**Table of Contents**

|  |  |  |
| --- | --- | --- |
| **Sl. No** | **Chapter** | **Page Number** |
| **1** | **Executive Summary** | **4** |
| **2** | **Introduction**  **Domain**  **Problem Statement**  **Outline of the Report**  **Background**  **Sentiment Classification & Analysis**  **Sentiment Classification Using ML Methods**  **Methods of Algorithm used**  **Naïve Bayes & KNN**  **Convolutional Neural Network** | **5**  **5**  **6**  **6**  **6**  **6**  **7**  **9**  **11**  **12** |
| **3** | **Related Work** | **13** |
| **4** | **System Architecture**  **Import Modules & Collecting Dataset**  **Data Pre-Processing**  **Training the Model**  **Testing the Model** | **14**  **15**  **18**  **21**  **22** |
| **5** | **Results**  **Accuracy of Naïve Bayes**  **Accuracy of Logistics Regression**  **Accuracy of CNN** | **24**  **24**  **25**  **26** |
| **6** | **Conclusion & Future Work** | **28** |
| **7** | **Bibliography** | **29** |

**List of Figures**

|  |  |  |
| --- | --- | --- |
| **Sl. No** | **Figure Name** | **Page Number** |
| **1** | **Number of Paper Published on sentiment analysis over the past decade** | **7** |
| **2** | **Overall procedure diagram of the sentiment analysis of Amazon Reviews** | **10** |
| **3** | **whole system architecture of the Sentiment Analysis on Amazon Product Reviews** | **14** |
| **4** | **How the model is predicted by Training and testing** | **17** |
| **5** | **High loss in the left model; low loss in the right model.** | **21** |
| **6** | **Showing the Best model in Bar graph** | **28** |
|  |  |  |

**EXECUTIVE SUMMARY**

Nowadays, social media has become a tremendous source of acquiring user’s opinions. With the advancement of technology and sophistication of the internet, a huge amount of data is generated from various sources like social blogs, websites, etc. In recent times, the blogs and websites are the real-time means of gathering product reviews. However, excessive number of blogs on the cloud has enabled the generation of huge volume of information in different forms like attitudes, opinions, and reviews. Therefore, a dire need emerges to find a method to extract meaningful information from big data, classify it into different categories and predict end user’s behaviors or sentiments. Long Short-Term Memory (LSTM) model and Convolutional Neural Network (CNN) model have been applied to different Natural Language Processing (NLP) tasks with remarkable and effective results. The CNN model efficiently extracts higher level features using convolutional layers and max-pooling layers. The LSTM model is capable to capture long-term dependencies between word sequences. In this study we propose various types of model to get the best accuracy as possible. We have used model such as Natural Language Processing (NLP), Naïve Bayes Model, Support Vector Classifier (SVC), Naïve Bayes Algorithm (NB). We use CountVectorizer to transform a given text into a vector on the basis of the frequency (count) of each word that occurs in the entire text. This is helpful when we have multiple such texts, and we wish to convert each word in each text into vectors (for using in further text analysis). Our approach achieved competitive results using state-of-the-art techniques on the Amazon movie reviews dataset.

**Chapter 1: Introduction**

**1.1: Domain**

As online marketplaces have been popular during the past decades, the online sellers and merchants ask their purchasers to share their opinions about the products they have bought. Everyday millions of reviews are generated all over the Internet about different products, services and places. This has made the Internet the most important source of getting ideas and opinions about a product or a service.

However, as the number of reviews available for a product grows, it is becoming more difficult for a potential consumer to make a good decision on whether to buy the product. Different opinions about the same product on one hand and ambiguous reviews on the other hand makes customers more confused to get the right decision. Here the need for analyzing these contents seems crucial for all e-commerce businesses.

Sentiment analysis and classification is a computational study which attempts to address this problem by extracting subjective information from the given texts in natural language, such as opinions and sentiments. Different approaches have used to tackle this problem from natural language processing, text analysis, computational linguistics, and biometrics. In recent years, Machine learning methods have got popular in the semantic and review analysis for their simplicity and accuracy.

Amazon is one of the e-commerce giants that people are using every day for online purchases where they can read thousands of reviews dropped by other customers about their desired products. These reviews provide valuable opinions about a product such as its property, quality and recommendations which helps the customers to understand almost every detail of a product. This is not only beneficial for consumers but also helps sellers who are manufacturing their own products to understand the consumers and their needs better.

This project is considering the sentiment classification problem for online reviews using supervised approaches to determine the overall semantic of customer reviews by classifying them into positive and negative sentiment. The data used in this study is a set of reviews from Amazon that is collected and provided by Quant Masters.

**1.2: Problem Statement**

Sentiment classification aims to determine the overall intention of a written text which can be of admiration or criticism type. This can be achieved by using machine learning algorithms such as NLP, Naive Bayes, Support Vector Classifier, CNN & RNN. So, the problem that is going to be investigated in the project is as follow:

*Which machine learning approach performs better in terms of accuracy on the Amazon products reviews?*

**1.3: Outline of the Report**

The rest of the thesis is structured as follow: Section 1.4, the Background, consists of essential definitions and theory to understand the other sections of this thesis. It also introduces related work done in the next area of research. This is followed by the Methods, in section 3, where the procedure of the study has been described. The results from the experiments are gathered in section 4 and discussed in section 5. Finally, section 6 concludes the study.

**1.4: Background**

**1.4.1: Sentiment classification and analysis**

Electronic commerce is becoming increasingly popular due to the fact that e-commerce websites allow purchasers to leave reviews on different products. Millions of reviews are being generated everyday by costumers which makes it difficult for product manufacturers to keep track of customer opinions of their products. Thus, it is important to classify such large and complex data in order to derive useful information from a large set of data. Classification methods are the way to tackle such problems. Classification is the process of categorizing data into groups or classes based on common traits (Pandey et al. 2016; Rain 2013). A common concern for organizations is the ability to automate the classification process when big datasets are being used (Liu et. al 2014).

Sentiment analysis, also known as opinion mining, is a natural language processing (NLP) problem which means identifying and extracting subjective information of text sources. The purpose of sentiment classification is to analyze the written reviews of users and classify them into positive or negative opinions, so the system does not need to completely understand the semantics of each phrase or document (Liu 2015; Pang et. al 2002; Turney & Littman 2003).

