4/26/2020

Analyzing Amazon Kindle Store reviews

Team Members:

Kalyan Kumar Alisetty (A20199542)

Vinil Rayala (A20220246)

Sanjoy Kundu (A20263212)

Project Summary:

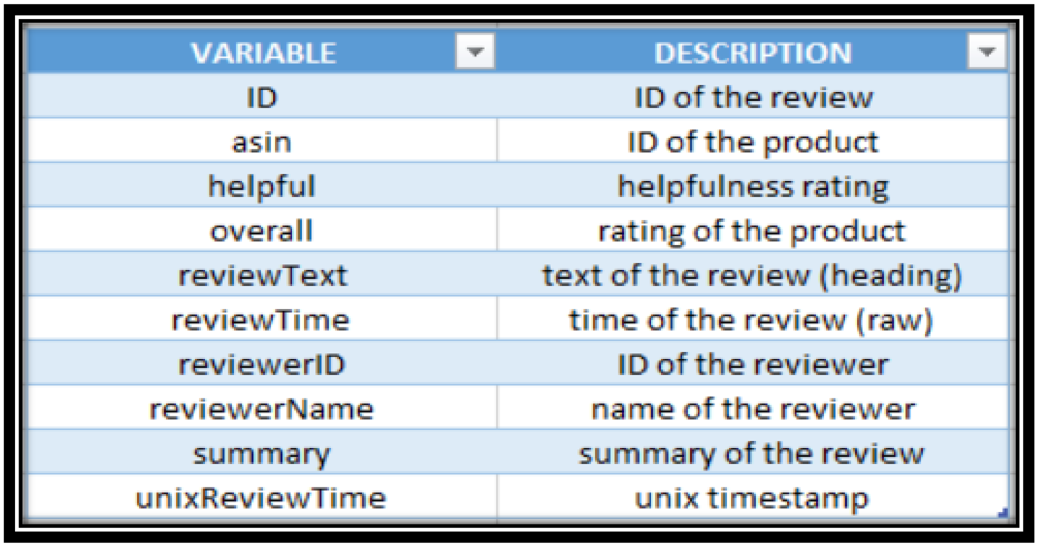
Leveraging unstructured data has gained a lot of attention these days due to its significant importance in giving a competitive advantage to the companies. Imagine how much time consuming it is for a company to manually go through millions of user reviews to understand what user likes or dislikes about a service. Our main goal is to effectively use text analytics to understand the user reviews on books and automate the sentiment classification of a review.

The main intent of this project is to use PySpark (specifically SQL data frames) and Google cloud cluster for performing topic modelling of large-scale data and build machine learning models for sentiment classification. In this process, we will also explore available NLP libraries and build NLP pipelines in Spark. We will perform a comparative study of the performance of different machine learning techniques with different feature extraction methods in case of sentiment classification.

Dataset:

https://www.kaggle.com/bharadwaj6/kindle-reviews

Dataset consists of product reviews from the Amazon Kindle Store category from May 1996 - July 2018. For this study, we have considered reviews for the year 2018 only. It consists of around 212413 records. Each reviewer has at least 5 reviews and each product has at least 5 reviews in this dataset.



VARIABLE	DESCRIPTION
ID	ID of the review
asin	ID of the product
helpful	helpfulness rating
overall	rating of the product
reviewText	text of the review (heading)
reviewTime	time of the review (raw)
reviewerID	ID of the reviewer
reviewerName	name of the reviewer
summary	summary of the review
unixReviewTime	unix timestamp

Fig 1: Variables available for analysis

There are 10 variables in the dataset. The “overall” variable has the rating of the review. This is used to create a new column “sentiment” which is binary. The reviews with rating 4 - 5 are considered as positive. The reviews with rating 1-3 are considered as negative. The “reviewText” variable has the review text data. The “summary” variable contains a short summary of the review.

These variables are important for performing topic modelling across different sentiments and building sentiment classification models.

Below are the visualizations illustrating the distribution of data across different sentiments and ratings.

