IJCRT.ORG ISSN : 2320-2882

INTERNATIONAL JOURNAL OF CREATIVE RESEARCH THOUGHTS (IJCRT)
An International Open Access, Peer-reviewed, Refereed Journal

SMART URT A NLP FRAMEWORK

[1]B. Upendra Varma, [2]DR. Rajesh T.M
M. Tech Student, Assistant Professor
Department of Computer Science and Engineering
Dayanand Sagar University, Bangalore, India

Abstract: Unstructured data remains to be a challenge in every data intensive application domains like business, research and technology driven companies and they are in the form of tweets, news, emails, reviews etc. Using text analytics we can extract meanings, patterns, and structure hidden inside unstructured text data. The term “text analytics” is an integrated framework by using techniques from data mining, machine learning, natural language processing (NLP). Text analytics uses techniques and methods that are used to get insights from unstructured data. These techniques can be broadly classified as topic extraction or modelling, cluster analysis which is a part of exploratory data analytics and sentiment analysis also called text classification which is a part of predictive text mining which is also called machine learning. To apply any kind of text mining technique like clustering or classification on text it has to be first pre-processed and need to be converted to a bag of words model so that we can extract features form the pre-processed data where we convert documents into vectorized format using word frequency count which has values in binary form so the machine(computer) can understand. SMART URT extends for Smart Unified extraction of sentiments from reviews and topics from documents which is a natural language processing unified framework in which I built a topic extraction model using text cluster analytics and also built a sentiment analyser which predicts if a movie review is positive negative or neutral.

The organization of the paper is as follows, section 1 gives the introduction of SMART URT, section 2 presents a brief Literature review on all the topics of the paper. Methodology of building our framework with explanation is presented in section 3. The next section 4 concerns Verification and validation of the machine learning models built by us along with the EDA using plots and evaluation metrics. Conclusion & future work is followed in section 5 and last but not least the reference.

Keywords: Text Analytics; Text Mining; Machine Learning; topic extraction; clustering; classification; sentiment analysis; bag of words; pre-processing

1. Introduction

Smart URT is a natural language processing framework which processes interactions between computers and human which are in the form of natural language. As a part of building this framework I have used various tools of text analytics like named entity extraction and chunking to mention a few. I have used various text mining tools also called machine learning which gives computers the ability to learn itself without any programming separately. I have built various machine learning models like topic modelling which discovers topics in a corpus of documents, text cluster analysis which groups similar text data into various clusters, sentiment analysis which we can use to classify various emotions like positive negative and neutral. There are various classification techniques like logistic regression, k nearest neighbours, decision trees and random forests to mention a few but I have used Naive Bayes and Scalar Vector Machines because they give accurate predictions and fits well on text data when we compare with other machine learning models. I did apply K means clustering for clustering the text. To apply various mentioned data mining techniques we have to preprocess and extract features from our text documents and convert it into a format which our machine can understand, so I have used bag of word model(BOW) which converts text documents into vectors. After applying machine learning models on our text we have to do exploratory data analysis which is a way of analysing our data and giving a summary about characteristics of our dataset. I have used a technique called Word clouds which shows us most frequent words in our documents, in our project we can show the most frequent words in all the clusters, most frequent positive words and also most frequent negative words when we are building our sentiment analyser. We also visualized the models we have built using scatterplots by reducing our features which we get while feature extraction by a technique called dimensionality reduction. Finally we have evaluated the model performance using confusion matrix and optimized our model using k fold cross validation. We used Quora questions for clustering and topic modelling and IMDB reviews for sentiment analysis.

2. Literature Review

Zhang Wang et al [1] in this survey they found that the processing of digital information is an increasingly important topic of research. Text information is becoming particularly prevalent and makes certain changes to processing methods. In their paper they used a feature extraction method using singular value decomposition and principal component analysis for optimal information reconstruction. They first used singular value decomposition to find eigen values of the test set. This new test set may have redundant information and causes computational complexity. So, they used principal component analysis as a second stage of processing. This method reduces the amount of redundant information and the data dimensionality and banishes the correlation between each input variable. This greatly decreases the computational complexity of the feature extraction.

Mani Verma et al [2] in their survey discovered that as the no of documents on the web grows exponentially multi document summarization becomes more and more significant as it provides the main ideas in a document which was set in a short time. In their paper they presented an unsupervised centroid based document reconstruction framework using bag of words model. Their model selects summary sentences to make reconstruction error minimum between summary and the documents. They applied sentence selection and beam search to improve the model performance. They conducted many experiments and the results which came from two different data sets shows major performance gains with the state of art baseline technologies.

Y Li et al [3] took in their survey Mongolian based named entity dictionary and with the rise of vocabulary that dictionary has not updated in time. So, in their paper they came up with a named entity extraction method based on multiple features and conditional random fields combination. Here in their paper they first clustered the entites that never came in the repository using hierarchical clustering by building dendograms. Then they carried word recognition and changed the results based on feature vectors.

Marutho Hendra Wijaya et al [4] in his survey stated that information is one of the most must need things in humans lives as they are basically becoming more impatient when they search for information from internet. Users always tries to get the right answer right away at the spot with no effort at all. News headlines are used to categorize news types. This approximation made it very easy for us so we can choose the particular topic that we need. Similarly, a title is used to cluster news. So, they took online news sites titles as data set. They used TFIDF as preprocessing method for their documents, k means for clustering and elbow method to optimize no of clusters. Sum of squared error of each cluster is calculated and compared to optimize no of clusters in elbow method. They used purity for internal evaluation which separates ideal clusters out of clusters. From the elbow method they got 8 as optimal k value and 0.228 as sum of squared errors between 7th and 8th clusters and they got 0.514 as purity value. Thus, they used elbow method to optimize the clustering model.

Bian Chen et al [5] they compared vector space models and graph ranking models in their research. Latent Dirichlet Allocation model can find out latent topics in the text corpus and latent topics gets benefits because of using sentence ranking method to get a decent summary. So, in their paper based on LDA a new sentence ranking method is proposed. This method combines distribution of topics of each sentence with importance of topic of the corpus to calculate subsequent probability of sentence so that we can select sentences to extract a summary. Topic distribution of each sentence represents likelihood of a sentence that belongs to a particular topic and topic importance depicts extent that topic covers decent amount of corpus. Their method highlights latent topics and optimizes summarization. Their experiment results showed us the advantage of multi document summarization algorithm that they proposed in their paper.

Praveen and Pandey et al [6] in their paper stated that in the last few years use of social networking sites haven been increased tremendously. They generate humongous amounts of data. Millions of people express their views and opinions on a wide variety of topics via these social networking sites. In their paper they discussed extraction of sentiment from Twitter where user posts their views and opinions. They used Hadoop to process movie data set in Twitter which are in form of reviews and comments using Naïve Bayes for sentiment analysis.

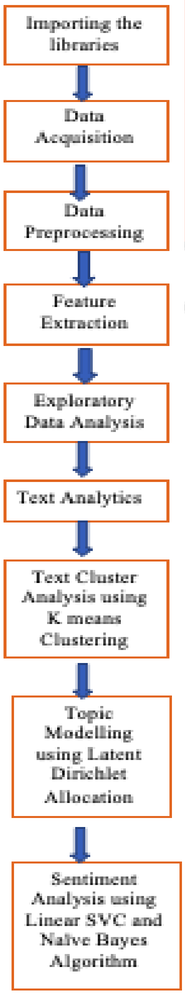
Rana et al [7] in their research stated that in the last few years sincere efforts were made for mining sentiments from natural language in messages, news and product reviews. In their paper they explored sentiment analysis considering positive and negative sentiments of film reviews. They built sentiment analyser using both have Naïve Bayes and Linear SVC and after experimentation the results that they got showed that Linear SVM gave best accuracy when they have done model selection.

3. Methodology

3.1. System Design

SMART URT is a nlp framework which contains basic techniques of text analytics like named entity extraction, text cluster analysis, topic modelling, sentiment analysis/ text classification. I took Quora questions for topic modelling and cluster analysis whereas IMDB reviews for sentiment analysis. I used K Means clustering algorithm for cluster analysis whereas Naïve Bayes, Linear SVC for sentiment analysis from which I will select model with best accuracy.