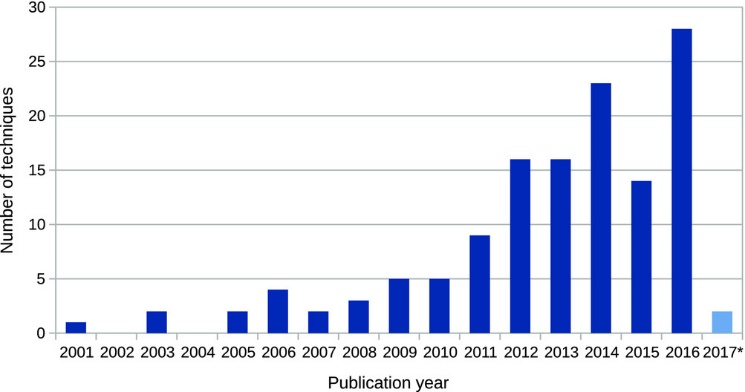
This however is not done by just labeling words as positive or negative. There are some challenges involved. Classifying words and phrases with prior positive or negative polarity will not always work. For example, the word “amazing” has a prior positive polarity, but if it comes with a negation word like “not”, the context can completely change (Singla et. al 2013).

As Ye et. al (2009) state the word “unpredictable” camera has a negative meaning to that camera while “unpredictable” experience is considered as positive for tourists. Sentiment classification has been attempted in different fields such as movie reviews, travel destination reviews and product reviews (Liu et al. 2007; Pang et al. 2009; Ye et al. 2009). Lexicon based methods and machine learning methods are two main approaches that are usually used for sentiment classification.

**1.4.2: Sentiment classification using Machine learning methods**

There is a large number of papers that have been published in the field of machine learning. One of the most used approaches for sentiment classification is machine learning algorithms. This section attempts to cover some of them. One of the first definitions of machine learning that has been provided by Tom Mitchell (1997) in his book Machine Learning is as follow:

*“A computer program is said to learn from experience E with respect to some class of tasks T and performance measure P if its performance at tasks in T, as measured by P, improves with experience E.”*



***Fig 1: Number of Paper Published on sentiment analysis over the past decade***

Machine learning aims to develop an algorithm in order to optimize the performance of the system by using example data. The solution that machine learning provides for sentiment analysis involves two main steps. The first step is to “learn” the model from the training data and the second step is to classify the unseen data with the help of the trained model (Khairnar & Kinikar 2013). Machine learning algorithms can be classified in different categories:

a. supervised learning

b. semi-supervised learning

c. unsupervised learning

***a. Supervised Learning*** the process where the algorithm is learning from the training data can be seen as a teacher supervising the learning process of its students (Brownlee 2016). The supervisor is somehow teaching the algorithm what conclusions it should come up with as an output. So, both input and the desired output data are provided. It is also required that the training data is already labeled. If the classifier gets more labeled data, the output will be more precise. The goal of this approach is that the algorithm can correctly predict the output for new input data. If the output were widely different from the expected result, the supervisor can guide the algorithm back to the right path.

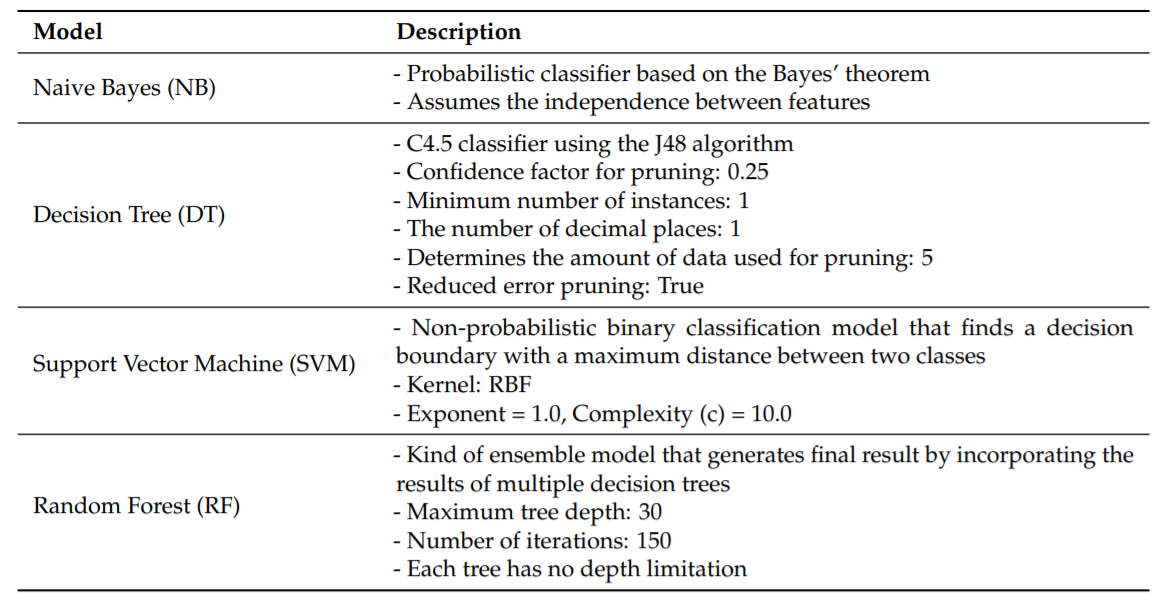
There are however some challenges involved when working with supervised. The supervised learning works fine as long as the labelled data is provided. This means that if the machine faces unseen data, it will either give wrong class label after classification or remove it because it has not “learnt” how to label it (Cunningham et al. 2008).

***b. Unsupervised learning*** in difference with supervised learning is trained on unlabeled data with no corresponding output. The algorithm should find out the underlying structure of the data set on its own. This means that it has to discover similar patterns in the data to determine the output without having the right answers. One of the most important methods in unsupervised learning problems is clustering. Clustering is simply the method of identifying similar groups of data in the data set (Kaushik 2016). For sentiment classification in an unsupervised manner it is usually the sentiment words and phrases that are used. This means that the classification of a review is predicted based on the average semantic orientation of the phrases in that review (Turney 2002). This is obvious since the dominating factor for sentiment classification is often the sentiment words (Berk 2016). This technique has been used in Turney’s study (2002).

***c. Semi-supervised*** learning which has the benefit of both supervised and unsupervised learning, refers to problems in which a smaller amount of data is labelled, and the rest of the training data set is unlabeled. This is useful for when collecting data can be cheap but labelling it can be time consuming and expensive. This approach is highly favorable both in theory and practice because of the fact that having lots of unlabeled data during the training process tends to improve the accuracy of the final model while building it requires much less time and cost (Zhu 2005). In Dasgupta and Vincent Ng. (2009) a semi-supervised learning was experimented where they used 2000 documents as unlabeled data and 50 randomly labeled documents.

**1.5: Methods**

This section presents the method of the study. The programming environments will be described in the first part. How and where the data was gathered as well as the data preparation approach will be discussed in the second part. In the last part, the procedure of machine learning classifiers will be explained. Below is the description of the traditional machine learning models and the parameter settings.