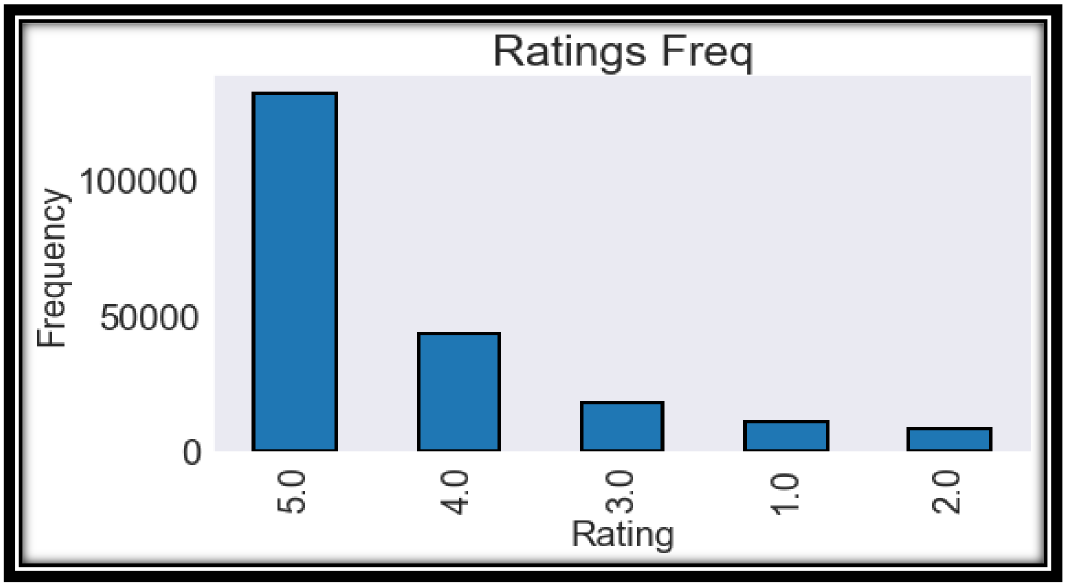


Fig 2: Number of reviews with each rating

The majority of the comments have rating 5.

We have created a new variable named "sentiment" which is positive if the rating is greater than 3

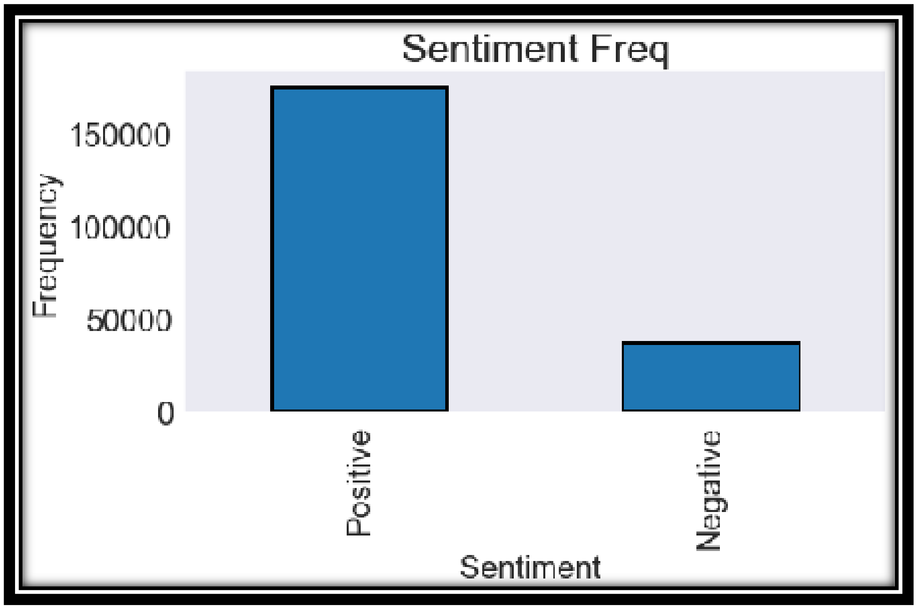


Fig 3: Number of reviews with each sentiment

There are more positive comments in the year 2018 compared to negative ones.

Following are the word clouds for the reviews across different sentiments.