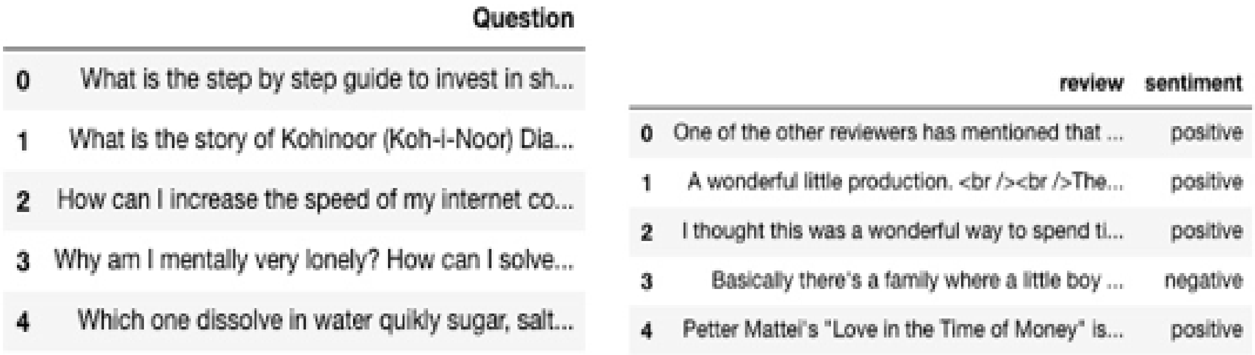
First, we start our project by importing all the necessary libraries of the language chosen, we have used python in our project and numpy, pandas, matplotlib and scikit learn are some of the libraries that we have used in our project. Then I have done data acquisition step where I have loaded my data sets in our case, I have used IMDB reviews for sentiment analysis and Quora questions for topic modelling and text cluster analysis. Then I have preprocessed my data by removing unnecessary punctuations and removed stop words and duplicates and missing values to name a few. Then I have done some text analytics on my data set like stemming, tokenization, named entity extraction and chunking to name a few. Then I have extracted features from my text documents using vectorizer which uses Bag of words model. I have done some basic exploratory data analysis like showing most frequent positive and negative words in IMDB reviews. I have extracted clusters of topics of Quora questions using k means clustering and allocated those topics to my documents using well known topic modelling technique called Linear Dirichlet Allocation. Then I have built sentiment analysers using Linear SVC and Naïve Bayes algorithms and then evaluated their accuracies and selected the model with the best accuracy and least loss/ error. I have used cross validation technique called k fold cross validation to tune hyper parameters of our trained machine learning models. I have evaluated models by building a confusion matrix which gives us various metrics like precision, accuracy, support and f1-score. At last I have plotted the results of clustered and classification algorithms using Principal Component Analysis which is the most useful dimensionality reduction technique.


Importing the libraries
Data Acquisition
Data Preprocessing
Feature Extraction
Exploratory Data Analysis
Text Analytics
Text Cluster Analysis using K means Clustering
Topic Modelling using Latent Dirichlet Allocation
Sentiment Analysis using Linear SVC and Naïve Bayes Algorithm

. Fig 3.1 System Design of Smart Urt

3.2. Data Acquisition

In our paper I have used Quora questions and IMDB reviews data sets which I got from well famous UCI repository. Quora questions where Quora is a repository where we can gain and share our knowledge. It's a platform where we can ask questions and connect with people who shares unique insights and real quality answers of whatever domain you want, which I will be using for clustering and topic modelling.
IMDB is the world's most popular resource for movie and television content. We can find ratings and reviews for the newest movie released and TV shows currently running, which I will be using for sentiment analysis using Naïve Bayes and Linear SVC has reviews as features and sentiment as labels.



	Question
0	What is the step by step guide to invest in sh...
1	What is the story of Kohinoor (Koh-i-Noor) Dia...
2	How can I increase the speed of my internet co...
3	Why am I mentally very lonely? How can I solve...
4	Which one dissolve in water quikly sugar, salt...

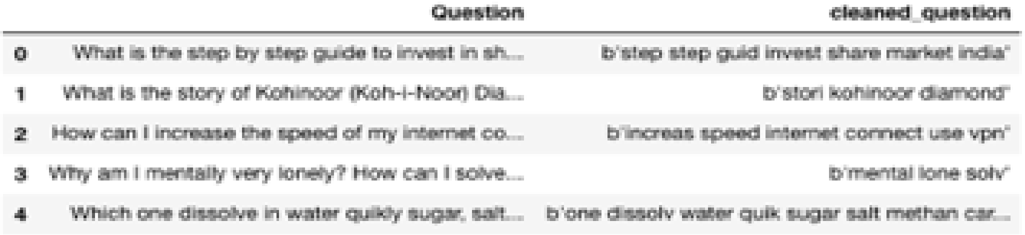
	review	sentiment
0	One of the other reviewers has mentioned that ...	positive
1	A wonderful little production. <br /><br />The...	positive
2	I thought this was a wonderful way to spend ti...	positive
3	Basically there's a family where a little boy ...	negative
4	Petter Mattei's "Love in the Time of Money" is...	positive

Fig 3.2 Data sets

3.3. Data Preprocessing

Text preprocessing is one of the important steps in handling natural language. It transforms text into a more understandable format so that machine learning algorithms can perform better. Below are the various preprocessing steps I have done in my project.

- Check for missing values if so remove them, they occur when no data is stored in that variable for an observation. Missing data can have a very bad effect on the conclusion drawn from the data.
- Remove stop words, Words like "a" and "the" appear so frequently that they don't need to be tagged as nouns and verbs. We call these stop words, and they can be removed from the text to keep our data cleaned.
- Remove html tags, If the reviews are web scraped there might be chances that they can have some HTML tags inside. So, it’s better to remove them as they are not useful for nlp tasks.
- Remove punctuations like question marks, quotations, hash tags, slashes etc.
- Word stemming using nltk’s port stemmer, it reduces a word to its stem or to its root form (stem of friendships is friendship) by removing the affixes of a word.
- Cleaned words should be alphabets only in lower case and having length > 2.
- Categorical labels like positive and negative should be encoded as 1 for positive and 0 for negative.
- Remove duplicate documents.



	Question	cleaned_question
0	What is the step by step guide to invest in sh...	b'step step guid invest share market india'
1	What is the story of Kohinoor (Koh-i-Noor) Dia...	b'stori kohinoor diamond'
2	How can I increase the speed of my internet co...	b'increas speed internet connect use vpn'
3	Why am I mentally very lonely? How can I solve...	b'mental lone solv'
4	Which one dissolve in water quikly sugar, salt...	b'one dissolv water quik sugar salt methan car...

Fig 3.3 Preprocessed data

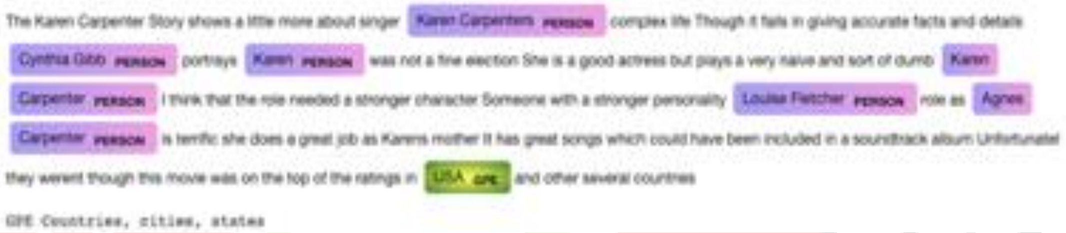
3.4. Text Analytics

With the help of text analytics a data analyst can extract meanings, hidden patterns, and structure lying within an unstructured text data. We can see text analytics as an integrated framework by using techniques from data mining, machine learning, natural language processing (NLP). Text analytics has been nowadays extensively used for preventing cybercrime and fraud detection. Hospitals are also providing patients better care by using text analytics.

Scientists in pharma industry are also using it to mine biomedical data to discover new kinds of drugs. It's basically an automated process of converting humongous volumes of unstructured text into measurable data to uncover insights, trends, and patterns. Natural Language Processing is the process of understanding text or speech by a machine. A resemblance is that humans communicate, understand each other perspectives, and respond with a correct answer. In NLP, all these processes are done by a computer rather than a human.

I have implemented various nlp techniques in SMART URT as of below,

- Parts of speech tagging, in English we have various pos like noun, pronoun, verb, adverb, preposition, adjective, conjunction etc.
- Derive noun chunks from text like autonomous cars, insurance liability etc.
- Named entity extraction, where entities are the words or group of words that represent information about common things such as persons, locations, nationalities, religious or political groups, buildings, airports, highways, bridges, organisations, companies, countries, states, cities, mountain ranges, bodies of water, vehicles, natural calamities, sports events, wars, books and songs titles, laws, language, dates, measurements etc. these entities have proper nouns. I have used spacy for doing this task.
- Get synonyms and antonyms of a word.
- Chunking where we draw a phrase as a parse tree of pos tags and groups of words in chunks using a regex pattern. I have used nltk for doing this task.
- Breaking documents into tokens.
- Collocations using bigrams(2 words) and trigrams(3 words).



The Karen Carpenter Story shows a little more about singer Karen Carpenters PERSON complex life Though it fails in giving accurate facts and details Cynthia Gibb PERSON portrays Karen PERSON was not a fine election She is a good actress but plays a very naive and sort of dumb Karen Carpenter PERSON I think that the role needed a stronger character Someone with a stronger personality Louise Fletcher PERSON role as Agnes Carpenter PERSON is terrific she does a great job as Karens mother It has great songs which could have been included in a soundtrack album Unfortunatel they werent though this movie was on the top of the ratings in USA GPE and other several countries

GPE Countries, cities, states

Fig 3.4 Named entity extraction

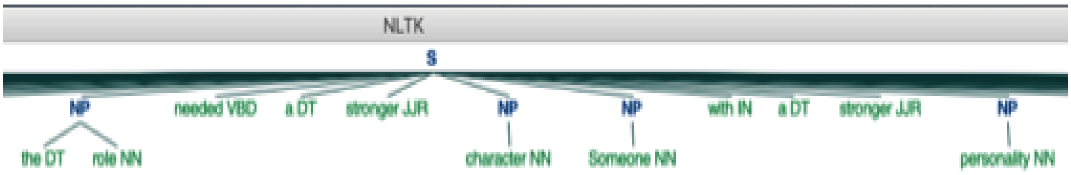


Fig 3.5 Chunking

3.5 Feature Extraction using BOW model

We need a method to convert text for our machine learning algorithm in a way that computers can understand and here comes bag-of-words model to the rescue which helps us to perform this task. It extracts features from the text to use it in any machine learning algorithm. we create tokenized words from documents and find out the frequency of each word and we call this process vectorization. We treat each sentence as a separate document, and we make a corpus of all words from all the documents by removing the punctuation and the next step is to convert to vectors which can be used by our machine learning model. For any machine learning model that works with text data we should use this model to extract features which will be basically in a sparse matrix on which a library like numpy can be used.

