiers)
    a. Preprocessing
    b. Model Evaluation and Optimization
5. Machine Learning Modeling (BERT)

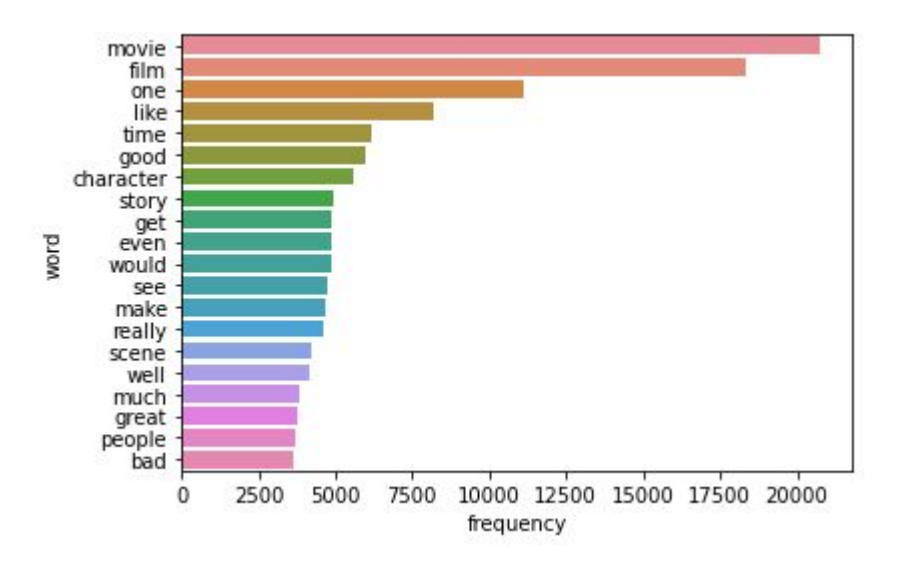
Problem Statement

IMDB wants to deeply analyze the movie review and figure out whether the review contains negative or positive sentiment.

Dataset: IMDB_Dataset.csv

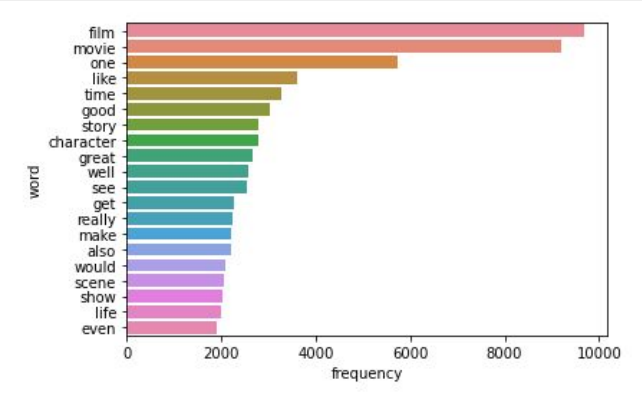
- The file has 2 columns with 50,000 observations.
- Data doesn’t have any missing values.
- Dependent Variable will be ‘sentiment’ variable.
- Independent Variable will be ‘review’ variable.

Frequency Analysis
Both Positive and Negative Sentiment

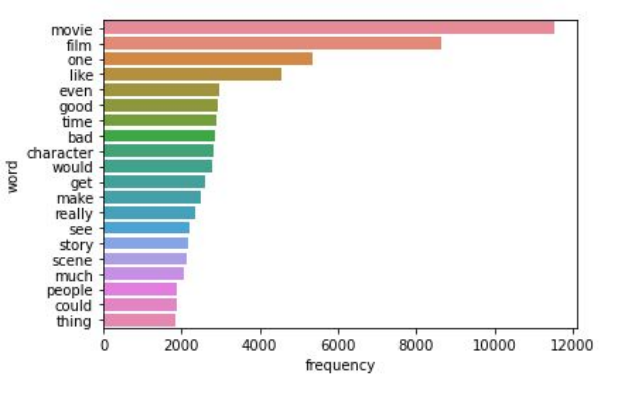


Top 2 mentioned words are ‘movie’ and ‘film’.
Since it includes both positive and negative reviews, chart presents both words ‘good’ and ‘bad’.

Frequency Analysis
Each Positive and Negative Sentiment



Positive Sentiment
Word ‘bad’ disappeared.



Negative Sentiment
On the other hand, the frequency chart for only negative review still has word ‘good’. This means that even though sentiment of review is negative, many reviewers still use the word ‘good’.

Preprocessing

- Subset dataset 10,000 out of 50,000 observations
- Clean texts from ‘review’ variables
    - Lower case
    - Delete punctuation except ‘.’,’?’,’!’
    - Remove URLs
    - Remove text ‘br’
    - Remove stopwords
    - Lemmatize word
- Verify cleaning dataset gives better outcome

Machine Learning Model (Classifiers)

Preprocessing

Data Subset & Data Cleaning	Modeling	Model Evaluation
The whole observation is 50,000. Since it took so long to train the dataset, I initially subset 10,000 observation samples to train and test the model.	I used two different natural language processing techniques (CountVectorizor and TFIDF) with multiple classifiers.	I chose f1 score as a measurement because it's is the harmonic mean of precision and recall and is a better measure than accuracy.

Machine Learning Model (Classifiers)
Model Evaluation

	Cleaned	Model	F1 Mean	F1 Std	Time
0	Yes	KNeighborsClassifier(n_neighbors=3)	0.730401	0.016725	8.675263
1	Yes	RandomForestClassifier()	0.838274	0.013113	105.164559
2	Yes	XGBClassifier(base_score=None, booster=None, c...	0.834534	0.007772	187.428415
3	Yes	AdaBoostClassifier()	0.792533	0.013028	114.361226
4	Yes	GradientBoostingClassifier()	0.812637	0.010069	297.003249
5	Yes	LogisticRegression()	0.873159	0.010611	4.780288
6	Yes	MultinomialNB()	0.859104	0.008988	0.275744

BERT

1. Subset dataset and split into training, testing
2. Encoding
3. Tokenization
4. Model evaluation with F1 score and accuracy