Positive:

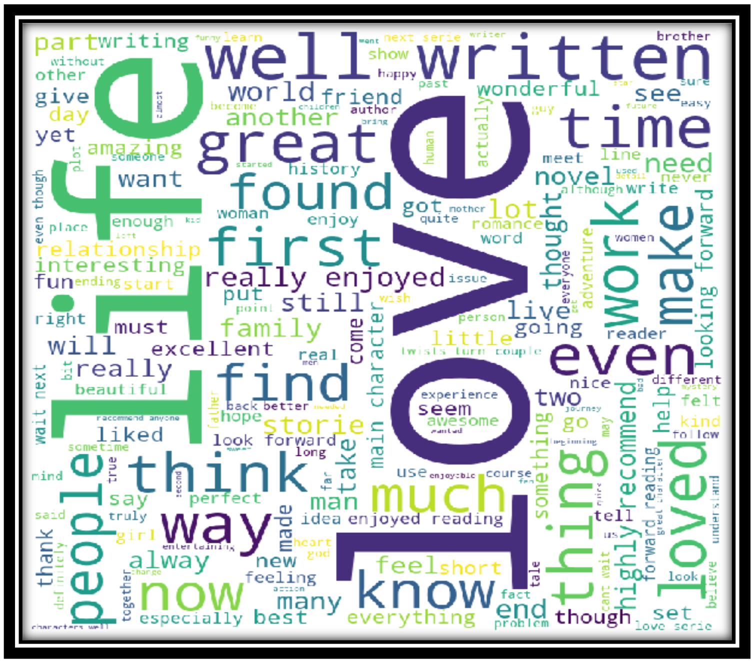


Fig 4: Word Cloud of Positive words

Negative:

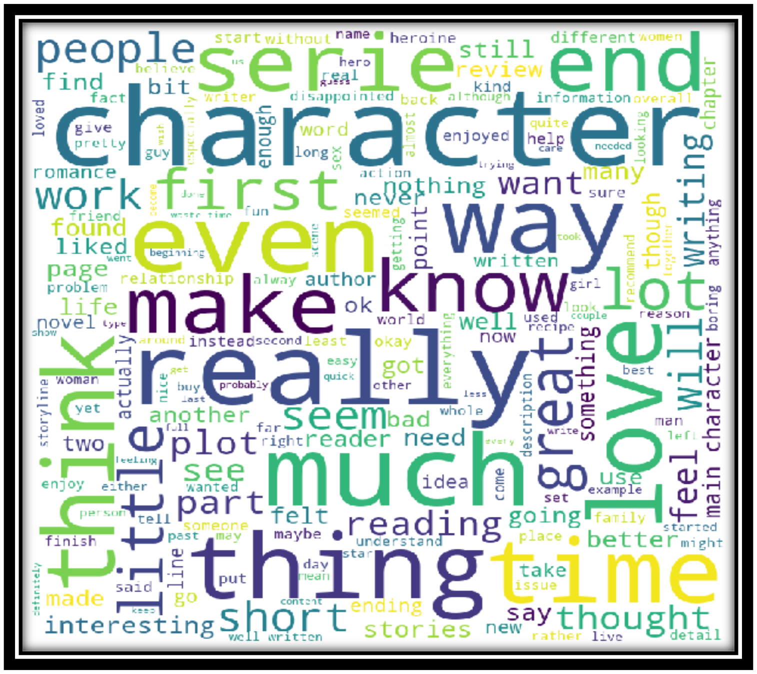


Fig 5: Word Cloud of Negative words

These word clouds illustrate that words such as love, great, life have frequently appeared in positive comments and words like characters, end appeared frequently in negative comments. Some overlap between them is expected as sometimes people discuss both negative and positive things in a single review.

Methodologies:

Topic Modelling

Topic modelling is an unsupervised machine learning technique that's capable of scanning a set of documents, detecting word and phrase patterns within them, and automatically clustering word groups and similar expressions that best characterize a set of documents. As training is not required for this, it can be used to quickly analyse the data. The core of this technique involves grouping similar word phrases and counting words within unstructured data for inferring the topics. By detecting patterns such as word frequency and distance between words, a topic model clusters feedback that is similar, and words and expressions that appear most often.

There are multiple algorithms to perform topic modelling. We are using the Latent Dirichlet Allocation (LDA) technique for our analysis.

Latent Dirichlet Allocation (LDA) is based on the assumptions of distributional hypothesis, (i.e. similar topics make use of similar words) and the statistical mixture hypothesis (i.e. documents talk about several topics) for which a statistical distribution can be determined. The purpose of LDA is mapping each document in our corpus to a set of topics that covers a good deal of the words in the document. LDA assumes that the distribution of topics in a document and the distribution of words in topics are Dirichlet distributions.