Different ways to convert text into vectors are:
1.) Count vectorizer: - Calculates frequency of the words which appears in a document by building a sparse matrix of documents x tokens
2.) Term Frequency - Inverse Document Frequency (tf- idf): - It basically evaluates how important a word is to a document in a collection of corpus of documents. The importance increases proportionately to the no of times a word appears in the document but is counterweighed by the frequency of the word in the corpus.

TF(t) = Number of times term t appears in a document / Total no of terms in the document (3.1)

IDF(t) = $\log_e$ (Total no of documents / No of documents with term t in it) (3.2)

3.6 Clustering Quora questions using k means

K-means clustering is the most popular unsupervised machine learning algorithms. Unsupervised algorithms get inferences from datasets using only input vectors without using any sort of labels. K-means groups similar data points together and discover hidden patterns. To achieve this, K-means looks to find out a fixed number (k) of clusters in a dataset. A cluster is a collection of observations accumulated together due to evident similarities unearthed by using several distance metrics like Euclidean distance. You'll define a target number k, which states the number of centroids you need in the dataset. A centroid is an imaginary location indicating the centre of the cluster. K-means algorithm identifies k centroids, and then allocates every observation to the nearest cluster and keeps the centroids as small as possible where points in different clusters are not similar whereas points in the same cluster are similar. A larger k creates smaller groups with more granularity which means they are distinguishable whereas a lower k means larger groups with less granularity.
K-means algorithm initiates with a first group of randomly selected centroids, which are used as the starting points for every cluster, and then performs calculations to optimize the positions of the centroids iteratively and stops creating clusters once there is no change in their values because the clustering has been done successfully or the number of iterations defined are completed.

Distance metrics used in K means Clustering model

We can find distance between any point and centroid by using a distance metric like,

- Minkowski.
- Euclidean.
- Manhattan.

$$\left(\sum_{i=1}^{n} \left|x_i - y_i\right|^p\right)^{\frac{1}{p}} \qquad (3.3)$$

Minkowski distance metric is the most general form of distance metric. We can change the value of p and calculate the distance in the following ways,
P=1, Manhattan Distance.
P=2, Euclidean Distance.

Choosing the right k value

Whatever k value we select, sum of squared distances of observations to their closest cluster centre should be minimum. We apply a method called elbow method to select the optimum k value where we plot k value vs inertia which tell us how far away the points are within a cluster. A good optimal clustering model is which having small value of inertia and small no of clusters given that value of inertia decreases as the no of clusters increases and such point in the plot, we have drawn is called elbow point.

Steps used in building a k means clustering model,

- Data acquisition of Quora dataset.
- Get the corpus of all text documents from csv file.
- Data cleaning or preprocessing as described in previous sections.
- Feature extraction using tf-idf as described in previous sections.
- Parameter tuning using elbow method to select optimal k value as described in previous sections, we got k as 5 using elbow method which means we can have 5 clusters and plot it using inertia vs k value.
- Train the model using k means.
- Exploratory Data Analysis, a method with which we can analyse data sets to get summary of their main attributes through visualization. Mainly EDA is used for seeing what the data can tell us after seeing the data beyond building a model and hypothetical testing and evaluating the built model.
- With top score we can draw word clouds which is used for representing text data in which the size of each word indicates its frequency or importance.
- Show the result of the clustering using dimensionality reduction (PCA) to plot clusters, centroids and outliers.

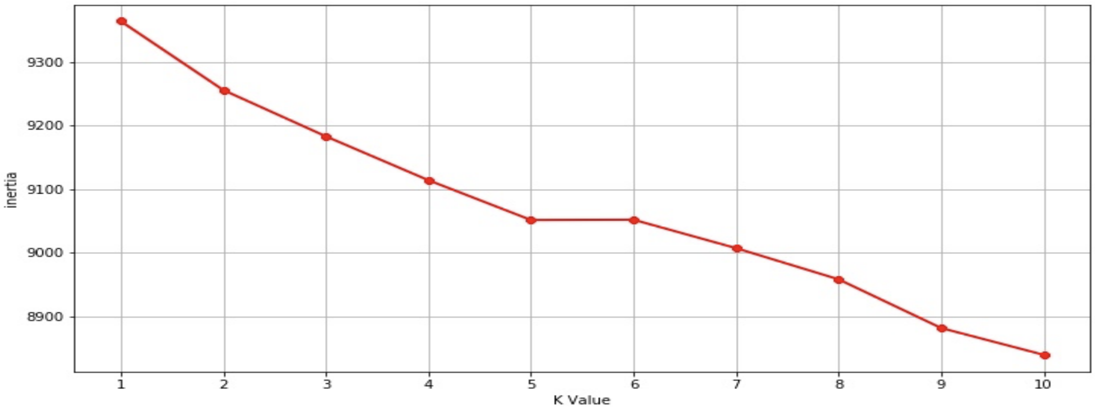


Fig 3.6 Elbow method

Below you can see that after using elbow method by tuning the parameter we get a k value of 5 which tells us that we will get an optimal clustering algorithm if we train it with 5 clusters here, we take k value as 5.

These are some important parameters in k means clustering model

- n_init which tells us no of times k means algorithm is run with different centroids.
- k-means++ selects initial cluster centres in a smart way to speed up convergence where the loss function reaches local minima, here our loss function is inertia.
- max_iter gives no of iterations of k means for a single run.
- n_clusters gives us no of centroids.
- tol, relative tolerance w.r.t difference in the cluster centres of two successive iterations.

Finally save the save the trained model in a .h5 file to avoid the load of the training time, so that it can be used anytime in the future by just loading them in order to reuse it to compare the model with other models and also to test the model on a new data. Saving data is called Serialization, while restoring the data is called Deserialization.

3.7 Topic Modelling Quora questions using LDA

Topic modelling is a part of unsupervised learning of natural language which is used to represent a text document with the help of several topics, that can be used to explain the hidden details in a particular document. This looks similar to clustering, but there is a small difference where instead of dealing with discrete features, let us say we have a collection of words that we want to group together in such a way that each group represents a topic in a document.

We can see a huge amount of textual data exists around us in an unstructured format in the form of news articles, social media posts and movie reviews etc. We need a way to understand, arrange and label this data to make proper decisions. Topic modelling is used in various applications like finding questions on stack overflow that are similar to each other, news topic documentation, recommender systems etc. All of these try to find the hidden structure in the text, as we know that every text that we write whether it's a tweet, post is composed of topics like sports, physics, aerospace etc.

Let's start by understanding what each word means in Latent Dirichlet Allocation,
Latent: - This refers to everything that we don't know in advance that we get independently without any experience and is hidden inside the data. Here, the topics that document consists of are unknown, but they are believed to be there from which the text is actually generated.
Dirichlet: - It's the distribution of topics in documents and distribution of words in the topic.
Allocation: - Now we allocate topics to documents and words of the document to topics.
So, in an overview what LDA means each word in each document comes from a topic and the topic is selected from a document.
we have two matrices:
1. Ѳtd = P(t|d) is the probability distribution of topics in documents.
2. Φwt = P(w|t) is the probability distribution of words in topics.
probability of a word given document,

$$P(w|d) = \sum_{t=1}^{T} p(w|t)\, p(t|d) \tag{3.4}$$

that is, we do dot product of 2 matrices for each topic t.
So, we divide the probability distribution matrix of word in document in two matrices which consists of distribution of topic in a document and distribution of words in a topic. We will allocate each document its respective topic from the topics and also words of the document to topics that we have extracted using k means clustering. We will use Latent Dirichlet Allocation to implement topic modelling. We have done various steps to implement LDA as below.

Steps used in building our lda model,

- Feature extraction using Count vectorizer by using document frequency thresholding, A term's document frequency is the number of documents in which the term occurs in the whole corpus. DF thresholding computes the document frequency for each unique term in the training corpus and then removes those terms whose document frequency are less than some predefined threshold. Which means basically, only the terms that occur many times are kept. Vocabulary reduction is very easy to do using this thresholding.
- Fit the model using no of clusters/components that we got above in k means clustering i.e. in our case 5.
- Transform the data to the model fitted.
- Allocate each document a topic.
- Save the trained model in a .h5 file.

3.8 Sentiment Analysis of IMDB reviews

Sentiment analysis is the classification of emotions (positive, negative and neutral) within text data with the help of text analytics. Sentiment analysis allows businesses to identify customer's sentiment towards products and brands during online interactions. Businesses today greatly depend on data. Majority of this data is unstructured text coming from sources like emails, chat messengers, social media, surveys, articles and documents. The social media content coming from Twitter and Facebook poses serious challenges due to the kind of the language used in them to express sentiments, memes and emojis. Examining this much of huge volumes of text data is very difficult and time taking. It requires a great amount of technical skill and resources to analyse all of that. Sentiment Analysis is also used by researchers, especially in fields like marketing, advertising, psychology, economics, and political science, which relies greatly on human-computer conversations.

