A

Project Report on

# "Aspect Based Sentiment Analysis"

Project submitted

In partial fulfilment of the requirements for the degree of

**MASTER OF COMPUTER APPLICATION**

By


**<u>PRAGATI RANI</u>**

Class Roll No.: 001910503003

Examination Roll No.: MCA226003

Registration No.: 149866 of 2019-20


Under the supervision of

**DR. SUDIP KUMAR NASKAR**


Department of Computer Science and Engineering

Faculty of Engineering and Technology

Jadavpur University

Kolkata – 700 032

India

# Jadavpur University

## Faculty of Engineering and Technology

## Department of Computer Science and Engineering

## <u>TO WHOM IT MAY CONCERN</u>

This is to clarify that the project entitled ***"Aspect Based Sentiment Analysis"*** has been completed by ***Pragati Rani.*** This work is carried out under the supervision of ***Dr. Sudip Kumar Naskar*** in partial fulfilment of the requirements for the degree of ***Master of Computer Application*** of the department of ***Computer Science and Engineering, Jadavpur University,*** during the session *2021-2022*. The project report has been approved as it satisfies the academic requirements in respect to project work prescribed for the said degree.

__________________________________

(Signature of Head of the Department)

**Prof. Anupam Sinha**

Head of the Department of Computer Science and Engineering.

__________________________________

(Signature of Project Supervisor)

**Dr. Sudip Kumar Naskar**

__________________________________

(Signature of Dean of the Faculty of Engineering)

**Prof. Chandan Majumdar**

# Jadavpur University

## Faculty of Engineering and Technology

## Department of Computer Science and Engineering

## <u>CERTIFICATE OF APPROVAL</u>

This is to clarify that the project entitled ***"Aspect Based Sentiment Analysis"*** has been completed by ***Pragati Rani.*** This work is carried out under the supervision of ***Dr. Sudip Kumar Naskar*** in partial fulfilment of the requirements for the degree of ***Master of Computer Application*** of the department of ***Computer Science and Engineering, Jadavpur University,*** during the session *2021-2022*. The project report has been approved as it satisfies the academic requirements in respect to project work prescribed for the said degree.

________________________________

(Signature of the internal examiner)

Date:

________________________________

(Signature of the external examiner)

Date:

# DECLARATION OF ORIGINALITY AND COMPLIANCE OF ACADEMIC ETHICS

I now certify that I am the sole author of this thesis and that no part of this thesis has been published or submitted for publication.

To the best of my knowledge, I certify that my thesis does not infringe upon anyone's copyright nor violate any proprietary rights. Any ideas, techniques, quotations, or material from other people's work included in my thesis, published or otherwise, are fully acknowledged under the standard referencing practices.

I declare that this is a true copy of my thesis, including any final revisions. This thesis has not been submitted for higher degree to any other University or Institution.

**Candidate's Name:** Pragati Rani

**Registration No.** **:** 149866 of 2019-20

**Class Roll No.** **:** 001910503003

**Project Title** **:** Aspect Based Sentiment Analysis

________________________________

(Signature of the Candidate)

# ACKNOWLEDGEMENTS

Firstly, I would like to thank my thesis advisor and also the project supervisor, ***Dr. Sudip Kumar Naskar*** of the department of ***Computer Science and Engineering*** of ***Jadavpur University.*** The door to his office was always open whenever I ran into a trouble spot or had questions about my project.

I would also like to thank the experts involved in the validation survey for this project, ***Dr. Anupam Sinha, the Head of The Department of Computer Science and Engineering*** and all the faculty members of the ***Department of Computer Science and Engineering***.

Finally, I must express my profound gratitude to my parents and my friends cum classmates for providing me with unfailing support and continuous encouragement throughout my years of study and throughout the process of my project work and writing this thesis. This accomplishment would not be possible without them.

Thank you.