There are two hyperparameters that control topics frequency in a document and word frequency in a topic known as Alpha and Beta, respectively. A small value of alpha will assign fewer topics to each document and a small value of beta will assign fewer words to each topic.

Topic modelling using LDA is performed on the data across different sentiments to understand the topics so that feedback can be used in making informed decisions about the catalogue of books selected for the kindle delivery.

Sentiment Classification

Sentiment classification is the task of looking at a piece of text and identify if someone likes or dislikes it. Sentiment classification is important in categorizing the text into positive or sentiment without manual input for analyzing the potential issues in each segment. We are performing sentiment classification using various machine learning methods by training them on the labelled data.

We need to perform data pre-processing and feature extraction before topic modelling and sentiment classification.

There are four main steps in data pre-processing which are tokenization, stop word removal, POS tagging, lemmatization. Tokenization refers to the process of breaking down text documents into tokens which are just words in our case. We also performed data cleaning and removed punctuations, numbers, hyperlinks, etc., in this phase. Stop words removal is the process of removing words that are not useful in understanding the context of the text. We have used stop words from the NLTK library along with words like book, kindle, etc. which are frequently occurring without any useful information in understanding the context of the review. Part of Speech tagging (POS tagging) is the process of determining the part of speech of every token in a document, and then tagging it as such. This is important in identifying the verbs and nouns for performing stemming and lemmatization. Lemmatization usually refers to doing things properly with the use of vocabulary and morphological analysis of words, normally aiming to remove inflectional endings only and to return the base or dictionary form of a word, which is known as the lemma. Information from POS tagging is used in this case for performing lemmatization.

The next process after data pre-processing is to perform feature extraction. There are different ways to perform feature extraction. We have used different machine learning techniques combined with various feature extraction techniques for our analysis. Following are the various feature extraction techniques we have used in our analysis

***Frequency* –** Bag of Words model is extended in this case to represent the frequency of words in documents instead of just presence or absence for better feature extraction.

***TFIDF* -** Bag of Words model is extended in this case to represent the TFIDF value of words in a document instead of just presence or absence for better feature extraction. The TFIDF value increases proportionally to the number of times a word appears in the document and is offset by the number of documents in the corpus that contain the word. This helps in adjusting for the fact that some words appear more frequently in general.

We have used Logistic regression, Support Vector Machine (SVM), Naïve Bayes machine learning techniques in combination with the above-mentioned feature extraction techniques for sentiment classification and performed a comparative study.

The NLP Pipeline is shown in the below process flow chart.