Complications involved in sentiment analyser while understanding emotions through text is not easy at all. Sometimes even humans can be mis leaded, so expecting accuracy of 100% from a computer is almost impossible.

A text may contain multiple sentiments. Like,

1) "Movie is so so". The above sentence has 2 polarities, Positive and Negative. So how can we say if the sentiment of the review if it was Positive or Negative?

2)"The best I can say about the movie is it was interesting." Here, the word 'interesting' does not necessarily depict positive sentiment and is kind of confusing for a ml algorithm.
3)Heavy use of emojis with sentiment values in social media texts like that of Twitter and Facebook also makes text analysis very difficult. For example, a smiley :) generally refers to positive sentiment whereas :(indicates a negative sentiment. Also, acronyms like "LOL "," OMG" and generally used slangs like "Nah", "meh"," giggly" etc are also strong markers of some sort of sentiment in a sentence.

We will be building 2 models Naïve Bayes and Linear SVC for sentiment analysis using text classification and then we will select the best model with more accuracy.

3.9 Sentiment Analysis using Naïve Bayes

A classifier is a machine learning model which is used to distinguish different objects based on different features. A Naive Bayes classifier is a probability-based machine learning model which can be used for classification. The core of this classifier is Bayes theorem.

$$P(A|B) = (\ p(B|A) * p(A)\ )\ /\ p(B) \qquad (3.5)$$

Using Bayes theorem, we can find the probability of **A** happening, given that **B** has already happened. Here, **B** is called evidence and **A** is called hypothesis. The assumption made here is that the features are independent. That is presence of a particular feature don't really affect existence of another feature. hence it is called naive. Another assumption made here is that all the features have same kind of effect on the outcome and there is no change at all.

Let us suppose y is our target label and X = (x1, x2, x3, …., xn) is our feature vector in our data set, for ex let our y is playing golf and features X are outlook, temperature, humidity and windy when we apply naïve Bayes algorithm on this scenario using chain rule we get,

$$P\ (y|x_1 \ldots, x_n) \infty\ p(y)\ \Pi_n^{i=1}\ p\ (x_i\ |\ y) \qquad (3.6)$$

Above our classification is binary which means either yes or no, but if our model is multivariate then we have to find y with maximum probability,

$$Y = \max\ (p(y)\ \Pi_n^{i=1}\ p\ (x_i\ |\ y)) \qquad (3.7)$$

Multinomial Naïve Bayes is used in the applications of document classification, i.e. whether a document belongs to a category of sports, politics, technology etc. Here features are frequency of the words in a document. Naive Bayes algorithms are mostly used in sentiment analysis, spam filtering and recommendation systems etc. They are fast and very easy to implement but their biggest disadvantage is that features have to be independent. In most of the real-world scenarios, features are dependent, this hampers the performance of the classifier. First, we will build a text classifier using Naïve Bayes Algorithm for sentiment analysis where we take an IMDB reviews data set which has review and sentiment positive or negative which we take as features and labels and train our Naïve Bayes model on that data, then we will take unseen or new data/review and predict its sentiment, I have performed various steps to implement Naïve Bayes algorithm.

Steps used in building our Naïve Bayes algorithm,

- Data acquisition.
- Data Cleaning
- Get our X and y variables which are review and sentiment and split them into training and testing data with a respective ratio in our case we are taking test size as 30%.
- Exploratory data analysis where we show topmost positive and negative words in our reviews w.r.t frequency of these words using Word Cloud and polarity score which tell us how much positive or negative or neutral a word is which we get using Vader a nlp in built sentiment analyser.
- Feature extraction using tf-idf.
- Tune the hyper parameter alpha using k fold cross validation and plot alpha vs misclassification error.
- Train the model.
- Test the model or do predictions on our testing data.
- Check the result with nltk's built in sentiment analyser Vader to test the accuracy of our model and also check if our predictions are correct.
- Evaluate the model using metrics like accuracy, precision, recall, f1-score and display the confusion matrix using a heat map.
- Plot the result of classification of positive (1) and negative (0) labels in scatterplot using Principal component analysis.

Cross validation for parameter tuning of alpha value

cross validation is a resampling technique used to evaluate machine learning models on limited amount of data samples and it prevents overfitting. Cross-validation is primarily used in machine learning to estimate how a machine learning model works on new data. It generally results in a less biased estimate of the model than other methods, such as a simple train/test split where less bias means less assumptions about the target function made by a model to make the target function very easy to learn. Cross-validation is a technique in which we train our model using the subset of the data- set and then evaluate using the remaining subsets of the data set. In this method, we split the dataset into k number of subsets (also known as folds) then we perform training on all the subsets leaving one subset i.e. k-1 for evaluation of the trained model. In this method, we iterate k times with a different subset kept for testing each single time. We try to Smooth data set by creating an approximation function that captures all the important patterns in the data, while leaving out noise. when a certain word is not there in the corpus of documents a pseudo count is added in every probability estimate. So, no probability can ever be zero. This is a way of regularizing Naive Bayes where regularization is a technique which makes small modifications to the learning algorithm such that the model generalizes even better, so that model's performance on the new data is improved as well.

After parameter tuning using k fold cross validation, we get an alpha value of 1.

3.10 Sentiment Analysis using Linear SVC

It belongs to Support Vector Machine (SVM) which is a supervised learning algorithm and is a classifier which discriminates data points by strictly outlining using a separating hyperplane. This algorithm outputs an optimal hyperplane that categorizes new examples once we give labelled data. In two-dimensional space this hyperplane is just like a line separating a plane in two parts where each class rest in either of the side.
The classifier divides data observations using a hyperplane with the largest margin. SVM finds an optimal hyperplane with which we can classify new data points by using a kernel for handling nonlinear data. SVM basically finds a maximum marginal hyperplane that divides dataset into different classes.

Properties of a SVM

Support Vectors: - These are the data points which are very close to hyperplane which defines the separating line Hyperplane: - It's a decision plane which separates data points having different classes.

Margin: - It’s a gap between the two lines where closest class points lies. This is calculated using perpendicular distance between line and support vectors. If the margin is larger in between the classes, then it is considered a good margin, a smaller margin is considered a bad margin. Kernel: - It changes our input into higher dimensional space by converting non separable problem to separable problem by adding more dimension to it, so that you can easily segregate these points using linear separation. Decision function: - one vs rest divides the multi-class problem into multiple binary problems which involves training of a single classifier per class, with the samples of that class as positive and all other remaining samples as negatives.

Types of kernels

1)Linear: - We do dot product of any two given observations,
K (x, xi) = sum(x * xi) (3.8)

2)Polynomial: - Separates curved or nonlinear input area,

K(x, xi) = 1 + sum(x * xi)^d (d is the degree of the polynomial) (3.9)

3)Radial Basis function (exponential): - Maps an input area to infinite dimensional area,
K(x, xi) = exp(-gamma * sum((x – xi^2)) (3.10)

Hyper parameters in Scalar Vector Machine

These are parameters which can be altered which have to be tuned in order to obtain a model with optimal performance. In SVM’s we have the following hyper parameters which we have to fine tune to get an optimal model, 1) Regularization (c): - Tells the SVM how much you want to avoid misclassification of each training example. For large values of C, the optimization chooses a smaller- margin hyperplane even if that hyperplane classifies all the data points correctly and for a very small value of C causes the optimizer to look for a larger margin separating hyperplane, even if that hyperplane leads to misclassification of more data points in the entire population of the data. 2) Gamma (RBF): -It defines up to how extent the influence of a single training example goes, where low values suggest ‘far’, and high values indicates ‘close’. With low gamma, points far from hyperplane are considered during calculation. Whereas high gamma means the points close to hyperplane are also considered during calculation. Linear SVC is known to be the most optimal model for text classification as it’s really effective in higher dimension and also effective when no of features are more than training examples.

Steps involved in building our Linear SVC algorithm,

- Data acquisition.
- Data Cleaning
- Get our X and y variables which are review and sentiment and split them into training and testing data with a respective ratio in our case we are taking test size as 30%.
- Feature extraction using tf-idf.
- Tune the model regularization parameter (C) using grid search cross validation which does thorough search over specified parameter values for an estimator. Parameters of the estimator are optimized by cross validated grid search over a parameter grid, for ex when I have passed 0.001, 0.01, 0.1, 1, 10 as values for c in the parameter grid after doing grid search cross validation I got 1 as optimal value for fitting Linear SVC model for our data set.
- Tune the hyper parameter alpha using k fold cross validation and plot alpha vs misclassification error.
- Train the model.
- Test the model or do predictions on our testing data
- Check the result with nltk’s built in sentiment analyser Vader to test the accuracy of our model and also check if our predictions are correct.
- Evaluate the model using metrics like accuracy, precision, recall, f1-score and display the confusion matrix using a heat map and plot the results of classification of pos and neg with a scatter plot by using PCA

4. Verification and Validation

4.1 Exploratory Data Analysis of Clustered Quora Questions

Exploratory Data Analysis is the process of analysing data sets to give summary of their main properties through visualization. EDA is used for seeing what the data tells us beyond building machine learning model or hypothesis testing. Here in cluster analysis of Quora questions we have performed 2 visualizations.