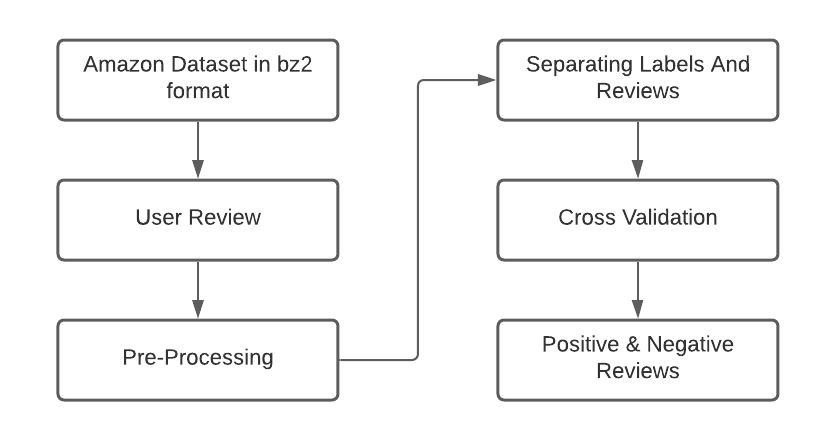


**1.5.1: Programming Environments**

Python is one of the most widely used programming language in machine learning and data science. Python has a huge set of libraries that can be used for solving various machine learning algorithms. The programming language used in this study is Python because of its wealth of libraries and ease of use. Sci-kitlearn is one of many libraries in Python that features a variety of supervised machine learning algorithms (Pedregosa et al. 2011). It provides different classification techniques such as SVC, Naive Bayes. It also orders techniques for feature extraction.

**1.5.2: The data set**

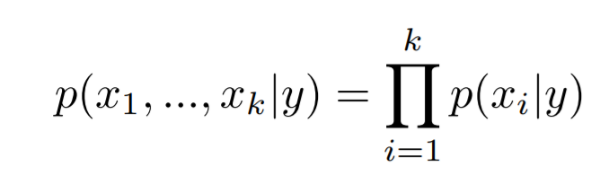
The first step for conducting the research includes data collection for training and testing the classifiers. The data is Provided by Quant Masters. The format of the downloaded file was bz2. The file was is bz2 as there was thousands of reviews and it would take a lot of time to download and read the data. The data set consists of 3600000 reviews of different beauty products. Each review includes nine features as follow:



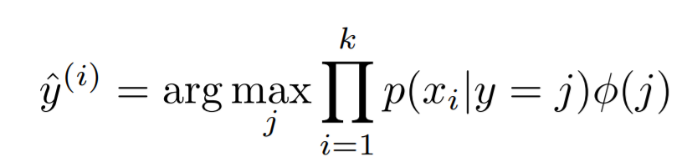
***Fig 2. Overall procedure diagram of the sentiment analysis of Amazon Reviews***

**1.5.3: Naïve Bayes**

Naive Bayes is one of the most common generative learning algorithms for classification problems. This algorithm assumes that x 0 i s are conditionally independent given y, which is called Naive Bayes assumption



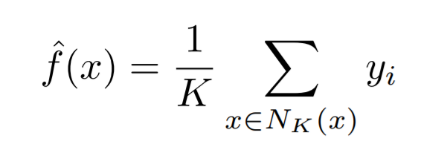
We also incorporated Laplace Smoothing in our model to make it work better. The prediction of an example is given by the formula below:



With the first way of representing review texts, it takes an array of non-negative integers, and models p(xi |y) with multinomial distribution. With the second way of representing review texts using glove dictionary, the inputs are no longer non-negative integers, so we chose to model p(xi |y) with Gaussian distribution.

**1.5.4: K-Nearest Neighbor**

K-nearest Neighbor (KNN) is a nonparametric classification method. It has been widely used recently. When making a prediction, this method fist look for the K = n nearest neighbors of the input. Then, it will assign the majority of that n neighbors’ class. The distance between each neighbor is euclidean distance, which is able to measure the similarity between each data point



The equation above shows the mathematical representation of KNN algorithm. The general idea of KNN is that if the inputs are similar to each other, then the output would be the same.

**1.5.5: Convolutional Neural Network**

Among the existing studies using deep learning to classify texts, the CNN takes advantage of the so-called convolutional filters that automatically learn features suitable for the given task. For example, if we use the CNN for the sentiment classification, the convolutional filters may capture inherent syntactic and semantic features of sentimental expressions. It has been shown that a single convolutional layer, a combination of convolutional filters, might achieve comparable performance even without any special hyperparameter adjustment. Furthermore, the CNN does not require expert knowledge about the linguistic structure of a target language. Thanks to these advantages, the CNN has been successfully applied to various text analyses: semantic parsing, search by query, sentence modeling.

One may argue that the Recurrent Neural Network (RNN) might be better for the text classification than for the CNN, as it preserves the order of the word sequence. However, the CNN is also capable of capturing sequential patterns, as concerns the local patterns by the convolutional filters; for example, the convolutional filters along with the attention technique have been successfully applied to machine translation. Moreover, compared to the RNN, the CNN mostly has a smaller number of parameters, so that the CNN is trainable with a small amount of data. The CNN is also known to explore the richness of pretrained word embeddings. In this project we have designed various model along with CNN model for the sentiment classification and show that our network is better than other deep learning models through experimental results.

**Chapter 2: Related Work**

Due to the proliferation of online reviews, Sentiment analysis has gained much attention in recent years. Therefore, many studies have been devoted to this research area. In this section, some of the most related research works to this thesis are presented.

Joachim (1998) experimented SVM for text classification and showed that SVM performed well in all experiments with lower error levels than other classification methods.

Pang, Lee and Vaithyanathan (2002) tried supervised learning for classifying movie reviews into two classes, positive and negative with the help of SVM and Naive Bayes and maximum entropy classification. In terms of accuracy all three techniques showed quite good results. In this study they tried various features and it turned out that the machine learning algorithms performed better when bag of words was used as features in those classifiers.