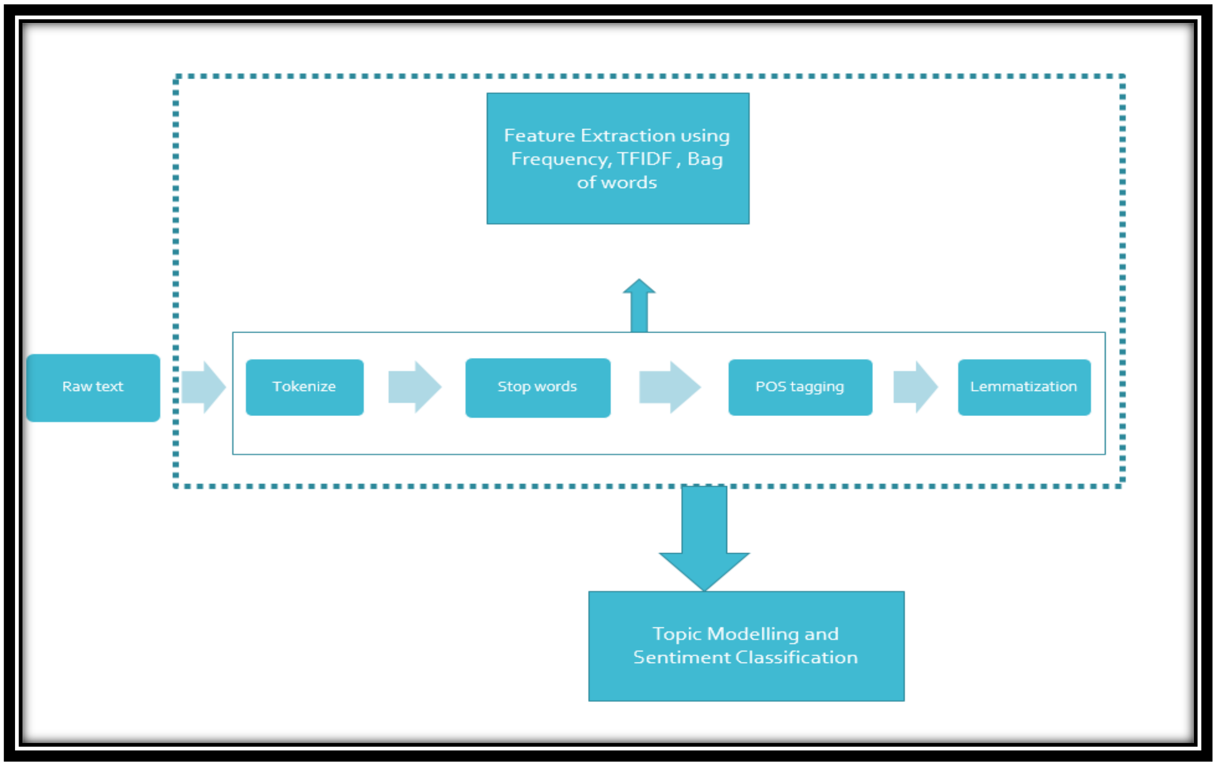

Feature Extraction using Frequency, TFIDF , Bag of words
Raw text
Tokenize
Stop words
POS tagging
Lemmatization
Topic Modelling and Sentiment Classification

Fig 6: Process Flow Pipeline

The Pipeline is built-in Spark using PySpark Data Frames and ML packages. LDA is performed using the PySpark ML clustering module. The entire models are run on google cloud clusters.

Setting up the Cluster:

In the Google cloud platform, we have used the Dataproc to create our cluster. We have set up a cluster in standard cluster mode with 1 Master node and 4 worker nodes. The master node consists of 4 CPU cores with 15GB memory and for the worker nodes, we configured it with 24 YARN cores and 48GB YARN memories. The image version we used is 1.5.2-debian10(Latest version of Hadoop and Pyspark available on the Google cloud platform). We also created a bucket in Google cloud storage to store the input data file and the output data file. We have executed our code with an SSH in the master node from VM instances.

Results:

We have chosen to move with around 25000 features from different feature extraction techniques. From the models trained on features extracted from different feature techniques, the Naïve Bayes model with count vectorizer feature extraction performed the best with an accuracy of around 87%.

The below graph shows the accuracy of different models built using different feature extraction techniques.