Steps involved in exploratory Data Analysis of Clustered Quora Questions,
1) Viz top words in each cluster with top score using word clouds which is used for representing text data in which the size of each word indicates its frequency or importance,
2) plotted vertical bar charts showing most common words in each cluster using features/ words vs sum of squared distance (inertia score) from its cluster centre,

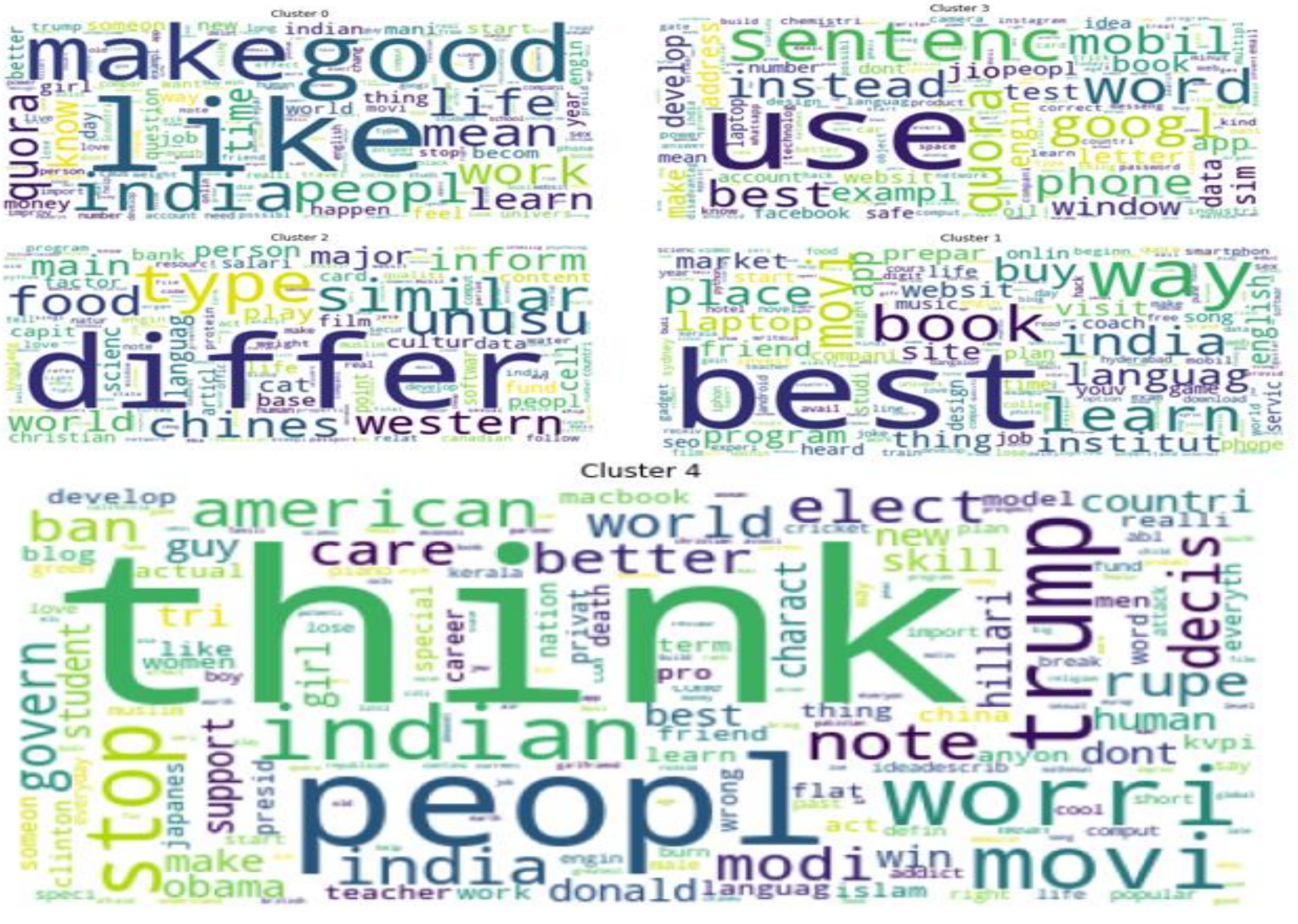


Fig 4.1 Word Clouds of the most frequent words in the 5 clusters created from our k means clustering model

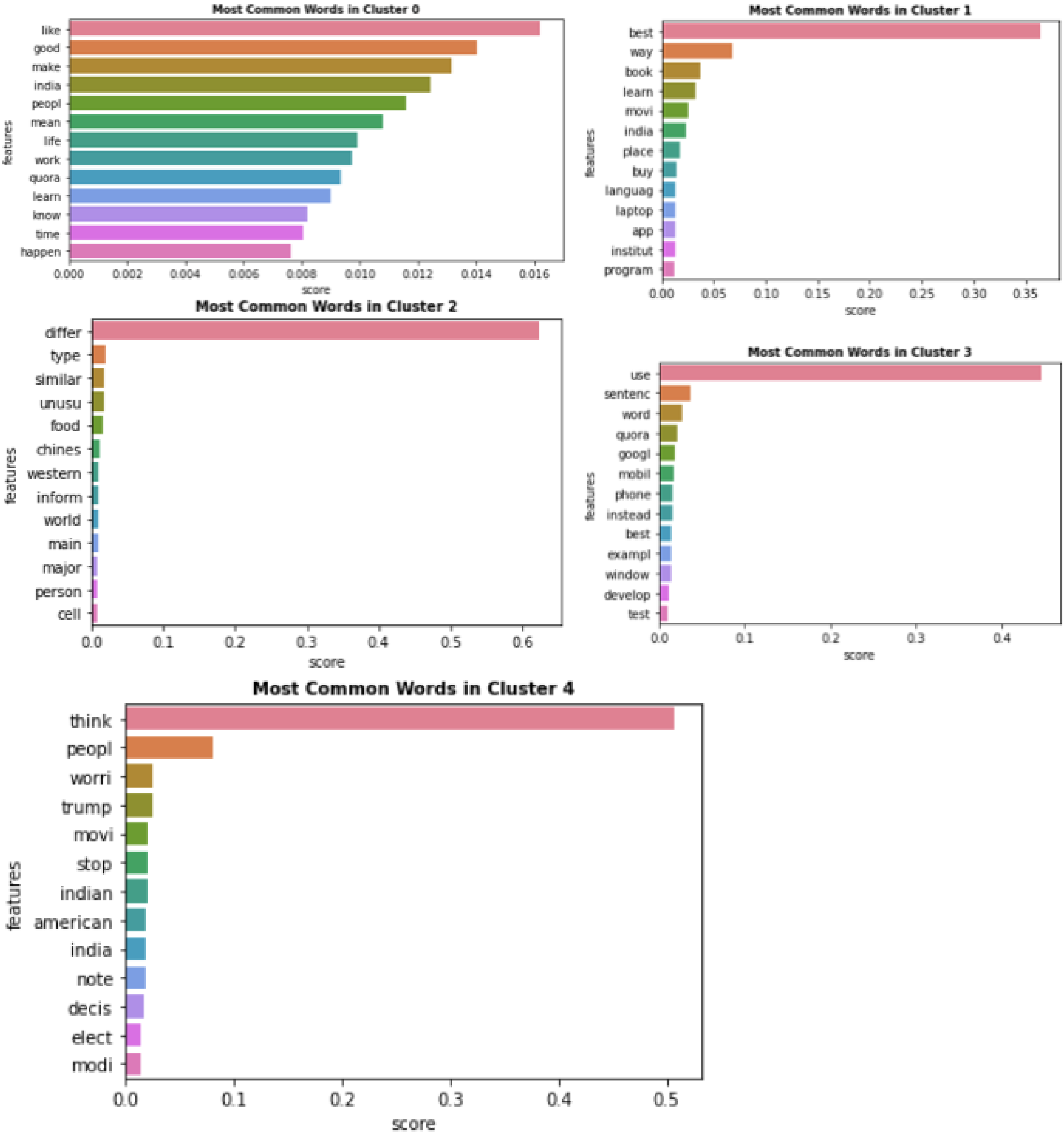


Fig 4.2 Bar Plots of the most frequent words in the 5 clusters created from our k means clustering model

From the above two visualizations we can summarize the 5 formed clusters as,

1. General life (emotions).
2. Best laptop/book/site.
3. Lifestyle.
4. Work.
5. Education

4.2 Cluster Analysis using Dimensionality Reduction

Dimensionality Reduction

Dimensionality reduction is a technique of reducing the number of input variables in a dataset. More input features often make a task of building a machine learning model more perplexing , which we generally refer to curse of dimensionality where increase in dimension causes overfitting and to avoid overfitting, the data has to be grown exponentially as you increase the number of dimensions, in large dimensional datasets, there tend be lots of inconsistent and redundant features in the dataset, which will increase the calculation time and make data processing and EDA more intricate. Therefore, it is often best to reduce the number of input features. This reduces the number of dimensions in our dataset, hence the name dimensionality reduction.

Principal Component Analysis

It's a technique of reducing the dimensionality of datasets, increasing variability but at the same time decreasing information loss by creating new non correlated variables that makes variance maximum. We get new variables from initial set of variables. The principal components are computed such that newly attained variables are hugely significant and independent of one another. The principal components get reduced and keeps most of the information that is useful and dispersed among the initial variables. If your data set consists of 4 dimensions, then 4 principal components are calculated, such that, the first principal component stores the maximum amount of information and the second one stores remaining maximum info and the process goes on like this.