(Signature of the Candidate)

# ABSTRACT

Nowadays, reviews have become one of the primary sources of feedback on many contents like the hotel, food, products, and more. These reviews are beneficial for the consumers and producers. We use sentiment analysis to analyze this review, which helps to give the polarity, i.e. positive, negative or neutral and allows producers to enhance their productions or businesses according to customer satisfaction, whereas consumers predict the future convenience. Most decision-making procedures are based on the decision maker's preferences and public perceptions of potential alternatives. In the multi-criteria decision-making field, user preferences have been heavily considered. Sentiment analysis, on the other hand, is a branch of natural language processing devoted to the creation of systems capable of analyzing reviews and determining their polarity. In this work, aspect-based sentiment analysis extracts the coarse-grained sentiments behind the hotel reviews depending on the specific aspects. In analyzing the sentiments of the aspects from the review, we first preprocessed the reviews and then feature selection was done. Then classifications of the sentiment polarity of the aspects were done using supervised learning models and deep learning models and various metric measures like F1-score, accuracy, precision and recall have been noted down.

***Keywords*** *- Natural Language Processing (NLP), Sentiment Analysis, Aspect Based Sentiment Analysis, Supervised Learning, Deep Learning.*

# CONTENTS

# LIST OF FIGURES & TABLES

**TITLES**

# 1. INTRODUCTION

In the Internet era, reviews play a significant role in everyday lifestyle. The Internet has become a forum where consumers give their feedback on products and services and use these reviews to buy the product and avail of the services in the future. Reviews increase customer interaction by gaining the customers' trust, leading to an upgrade in business profits. The consumer's decisions are very much affected by these reviews.

Sentiment Analysis is the technique in Natural Language Processing which predicts the sentiment or attitude of the given review or text. It classifies the text into polarities like negative, positive or neutral. Online commerce sites like Amazon, Book show, Uber, etc., use this technique to enhance their business with the help of customer interaction and create a reputation in the market. The government uses this to strengthen their weakness by analyzing the public opinion on social media sites like Twitter, Facebook, etc.

Aspect Based Sentiment Analysis is one type of Sentiment Analysis. It is a fine-grained categorization of opinion based on a specific element (aspects) in a given review and identifies each element's polarity. In the Sentiment Analysis, we mark overall feedback, whereas, in Aspect Based Sentiment Analysis, we categorize the text according to aspect and identify the polarity individually, leading to the overall polarity of the review. The first step in this analysis is Aspect Extraction to identify the aspects in the review. Secondly, extraction of sentiment for the aspect and formulation of interaction between aspect and sentiment word for obtaining the sentiment analysis accuracy.

The objective of this thesis is to do the Aspect Based Sentiment Analysis on the dataset of restaurant reviews using different deep learning models like LSTM, RNN and CNN models, and supervised models like SVM and Naive Bayes models. This process involves three steps: data preprocessing, feature selection and classification of aspect category and sentiment polarity. Data preprocessing focuses on cleaning the data. Feature selection is done by Bag of words and Glove embedding. Dataset used in this thesis is the restaurant reviews obtained from SemEval-2014, which focus on aspect-based sentiment analysis. Each review in this dataset has an aspect category (ambience, miscellaneous, price, food and services) and sentiments (positive, negative, neutral and conflict).

# 2. LITERATURE SURVEY

There have been several approaches to the Aspect based Sentiment Analysis proposed in the past , one of which is XRCE: Hybrid Classification for Aspect-based Sentiment Analysis (Brun et al., SemEval -2014)[1] , the research by Xerox Research Center Europe (XRCE) have demonstrate a system created for SemEval-2014 Task 4 on Aspect-Based Sentiment Analysis. The system is built on a powerful parser that feeds linguistic data dedicated to aspect categories and aspect categories polarity categorization into several classifiers. The work on the restaurant domain1 for the four subtasks, aspect term and category identification, and aspect term and category polarity, is mostly presented. They concentrate on four primary tasks: detecting aspect terms and categories, as well as their polarity. While the detection of aspect words and their respective polarities occurs at the language level, they use the liblinear package (Fan et al., 2008) to train our models for the detection of aspect categories and their corresponding polarities. They train a single classifier to detect the categories, and then a second classifier for each category to detect the polarities associated with that category. They employ 10-fold cross-validation in both scenarios. The performance of research by XRCE in aspect category detection was 82% and in aspect category polarity was 77%.