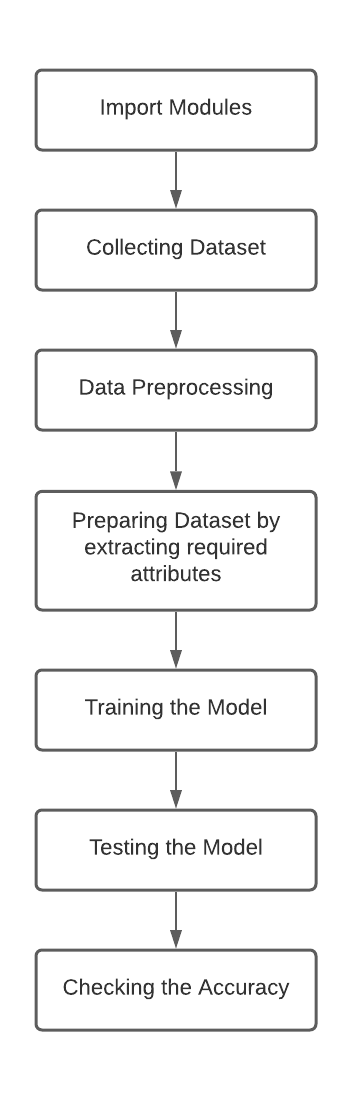
In a recent survey that was conducted by Ye et al. (2009), three supervised machine learning algorithms, Naive Bayes, SVM and N-gram model have been attempted on online reviews about different travel destinations in the world. In this study, they found that in terms of accuracy, well trained machine learning algorithms performs very well for classification of travel destinations reviews. In addition, they have demonstrated that the SVM and N-gram model achieved better results than the Naive Bayes method. However, the difference among the algorithms reduced significantly by increasing the number of training data set.

Chaovalit and Zhou (2005) compared the supervised machine learning algorithm with Semantic orientation which is an unsupervised approach to movie review and found that the supervised approach provided was more reliable than the unsupervised method. According to many research works, Naive Bayes, SVM are two most used approaches in sentiment classification problems (Joachims 1998; Pang et al. 2002; Ye et al. 2009). This thesis, therefore tries to apply supervised machine learning algorithms of Naive Bayes and SVM to the beauty product reviews of Amazon website

**Chapter 3: System Architecture**

**3.1: System Architecture**

**Below diagram depicts the whole system architecture of the Sentiment Analysis on Amazon Product Reviews**



**3.1.1: Import Modules**

***a. Pandas:* pandas** is a software library written for the Python Programming Language for data manipulation and analysis. In particular, it offers data structures and operations for manipulating numerical tables and time series. It is free software released under the three-clause BSD license. The name is derived from the term “panel data”, an econometrics term for data sets  that include observations over multiple time periods for the same individuals. Its name is a play on the phrase "Python data analysis" itself. Wes McKinney started building what would become pandas at AQR Capital while he was a researcher there from 2007 to 2010.

***b. NumPy:* NumPy** is a library for the Python Programming Language, adding support for large, multi-dimensional arrays and matrices, along with a large collection of high-level mathematical functions to operate on these arrays. The ancestor of NumPy, Numeric, was originally created by Jim Hugunin with contributions from several other developers. In 2005, Travis Oliphant created NumPy by incorporating features of the competing Num array into Numeric, with extensive modifications. NumPy is open-source software and has many contributors.

***c. SciKit-learn:* Scikit-learn** (formerly **scikits.learn** and also known as **sklearn**) is a free software machine learning library for the Python programming language. It features various classification, regression and clustering algorithm including support vector machines, random forests, gradient boosting, K-means and DBSCAN, and is designed to interoperate with the Python numerical and scientific libraries NumPy and SciPy.

**3.1.2: Collecting Data sets**

A **data set** is a collection of **data**. In other words, a **data set** corresponds to the contents of a single database table, or a single statistical **data** matrix, where every column of the table represents a particular variable, and each row corresponds to a given member of the **data set** in question.

In Machine Learning projects, we need a training **data set.**It is the actual **data set** used to train the model for performing various actions.

***Why do I need a data set?***ML depends heavily on data, without data, it is impossible for an “AI” to learn. It is the most crucial aspect that makes algorithm training possible… No matter how great your AI team is or the size of your data set, if your data set is not good enough, your entire AI project will fail! I have seen fantastic projects fail because we didn’t have a good data set despite having the perfect use case and very skilled data scientists.

A supervised AI is trained on a corpus of training data.

During an AI development, we always rely on data. From training, tuning, model selection to testing, we use three different data sets: the training set, the validation set ,and the testing set. For your information, validation sets are used to select and tune the final ML model.

You might think that the gathering of data is enough but it is the opposite. In every AI projects, classifying and labeling data sets takes most of our time , especially data sets accurate enough to reflect a realistic vision of the market/world.

I want to introduce you to the first two data sets we need — the training data set and test data set because they are used for different purposes during your AI project and the success of a project depends a lot on them.

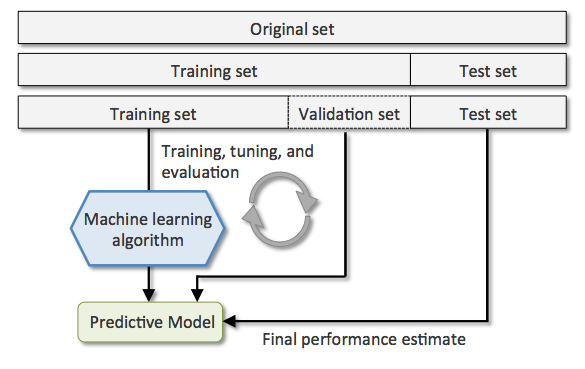
1. The **training data set** is the one used to train an algorithm to understand how to apply concepts such as neural networks, to learn and produce results. It includes both input data and the expected output.

***Training sets make up the majority of the total data, around 60 %.****In testing, the models are fit to parameters in a process that is known as adjusting weights.*

1. The **test data set**is used to evaluate how well your algorithm was trained with the training data set. In AI projects, we can’t use the training data set in the testing stage because the algorithm will already know in advance the expected output which is not our goal.

*Testing sets represent 20% of the data. The test set is ensured to be the input data grouped together with verified correct outputs, generally by human verification.*

Based on my experience, it is a bad idea to attempt further adjustment past the testing phase. It will likely lead to overfitting.



***Fig 4. How the model is predicted by Training and testing***

***What is overfitting?  
A well-known issue for data scientists…*Overfitting** is a modelling error which occurs when a function is too closely fit to a limited set of data points.

***How much data is needed?***All projects are somehow unique but I’d say that you need 10 times as much data as the number of parameters in the model being built. **The more complicated the task, the more data needed.**

***What type data do I need?***I always start AI projects by asking precise questions to the company decision-maker. What are you trying to achieve through AI? Based on your answer, you need to consider what data you actually need to address the question or problem you are working on. Make some assumptions about the data you require and be careful to record those assumptions so that you can test them later if needed.

**3.1.3: Data Pre-processing**

**1. Acquire the dataset**

Acquiring the dataset is the first step in data pre-processing in machine learning. To build and develop Machine Learning models, you must first acquire the relevant dataset. This dataset will be comprised of data gathered from multiple and disparate sources which are then combined in a proper format to form a dataset. Dataset formats differ according to use cases. For instance, a business dataset will be entirely different from a medical dataset. While a business dataset will contain relevant industry and business data, a medical dataset will include healthcare-related data.