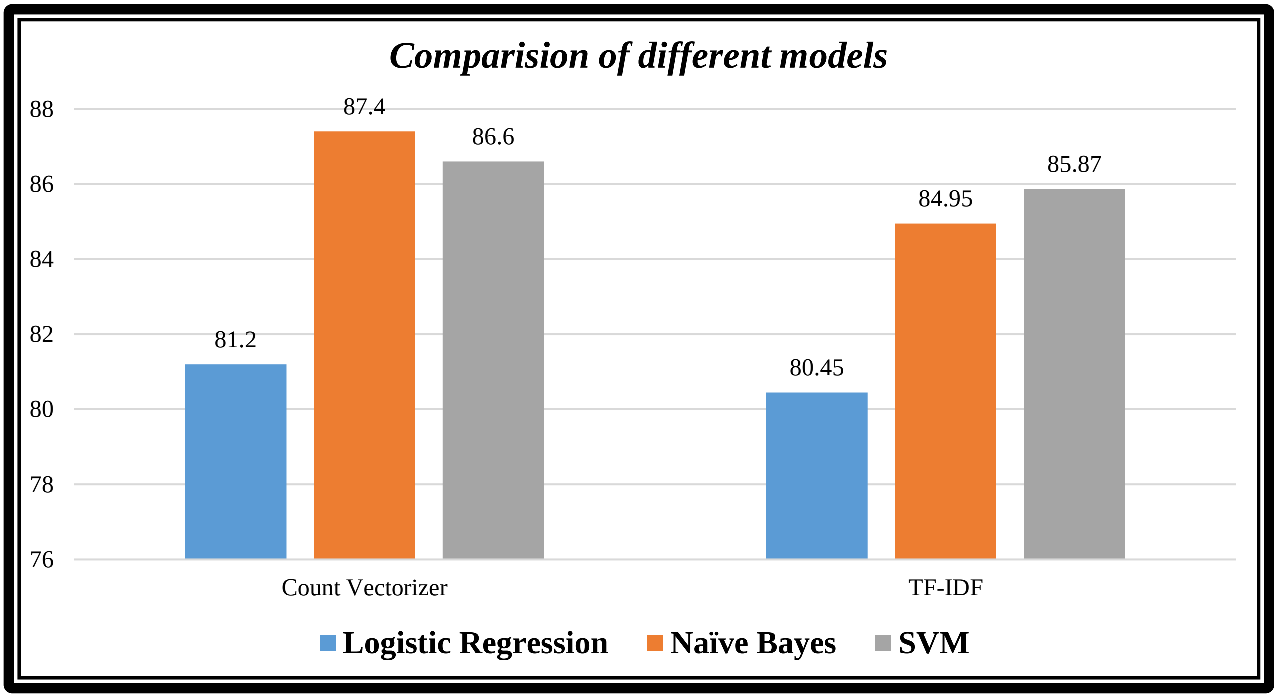
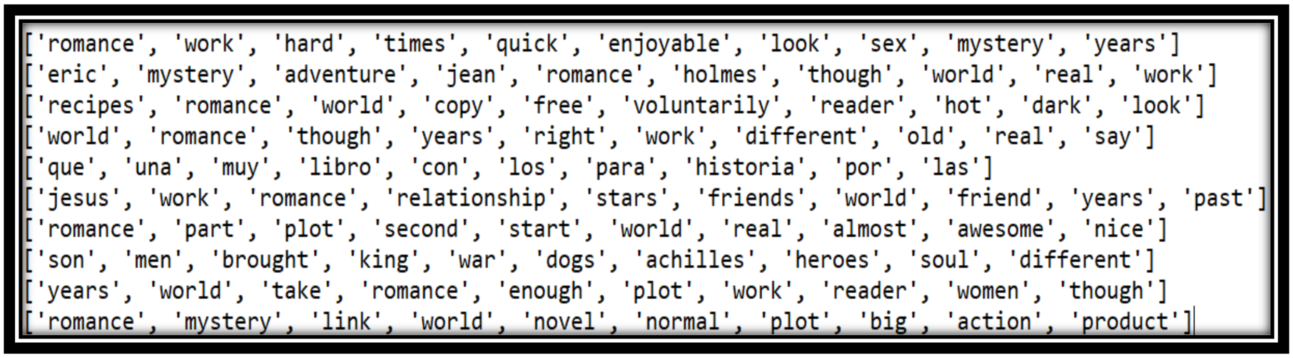


Fig 7: Comparison of accuracy of different models

From the total reviews of 2018, we have extracted the top and latest topics using LDA. The below screenshot shows the words in the top 10 topics in the order of importance of those words on that topic.

Positive:

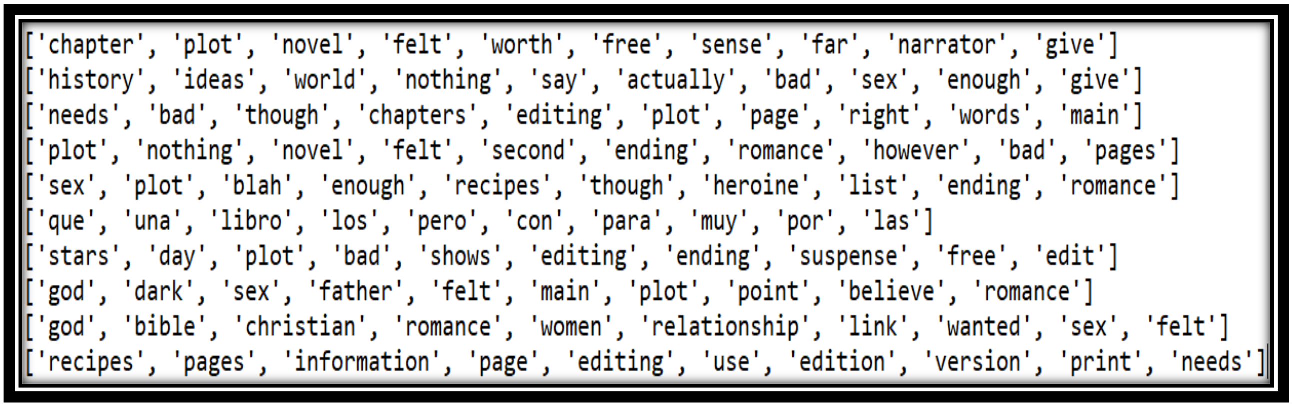


```
['romance', 'work', 'hard', 'times', 'quick', 'enjoyable', 'look', 'sex', 'mystery', 'years']
['eric', 'mystery', 'adventure', 'jean', 'romance', 'holmes', 'though', 'world', 'real', 'work']
['recipes', 'romance', 'world', 'copy', 'free', 'voluntarily', 'reader', 'hot', 'dark', 'look']
['world', 'romance', 'though', 'years', 'right', 'work', 'different', 'old', 'real', 'say']
['que', 'una', 'muy', 'libro', 'con', 'los', 'para', 'historia', 'por', 'las']
['jesus', 'work', 'romance', 'relationship', 'stars', 'friends', 'world', 'friend', 'years', 'past']
['romance', 'part', 'plot', 'second', 'start', 'world', 'real', 'almost', 'awesome', 'nice']
['son', 'men', 'brought', 'king', 'war', 'dogs', 'achilles', 'heroes', 'soul', 'different']
['years', 'world', 'take', 'romance', 'enough', 'plot', 'work', 'reader', 'women', 'though']
['romance', 'mystery', 'link', 'world', 'novel', 'normal', 'plot', 'big', 'action', 'product']
```