In the paper, “An unsupervised aspect-sentiment model for online reviews” by Samuel Brody and Noemie Elhadad [2], they offer an unsupervised method for extracting characteristics from review text and calculating sentiment. The method is straightforward and adaptable in terms of domain and language, and it considers the impact of aspect on sentiment polarity, which has been overlooked in earlier research. They show that it works well on both component tasks, producing results that are comparable to more complex semi-supervised algorithms that rely on manual annotation and vast knowledge sources. They devised the following approach to determine sentiment polarity. They extracted the relevant adjectives for each aspect, created a conjunction graph, determined the seed set automatically (or manually for comparison), and propagated the polarity scores to the rest of the adjectives.

As an extra element in the classification process, the National Research Council of Canada [3] used a Multi Class Support Vector Machine (SVM) and a dictionary-based technique. It achieves an F1-

Measure classification performance of 88.57 percent. Another study conducted by the University of West Boheemia [4] found that utilising the Maximum Entropy classifier with 12 characteristics such as words, LDA, bigrams, word clusters, tf-idf, and other features resulted in an F1-Measure of 81.04 percent.

In a paper, "Aspect-based sentiment analysis to review products using Naïve Bayes" by Mohamad Syahrul Mubaroka, Adiwijaya, Muhammad Dwi Aldhi [5], Sentiment analysis was utilised in this study to examine and extract sentiment polarity from product reviews based on a certain feature of the product. Data preparation, which included part-of-speech (POS) tagging, feature selection using Chi Square, and classification of sentiment polarity of aspects using Nave Bayes, were all part of this study. The system can perform aspect-based sentiment analysis, according to evaluation results, with the greatest F1-Measure of 78.12 percent.

The tasks in this paper "Mining and summarizing customer Review" by Minqing Hu and Bing Liu [6], are completed in three steps: (1) mining product features that customers have mentioned; (2) identifying opinion sentences in each review and determining whether each opinion sentence is positive or negative; and (3) summarising the results. The system first downloads (or crawls) all of the reviews and stores them in the review database, based on the inputs. It then locates the "hot" (or frequent) aspects on which a large number of people have voiced their thoughts. Then, using the resulting frequent features, the opinion words are retrieved, and semantic orientations of the opinion words are detected using WordNet. The machine then discovers those rare features using the extracted opinion words. The orientation of each opinion sentence is identified in the final two processes, and a final summary is created. A system, called FBS (Feature-Based Summarization), based on the proposed techniques has been implemented in C++ and has the average accuracy for the five product is 84%.

Another paper "Determining Sentiment of Opinions" by Soo-Min Kim and Eduard Hovy [7], proposed a system which identifies the person who holds the opinion about the topic automatically, and the sentiment related to each opinion. The system contains a module for determining word sentiment and another for combining sentiments within a sentence. We experiment with various models of classifying and combining sentiment at word and sentence levels.

According to the paper "Thumbs up? Sentiment classification using Machine Learning Techniques" by Pang, B. L. Lee, and S. Vaidyanathan [8], the authors look at the difficulty of categorising papers based on general sentiment rather than topic, such as evaluating whether a review is good or negative. They discovered that traditional machine learning algorithms beat human-produced baselines when using movie reviews as data. The three machine learning algorithms they used Naive Bayes, maximum entropy classification, and support vector machines did not perform as well on sentiment classification as they did on traditional topic-based categorization. Finally, they look into characteristics that make the sentiment classification task more difficult. The three-fold cross validation accuracy of the Naïve Bayes, maximum entropy classification, and support vector machines are 78.7%, 77.7% and 81.6% respectively and the baseline results raged from 50 to 69%. Here SVM works good whereas Naïve Bayes is worst but doesn't have large difference.