### 2. Import all the crucial libraries

Since Python is the most extensively used and also the most preferred library by Data Scientists around the world, we’ll show you how to import Python libraries for data pre-processing in Machine Learning. The predefined Python libraries can perform specific data pre-processing jobs. Importing all the crucial libraries is the second step in data pre-processing in machine learning. The three core Python libraries used for this data pre-processing in Machine Learning are:

* **NumPy** – NumPy is the fundamental package for scientific calculation in Python. Hence, it is used for inserting any type of mathematical operation in the code. Using NumPy, you can also add large multidimensional arrays and matrices in your code.
* **Pandas** – Pandas is an excellent open-source Python library for data manipulation and analysis. It is extensively used for importing and managing the datasets. It packs in high-performance, easy-to-use data structures and data analysis tools for Python.
* **Matplotlib** – Matplotlib is a Python 2D plotting library that is used to plot any type of charts in Python. It can deliver publication-quality figures in numerous hard copy formats and interactive environments across platforms (IPython shells, Jupyter notebook, web application servers, etc.).

### ****3. Import the dataset****

In this step, you need to import the dataset/s that you have gathered for the ML project at hand. Importing the dataset is one of the important steps in data pre-processing in machine learning. However, before you can import the dataset/s, you must set the current directory as the working directory.

**4. Identifying and handling the missing values**

In data pre-processing, it is pivotal to identify and correctly handle the missing values, failing to do this, you might draw inaccurate and faulty conclusions and inferences from the data. Needless to say, this will hamper your ML project.

Basically, there are two ways to handle missing data:

* **Deleting a particular row** – In this method, you remove a specific row that has a null value for a feature or a particular column where more than 75% of the values are missing. However, this method is not 100% efficient, and it is recommended that you use it only when the dataset has adequate samples. You must ensure that after deleting the data, there remains no addition of bias.
* **Calculating the mean** – This method is useful for features having numeric data like age, salary, year, etc. Here, you can calculate the mean, median, or mode of a particular feature or column or row that contains a missing value and replace the result for the missing value. This method can add variance to the dataset, and any loss of data can be efficiently negated. Hence, it yields better results compared to the first method (omission of rows/columns). Another way of approximation is through the deviation of neighbouring values. However, this works best for linear data.

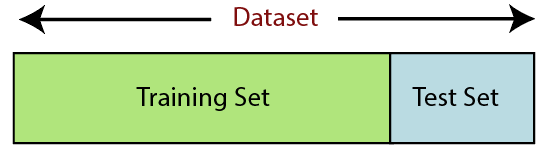
### ****5. Encoding the categorical data****

Categorical data refers to the information that has specific categories within the dataset. In the dataset cited above, there are two categorical variables – country and purchased.

Machine Learning models are primarily based on mathematical equations. Thus, you can intuitively understand that keeping the categorical data in the equation will cause certain issues since you would only need numbers in the equations.

### 6. Splitting the dataset

Splitting the dataset is the next step in data pre-processing in machine learning. Every dataset for Machine Learning model must be split into two separate sets – training set and test set.



Training set denotes the subset of a dataset that is used for training the machine learning model. Here, you are already aware of the output. A test set, on the other hand, is the subset of the dataset that is used for testing the machine learning model. The ML model uses the test set to predict outcomes.

Usually, the dataset is split into 70:30 ratio or 80:20 ratio. This means that you either take 70% or 80% of the data for training the model while leaving out the rest 30% or 20%. The splitting process varies according to the shape and size of the dataset in question.

### 7. Feature scaling

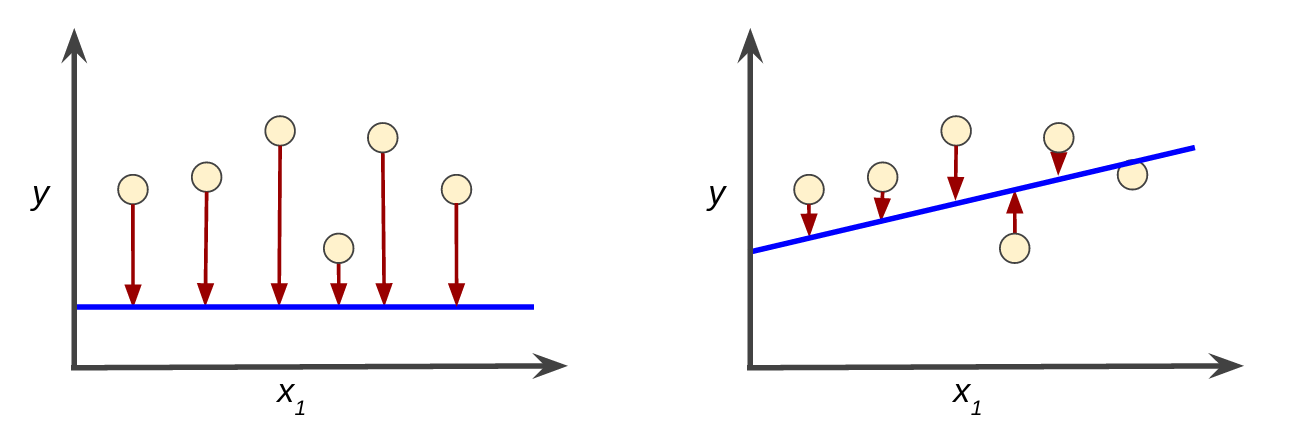
Feature scaling marks the end of the**data pre-processing in Machine Learning.** It is a method to standardize the independent variables of a dataset within a specific range. In other words, feature scaling limits the range of variables so that you can compare them on common grounds.

**3.1.4: Training the Model**

**Training** a model simply means learning (determining) good values for all the weights and the bias from labelled examples. In supervised learning, a machine learning algorithm builds a model by examining many examples and attempting to find a model that minimizes loss; this process is called **empirical risk minimization**.

Loss is the penalty for a bad prediction. That is, **loss** is a number indicating how bad the model's prediction was on a single example. If the model's prediction is perfect, the loss is zero; otherwise, the loss is greater. The goal of training a model is to find a set of weights and biases that have *low* loss, on average, across all examples. For example, Figure 3 shows a high loss model on the left and a low loss model on the right. Note the following about the figure:

* The arrows represent loss.
* The blue lines represent predictions.



***Fig 5. High loss in the left model; low loss in the right model.***

Notice that the arrows in the left plot are much longer than their counterparts in the right plot. Clearly, the line in the right plot is a much better predictive model than the line in the left plot.