Steps involved in finding principal components,
1)Standardisation of the data,
It scales your data such that all the values of the variables lie within a range, after scaling the distribution of data we should have mean 0 and variance 1.

2)Computing the covariance matrix,
It expresses the correlation between the different variables in the data set. It is necessary to find out highly dependent variables as they have biased and redundant information which reduces the total performance of the model, Negative covariance means variables are indirectly proportional to each other and positive covariance means variables are directly proportional to each other.

3)Calculating the eigenvectors and eigenvalues,
The idea behind calculating eigenvectors is to use the Covariance matrix to understand where our data has the most amount of variance. Since more variance in data means more information about the data, so basically eigenvectors do detect our Principal Components.

$$Av = \lambda v \text{ or } | A - \lambda I | = 0 \qquad (4.1)$$

A is covariance matrix, v is eigen vector and λ is eigen value and determinant. Eigen vectors transforms high dimension data to low dimension.

4)Computing the Principal Components,
Once we calculate the Eigenvectors and eigenvalues, sort them out in the descending order, where the eigenvector with the highest eigenvalue is the most significant and becomes the first principal component. The principal components of less significance have to be removed at any cost in order to reduce the dimensions of our data. Then, we create a matrix called feature matrix that has all the significant data variables which has maximum information about the data.

5)Reducing the dimensions of the data set,
Finally, we re-group the original data with the final principal components which we got that denotes the maximum and the most important information of the data set. In order to replace the original data with the newly formed Principal Components, multiply the transpose of the original data set by the transpose of the newly obtained feature vector.

In our cluster analysis we have almost 10,000 features which we got from feature extraction using tf-idf, now we need to reduce it to 2 dimensions using PCA and plot a scatter plot between these 2 principal components. In the below scatterplot we can see all the 5 clusters and centroids along with some outliers.

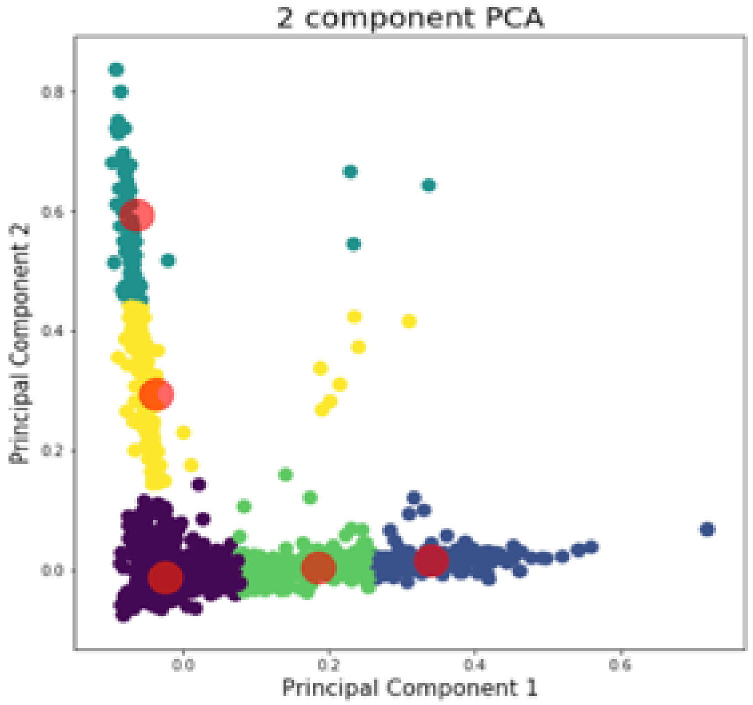
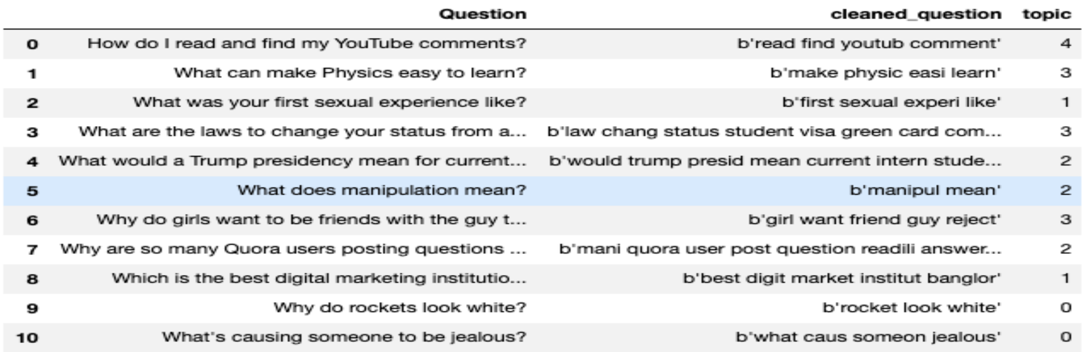


Fig 4.3 Scatterplot of clustered questions using PCA

4.3 Topics (Questions) Allocated using LDA

Here you can see the documents in our data set i.e. Quora questions being allocated their respective topics using LDA using a concept called topic modelling, In the below figure you can see there are 5 topics and each Question being allocated a topic.



	Question	cleaned_question	topic
0	How do I read and find my YouTube comments?	b'read find youtub comment'	4
1	What can make Physics easy to learn?	b'make physic easi learn'	3
2	What was your first sexual experience like?	b'first sexual experi like'	1
3	What are the laws to change your status from a...	b'law chang status student visa green card com...	3
4	What would a Trump presidency mean for current...	b'would trump presid mean current intern stude...	2
5	What does manipulation mean?	b'manipul mean'	2
6	Why do girls want to be friends with the guy t...	b'girl want friend guy reject'	3
7	Why are so many Quora users posting questions ...	b'mani quora user post question readili answer...	2
8	Which is the best digital marketing institutio...	b'best digit market institut banglor'	1
9	Why do rockets look white?	b'rocket look white'	0
10	What's causing someone to be jealous?	b'what caus someon jealous'	0

Fig 4.4 Topic Modelling of Quora Questions

4.4 Exploratory Data Analysis of IMDB Reviews using Polarity Score

Let us see what insights we can draw from our IMDB reviews using EDA, we need to perform the following steps for that,
Steps involved in exploratory data analysis of IMDB reviews,
1) Here first we use nltk's inbuilt sentiment analyser Vader to calculate Polarity score of each review which tells us how positive negative a review is, first we need to preprocess our data to apply Vader on our dataset. Where a polarity

score > 0 means positive and < 0 means negative.
2) we show topmost positive and negative words in our reviews w.r.t frequency of these words using Word Cloud and polarity score shown above and frequency bar charts.

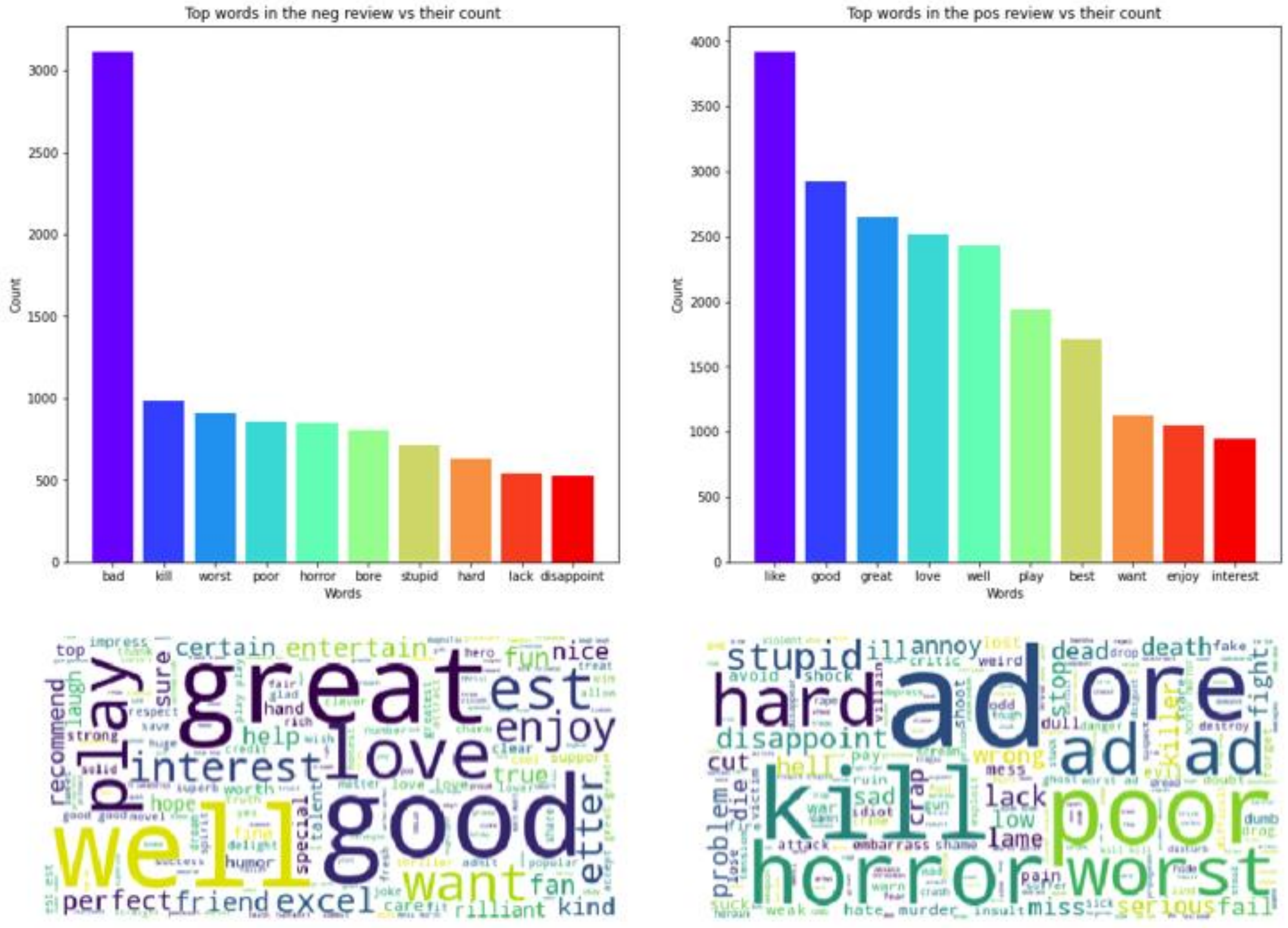


Fig 4.5 Bar plots & word clouds of most frequent positive and negative words in IMDB reviews data set