Fig 8: Top 10 topics from Positive reviews

The words from the positive topics suggest that people feel enjoyable about romantic books and adventure books. The words such as Jesus, bible suggest that the people feel more interested in the devotional books of god. And also it suggests that people like the plot and big action novel with mysteries in it. It also suggests that people like the book of Sherlock Holmes.

Negative:



```
['chapter', 'plot', 'novel', 'felt', 'worth', 'free', 'sense', 'far', 'narrator', 'give']
['history', 'ideas', 'world', 'nothing', 'say', 'actually', 'bad', 'sex', 'enough', 'give']
['needs', 'bad', 'though', 'chapters', 'editing', 'plot', 'page', 'right', 'words', 'main']
['plot', 'nothing', 'novel', 'felt', 'second', 'ending', 'romance', 'however', 'bad', 'pages']
['sex', 'plot', 'blah', 'enough', 'recipes', 'though', 'heroine', 'list', 'ending', 'romance']
['que', 'una', 'libro', 'los', 'pero', 'con', 'para', 'muy', 'por', 'las']
['stars', 'day', 'plot', 'bad', 'shows', 'editing', 'ending', 'suspense', 'free', 'edit']
['god', 'dark', 'sex', 'father', 'felt', 'main', 'plot', 'point', 'believe', 'romance']
['god', 'bible', 'christian', 'romance', 'women', 'relationship', 'link', 'wanted', 'sex', 'felt']
['recipes', 'pages', 'information', 'page', 'editing', 'use', 'edition', 'version', 'print', 'needs']
```

Fig 9: Top 10 topics from Negative reviews

The above words suggest the topics from the negative word cloud.

The words from the topic-1 suggest that people are not interested in buying the books instead they wanted to give the novels for free

The words from the topic-2 suggest that the ideas are bad and have nothing to say

The words from the topic-3 suggest that the editing is bad, and the right words are missing

Similarly, the words from the other topics suggest that people’s disinterest in different works such as editing, information, ending and also books on sex and suspense are more uninteresting for the people

Conclusion:

So, from the results, we can suggest that the Naïve Bayes model can be used for scoring the remaining or upcoming reviews for the Kindle Store and analyse for topics on it. As the current topic from 2018 suggest that people are more interested in the romantic and short adventurous stories or novels, so we can make more books available in these genres. As people's interests keep on changing, we must update our topics frequently and suggest the necessary books for them. It also suggests that the people are uninterested in the plot of the sex and suspense stories, So we can try to explain the stories more narratively and decrease the cost of these books to increase the sales. We can also focus our advertisements more on these types of books.

Future Work:

Word embedding can be used instead of a Bag of words approach to use the semantic structure of text for sentiment classification. We can also use other featuring algorithms to extract the features from the text. We can understand the words more in-depth and add some of them to the stop words list. We can also try some other deep learning modelling techniques such as CNN, DNN to increase the accuracy.

References:

1) https://monkeylearn.com/blog/introduction-to-topic-modeling/

2) https://spark.apache.org/docs/latest/ml-features.html