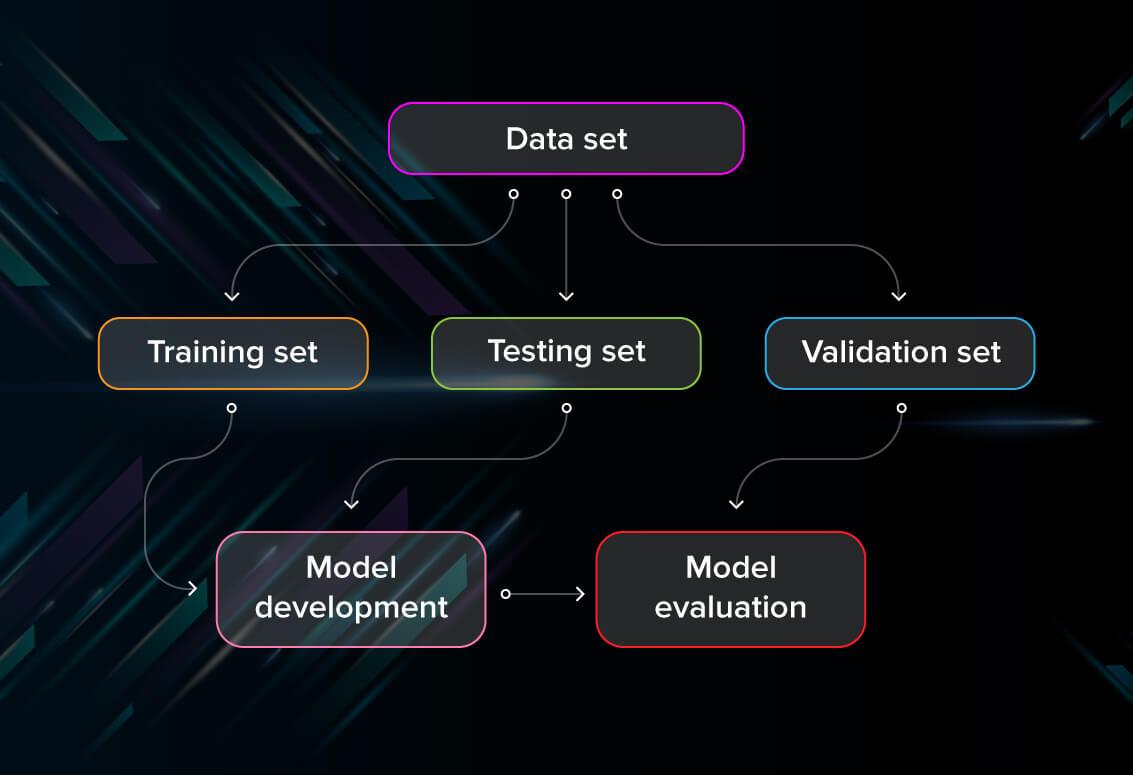
You might be wondering whether you could create a mathematical function—a loss function—that would aggregate the individual losses in a meaningful fashion.

**3.1.5: Testing the Model**

First of all, what are we trying to achieve when performing ML testing, as well as any software testing whatsoever?

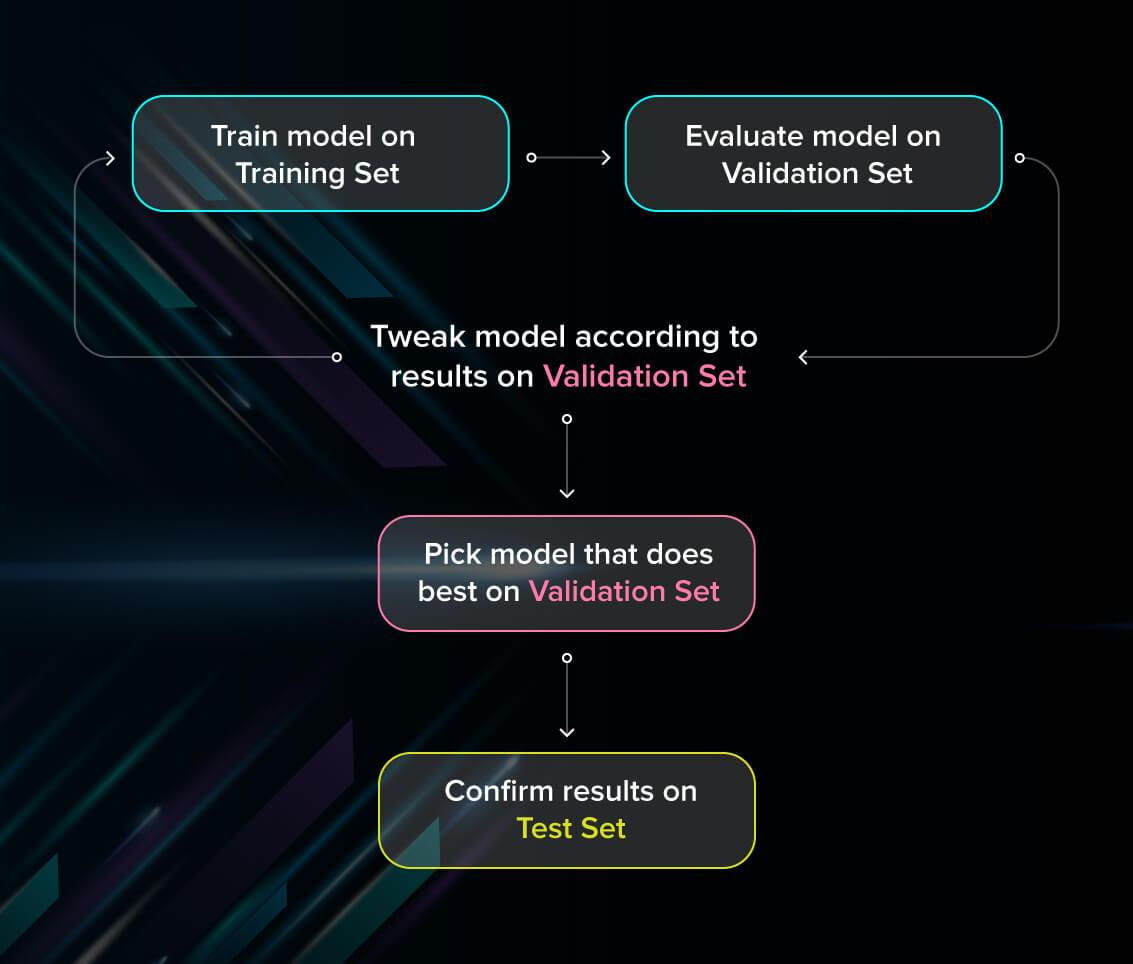
* Quality assurance is required to make sure that the software system works according to the requirements. Were all the features implemented as agreed? Does the program behave as expected? All the parameters that you test the program against should be stated in the technical specification document.
* Moreover, software testing has the power to point out all the defects and flaws during development. You don’t want your clients to encounter bugs after the software is released and come to you waving their fists. Different kinds of testing allow us to catch bugs that are visible only during runtime.