4.5 Sentiment Analyzer predictions of Naïve Bayes

After fitting the Naïve Bayes model now, we can predict sentiment of unseen reviews or we can say test data which we split in train test split process.
Let us see sentiment of an example review,
"Movie which I watched yesterday is so bad".
Prediction is 0, which means you can see our classifier is able to classify the sentiment accurately.
let us cross check the result with Vader nltk's sentiment analysis to make sure that our predictions are correct,
Vader's result is negative which is nothing but 0 which means our predictions are correct.

4.6 Evaluation Metrics of Naïve Bayes

Model Evaluation is the most important part of the model development. It helps finding the best model that represents our data and how well the selected model works in the impended future. We can't evaluate model performance with the training data in machine learning because it easily generalizes overfitted models. So, we have to use a test set to evaluate model performance.
The best way to evaluate is finding confusion matrix between actual and predicted data w.r.t both the classes i.e. negative and positive in our model.

Let us see the diagram of confusion matrix first to understand it,

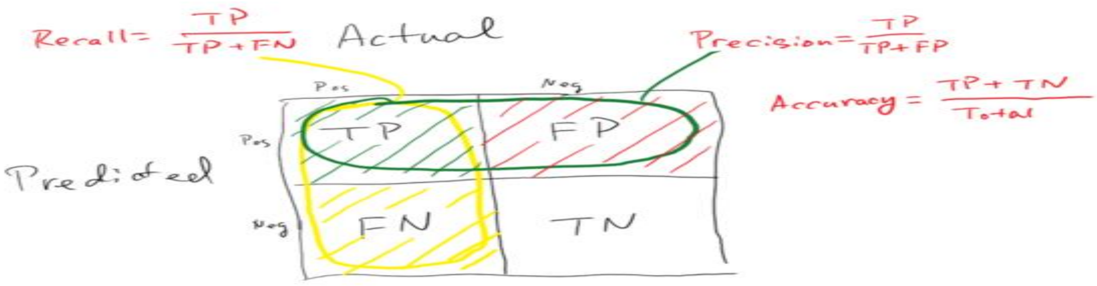


Fig 4.6 Confusion Matrix

Now let us understand each term,

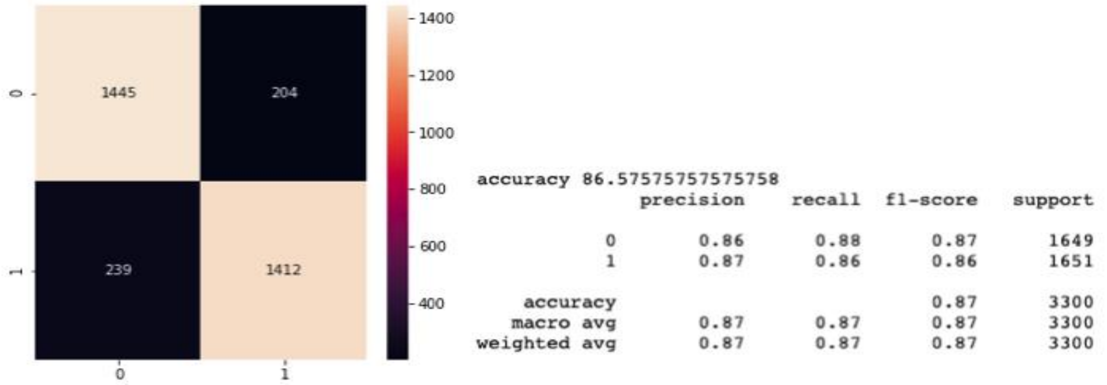
1. True Positive, predicted positive and are actually positive
2. False Positive, predicted positive and are actually negative.
3. True Negative, predicted negative and are actually negative.
4. False Negative, predicted negative and are actually positive.

Confusion matrix is just a representation of the above parameters in a matrix format now from this confusion matrix we can derive some evaluation metrics like,

1. Accuracy, out of all the classes, how much we predicted correctly.
2. Precision, out of all the positive classes we have predicted, how many are actually positive or percentage of positive examples out of the total predicted positive examples.
3. Recall/Sensitivity/True positive rate, out of all the positive classes, how much we predicted correctly. It should be high as possible, also can be defined as percentage of positive examples out of the total actual positive examples.
4. Specificity, Percentage of negative examples out of the total actual negative ex. TN / TN + FP.
5. F1 score, harmonic mean of precision and recall,

   (2 * precision * recall) / precision + recall (4.2)
6. Support gives us the no of elements in each class.

Let us find confusion matrix and apply these evaluation metrics on our Naïve Bayes model,



```
accuracy 86.57575757575758
              precision    recall  f1-score   support

           0       0.86      0.88      0.87      1649
           1       0.87      0.86      0.86      1651

    accuracy                           0.87      3300
   macro avg       0.87      0.87      0.87      3300
weighted avg       0.87      0.87      0.87      3300
```

Fig 4.7 Evaluation Metrics of Naïve Bayes classification

From the above figure we can see that accuracy is 86 % and precision for 0/positive is 86 % whereas 87 % for 1/negative and also, we can see recall and f1 score which is derived from precision and recall for both positive and negative classes.

4.7 Viz Naïve Bayes Classification using Dimensionality Reduction

Use PCA to reduce the no of features say 10,000 that we have extracted using tf-idf vectorizer up to 2 and plot a scatter plot between these 2 principal components w.r.t both the classes 0 and 1 i.e. positive and negative.

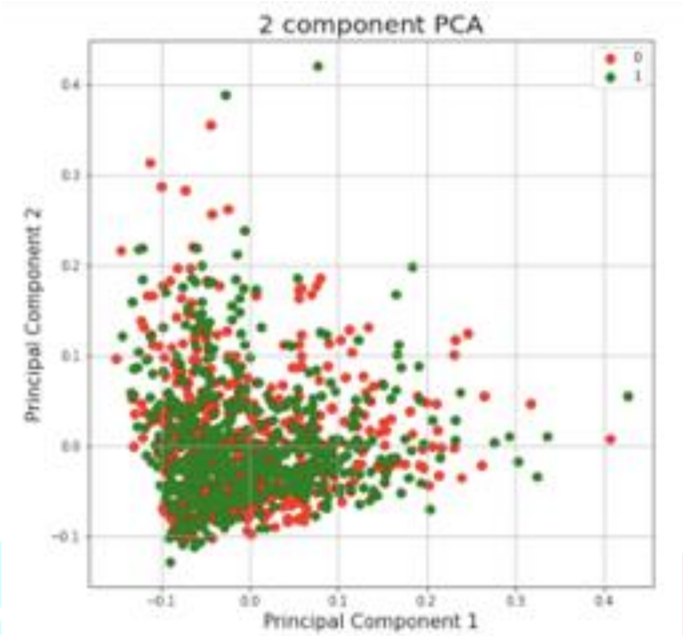


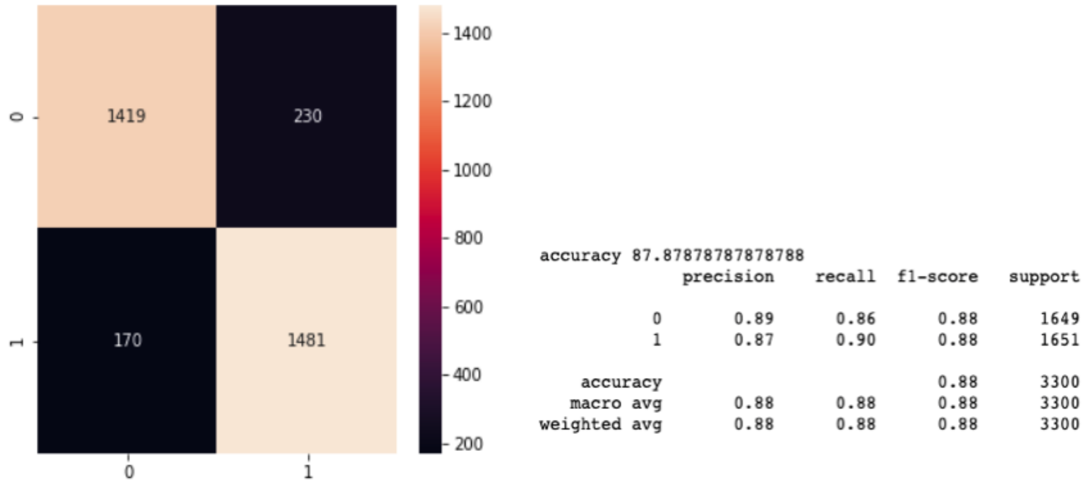
Fig 4.8 Scatterplot showing Naïve Bayes classification using PCA

4.8 Sentiment Analyzer Predictions of Linear SVC

After fitting the Linear SVC model now, we can predict sentiment of unseen reviews or we can say test data which we split in train test split process.
Let us see sentiment of an example review,
“Movie which I watched yesterday is so bad”.
Prediction is 0, which means you can see our classifier is able to classify the sentiment accurately.
We already seen prediction of same review with Naïve Bayes and we got 0 for that also and we have cross checked with Vader which also gave negative, so we can say that our model is predicting accurately based on these evidences.