However, in machine learning, a programmer usually inputs the data and the desired behavior, and the logic is elaborated by the machine. This is especially true for deep learning. Therefore, the purpose of machine learning testing is, first of all, to ensure that this learned logic will remain consistent, no matter how many times we call the program.



First of all, you split the database into three non-overlapping sets. You use a training set to train the model. Then, to evaluate the performance of the model, you use two sets of data:

* **Validation set.** Having only a training set and a testing set is not enough if you do many rounds of hyperparameter-tuning (which is always). And that can result in overfitting. To avoid that, you can select a small validation data set to evaluate a model. Only after you get maximum accuracy on the validation set, you make the testing set come into the game.
* **Test set (or holdout set).** Your model might fit the training dataset perfectly well. But where are the guarantees that it will do equally well in real-life? In order to assure that, you select samples for a testing set from your training set — examples that the machine hasn’t seen before. It is important to remain unbiased during selection and draw samples at random. Also, you should not use the same set many times to avoid training on your test data. Your test set should be large enough to provide statistically meaningful results and be representative of the data set as a whole.



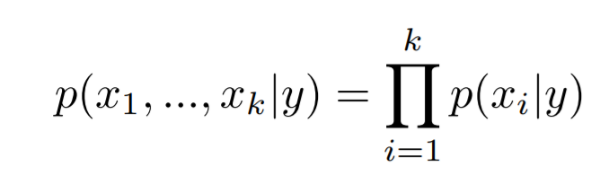
But just as test sets, validation sets “wear out” when used repeatedly. The more times you use the same data to make decisions about hyperparameter settings or other model improvements, the less confident you are that the model will generalize well on new, unseen data. So it is a good idea to collect more data to ‘freshen up’ the test set and validation set.

**Chapter 4: Results**

The results section is where you report the findings of your study based upon the methodology [or methodologies] you applied to gather information. The results section should state the findings of the research arranged in a logical sequence without bias or interpretation.

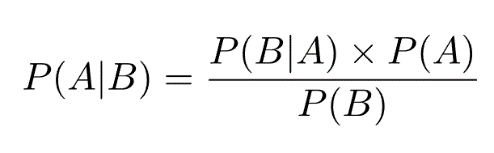
**4.1: Accuracy using Naïve Bayes Algorithm**

Naive Bayes is one of the most common generative learning algorithms for classification problems. This algorithm assumes that x 0 i s are conditionally independent given y, which is called Naive Bayes assumption

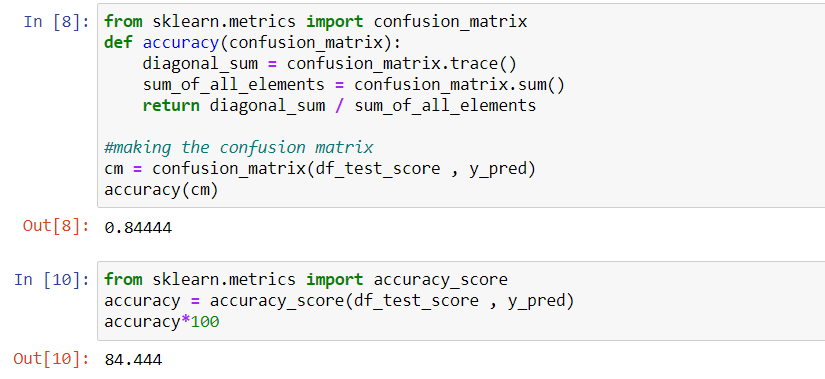


Naïve Bayes calculates the possibility of whether a data point belongs within a certain category or does not. In Text Classification of Amazon Reviews it can be used to categorize words or phrases as belonging to certain tag such as positive or negative.

To decide whether or not a phrase should be tagged as “Positiver” or “Negative”, you need to calculate



Or… the probability of A, if B is true, is equal to the probability of B, if A is true, times the probability of A being true, divided by the probability of B being true*.*



The results from the first experiment where it shows the accuracy of Naive Bayes. Accuracy for Naïve Bayes algorithm is 84.44%.

**4.2: Accuracy using Logistics Regression**

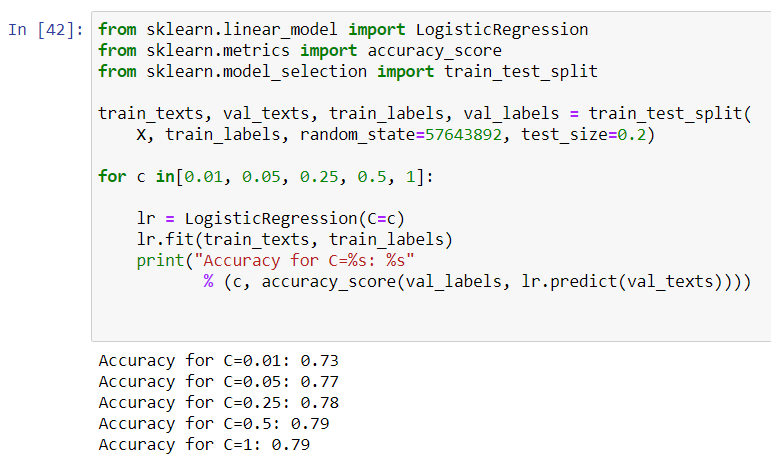
Logistic regression is a calculation used to predict a binary outcome: either something happens, or does not. This can be exhibited as Yes/No, Pass/Fail, Alive/Dead, etc.

Independent variables are analyzed to determine the binary outcome with the results falling into one of two categories. The independent variables can be categorical or numeric, but the dependent variable is always categorical. Written like this:

***P(Y=1|X) or P(Y=0|X)***

It calculates the probability of dependent variable *Y*, given independent variable *X*.

This can be used to calculate the probability of a word having a positive or negative connotation (0, 1, or on a scale between). Or it can be used to determine the object contained in a photo (tree, flower, grass, etc.), with each object given a probability between 0 and 1.



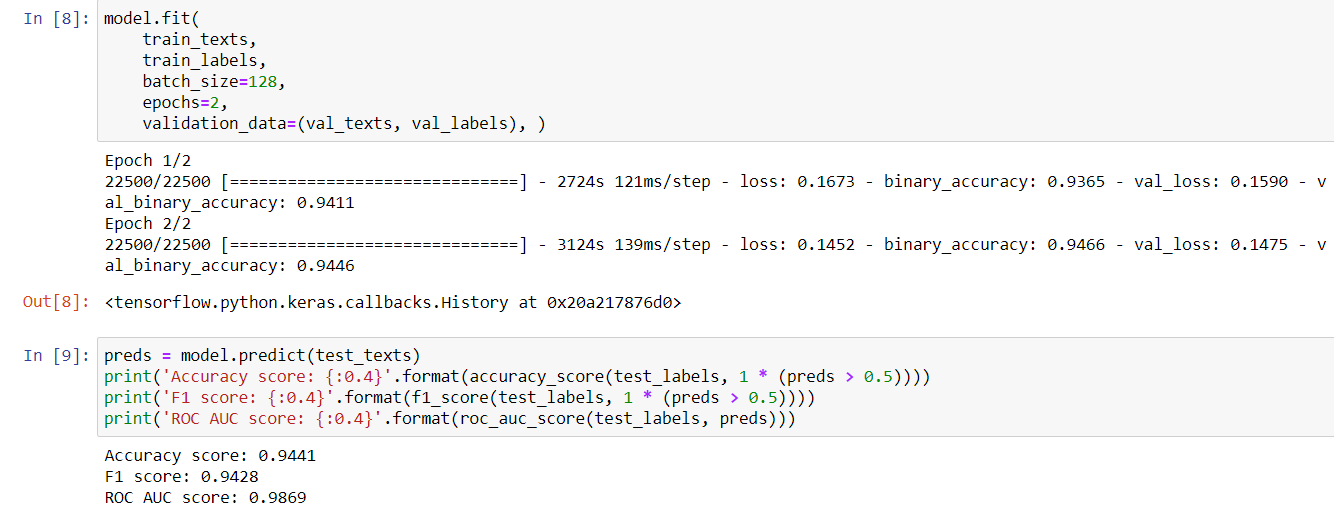
As seen by the above code for the model of Logistics regression. It shows that the accuracy in the model increases the penalty increases. The model provides the best accuracy with 79% when the penalty is 1.

**4.3: Accuracy using Convolution Neural Network**

A convolutional neural network is a class of deep learning which deals with processing image and video data by extracting features from them and build a neural network by assigning them weights and convolved them with a filter to classify and identify an image.

CNN is a prior choice of every data scientist to deal with any Image or video processing data. Using the transfer learning model and modifying it with our layers is also easy.

One may argue that the Recurrent Neural Network (RNN) might be better for the text classification than for the CNN, as it preserves the order of the word sequence. However, the CNN is also capable of capturing sequential patterns, as concerns the local patterns by the convolutional filters; for example, the convolutional filters along with the attention technique have been successfully applied to machine translation. Moreover, compared to the RNN, the CNN mostly has a smaller number of parameters, so that the CNN is trainable with a small amount of data. The CNN is also known to explore the richness of pretrained word embeddings. In this project we have designed various model along with CNN model for the sentiment classification and show that our network is better than other deep learning models through experimental results.

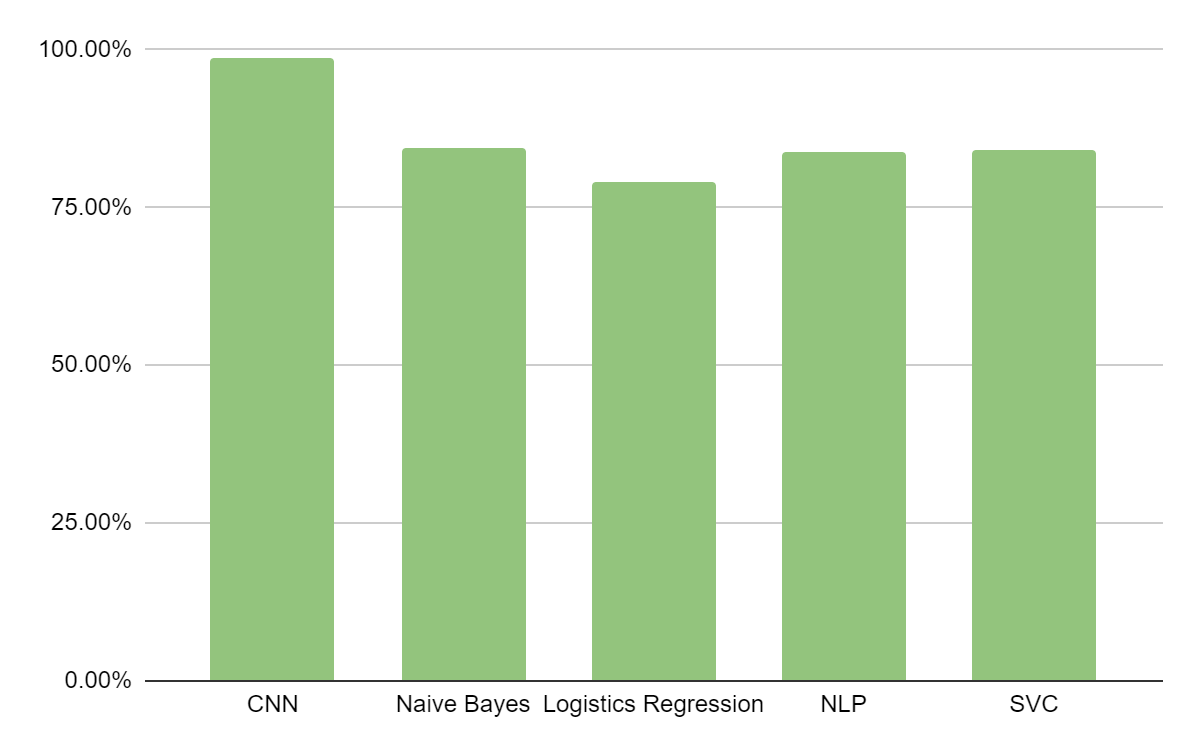


By seeing the above code the Convolution Neural Network provides the best accuracy of 98.69%.

**Chapter 4: Conclusion & Future Work**

This study has applied different machine learning algorithms such as CNN, Logistic Regression, SVC, NLP and Naive Bayes on the Amazon products reviews. The results from the study showed that in terms of accuracy achieved by the different algorithm when the whole data set was used as training and testing data set. As seen by the algorithms, below is the chart which shows the best accuracy by the applied algorithms.

Furthermore this data set can also be applied to DT, K-means & SVM to get better accuracy.



**Bibliography**

1. Jason Brownlee. Supervised and unsupervised machine learning algorithms, Mar 2016
2. **https://www.researchgate.net/publication/338832235\_Sentiment\_Analysis\_of\_Amazon\_Product\_Reviews\_using\_Machine\_Learning**
3. **Wanliang Tan, Xinyu Wang, Xinyu Xu. Sentiment analysis for Amazon Reviews, 2018**
4. Y. Xu, X. Wu, and Q. Wang. Sentiment analysis of yelps ratings based on text reviews, 2015.