4.9 Evaluation metrics of Linear SVC model

Let us find confusion matrix and derive evaluation metrics from it for our fitted Linear SVC model,



```
accuracy 87.87878787878788
              precision    recall  f1-score   support

           0       0.89      0.86      0.88      1649
           1       0.87      0.90      0.88      1651

    accuracy                           0.88      3300
   macro avg       0.88      0.88      0.88      3300
weighted avg       0.88      0.88      0.88      3300
```

Fig 4.9 Evaluation Metrics of Linear SVC classification

From the above figure we can see that accuracy is 87 % and precision for 0/positive is 89 % whereas 87 % for 1/negative and also, we can see recall and f1 score which is derived from precision and recall for both positive and negative classes.

By looking at accuracies of both Naïve Bayes and Linear SVC we can say that Linear SVC gives better or to say more accurate predictions.

4.10 Viz Linear SVC Classification using Dimensionality Reduction

Use PCA to reduce the no of features say 10,000 that we have extracted using tf-idf vectorizer up to 2 and plot a scatter plot between these 2 principal components w.r.t both the classes 0 and 1 i.e. positive and negative.

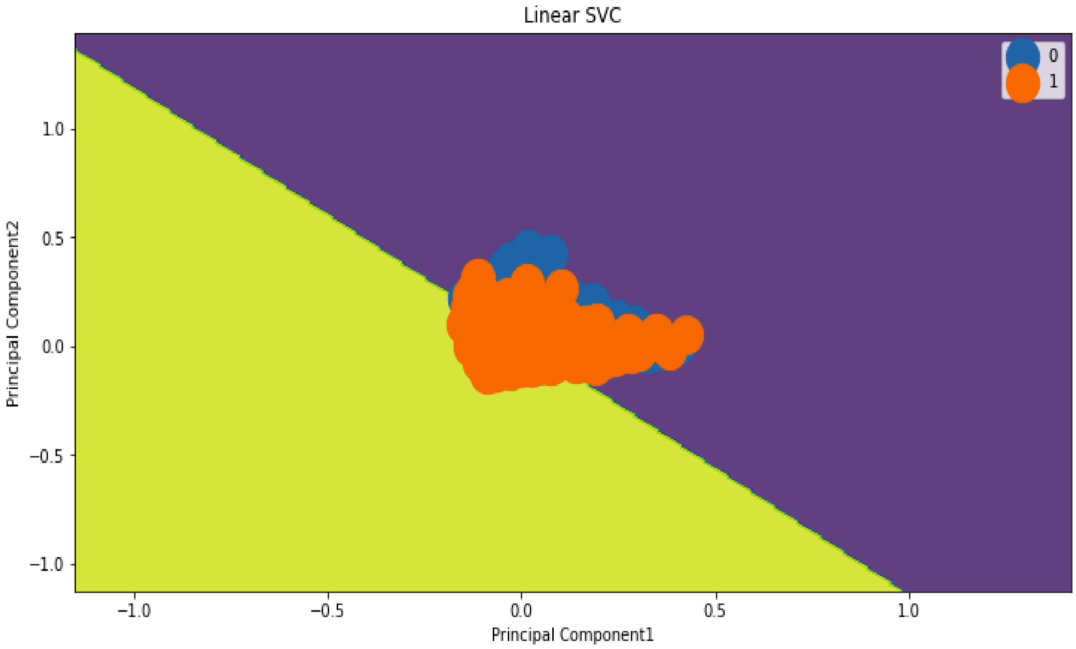


Fig 4.10 Scatterplot showing Linear SVC classification using PCA

From the above plot we can see that our kernel is linear and our C regularization parameter value 1 tells us that we choose a smaller-margin hyperplane and the hyperplane did a really good job of classifying data points correctly.

5. Summary and Conclusion

5.1 Summary

Let us see summary of tasks that I did as a part of implementing this framework of natural language processing,

1. I incorporated some text analytics like NER, POS and chunking.
2. I did some text cluster analytics using K Means clustering on Quora questions, I have clustered similar questions and I got 5 clusters,
   a. General life (emotions).
   b. Best (laptop, book, site).
   c. Lifestyle.
   d. Work.
   e. Education.
3. I allocated the extracted topics above to Quora questions using topic modelling technique called Latent Dirichlet Allocation.
4. I implemented sentiment analysis using IMDB review using Naïve Bayes Classifier and Linear Scalar Vector Classifier,
   a. I got 86% accuracy for Naïve Bayes model.
   b. I got 87% accuracy for Linear SVC so selected as final model for sentiment analysis.

5. I did feature extraction using term frequency inverse document frequency.
6. I have applied all the important steps to preprocess the text data like removing stop words, stemming removing missing values and duplicate values.
7. I have applied Exploratory Data Analysis using Word Clouds and Bar Plots.
8. I have applied dimensionality reduction technique Principal Component Analysis to show the result of Clustering analysis and Sentiment Analysis using Scatterplot.
9. I used Confusion Matrix to evaluate models using precision, recall, f1-score.
10. I have tested results of sentiment analysis using nltk's inbuilt sentiment analyser Vader and got accurate results.
11. I have saved the fitted models in .h5 file using serialization.

5.2 Conclusion

Including text in data analysis has changed the overall complexity of analytics over the last few years. You have seen how machine learning which learns hidden patterns in text are now replaced by much more sophisticated methods like text analytics by using NLP. Text mining is nowadays called text analytics which uses many techniques of preprocessing like bag of words model like vectorizers that we already mentioned.

Text Analytics is an integrated framework of tools and methods developed to retrieve, clean, analyse, and interpret that is get insights information from a broad range of data sources including big data. Various techniques have been developed, where each focus in answering a specific business problem based on text. Feature extraction, topic modelling, document classification, cluster analysis, information extraction, sentiment analysis, etc., are some of the techniques that we have dealt with in great detail in our project.

Finally, what I can say is that the framework which I have created using Quora questions and IMDB reviews can also be used as a baseline for text cluster analysis and text classification on other applications like modelling news articles and twitter sentiment analysis.

References

[1] aeama, J. Wang, J. Mou, X. Li and R. Wang, "Digital Text Feature Extraction Using Singular Value Decomposition and Principal Component Analysis," 2019 2nd International Conference on Information Systems and Computer Aided Education (ICISCAE), Dalian, China, 2019, pp. 10-13, doi: 10.1109 ICISCAE48440.2019.221578.

[2] K. Mani, I. Verma, H. Meisheri and L. Dey, "Multi-Document Summarization Using Distributed Bag-of-Words Model," 2018 IEEE/WIC/ACM International Conference on Web Intelligence (WI), Santiago, 2018, pp. 672-675, doi: 10.1109/WI.2018.00-14.

[3] Y. Li, "Named Entity Relation Extraction Based on Multiple Features," 2015 IEEE 29th International Conference on Advanced Information Networking and Applications Workshops, Gwangiu, 2015, pp. 213-216, doi: 10.1109/WAINA.2015.14.

[4] D. Marutho, S. Hendra Handaka, E. Wijaya and Muljono, "The Determination of Cluster Number at k-Mean Using Elbow Method and Purity Evaluation on Headline News," 2018 International Seminar on Application for Technology of Information and Communication, Semarang, 2018, pp.533-538, doi: 10.1109/ISEMANTIC.2018.8549751.

[5] J. Bian, Z. Jiang and Q. Chen, "Research on Multi-document Summarization Based on LDA Topic Model," 2014 Sixth International Conference on Intelligent Human-Machine Systems and Cybernetics, Hangzhou, 2014, pp. 113-116, doi: 10.1109/IHMSC.2014.130.

[6] H. Parveen and S. Pandey, "Sentiment analysis on Twitter Data-set using Naive Bayes algorithm," 2016 2nd International Conference on Applied and Theoretical Computing and Communication Technology (iCATccT), Bangalore, 2016, pp. 416-419, doi: 10.1109/ICATCCT.2016.7912034.

[7] S. Rana and A. Singh, "Comparative analysis of sentiment orientation using SVM and Naive Bayes techniques," 2016 2nd International Conference on Next Generation Computing Technologies (NGCT), Dehradun, 2016, pp. 106-111, doi: 10.1109/NGCT.2016.7877399.