In the paper "Beyond the Stars: Improving Rating Predictions using Review Text Content" by Ganu, Gayatree, Noemie Elhadad [9], and Amelie Marian, their research focuses on extracting this data from free-form text reviews and applying it to improve the user experience while browsing reviews. They have concentrated on enhancing recommendation accuracy in the context of a restaurant review. In their work, they discuss their categorization efforts as well as the insights they got into user-reviewing behaviour as a result of them. They present novel ad-hoc and regression-based recommendation techniques that take the textual component of user evaluations into consideration. Their findings suggest that employing textual information yields superior general or personalised review score predictions than those based on users' numerical star ratings. They have used the metric Mean Squared Error (MSE) to evaluate their predictions. They have presented a recommendation algorithm which relies on topic and sentiment information automatically obtained from the text of reviews. They evaluate the performance of our system by making fine grained rating predictions (in the range [1:5]) which is, notably, a harder task than making binary recommendations.

Another paper "Extracting Product Features and Opinions from Reviews (Popescu & Etzioni, HLT 2005) [10]" This study proposes OPINE, an unsupervised information extraction method that mines reviews to create a model of significant product attributes, how reviewers rate them, and how they compare among items. On the feature extraction test, OPINE achieves a 22 % higher precision (with only a 3 % lower recall) than previous studies. The new application of relaxation labelling by OPINE for determining the semantic orientation of words in context results in strong performance on tasks such as finding opinion phrases and polarity.

The work in the paper "Thumbs up or thumbs down? semantic orientation applied to unsupervised classification of reviews" by P.D. Turney [11], proposes a straightforward unsupervised learning approach for categorising reviews as either recommended (thumbs up) or not recommended (thumbs down) (thumbs down). The average semantic orientation of the sentences in the review that contain adjectives or adverbs predicts the classification of the review. When a phrase has nice connotations (e.g., "subtle nuances"), it has a positive semantic orientation; when it has bad associations (e.g., "extremely cavalier"), it has a negative semantic orientation. The mutual information between the supplied phrase and the word "excellent" minus the mutual information between the given phrase and the word "poor" is used to compute the semantic orientation of a phrase in this research. If the average semantic orientation of the terms in a review is positive, it is classified as recommended.

# 3. METHODOLOGY

## ***3.1 <u>TEXT PRE-PROCESSING</u>***:

Data prepossessing is a technique of converting the raw data into a structured data format that machine learning models can understand.

Data pre-processing mainly checks the quality of the data before applying machine learning by accuracy, completeness or interpretability.

In Natural Language Processing, text pre-processing is a method to clean the data and make it in a structured format for model training. Text pre-processing helps the machine learning model to predict more accurately.

The steps to clean the data are as follows:

1. Removal of punctuations like '!" #$%&'()*+,-./:;?@[\]^_`{|}~' and digits from the text.
2. Converting the text to lower case.
3. Removal of stop words (commonly used words) from the text: These words don't add any meaning to the analysis. NLTK library is used for this step as it consists of a list of considered stop words for the English Language.
4. Stemming: In this process, the words of the text are reduced to their root form, but it does not guarantee that its root form will not lose its meaning.

   Example: programmer --->program

   Early ---> Earli

5. Lemmatization: this process also stems the word to its root form and does not lose its meaning. It has a predefined dictionary for the English language that stores the context of words and uses it during the diminishing.

   Example:

| **<u>Original word</u>** | **<u>After stemming</u>** | **<u>After Lemmatization</u>** |
|---|---|---|
| geese | gees | goose |

**In this project**, text pre-processing is done by removing punctuations and digits, changing the text to lower case, removing the stop words except for the not terms, and lemmatization is done using the spacy library.

# 3.2 SENTIMENT ANALYSIS MODELS:

## 3.2.1 SUPERVISED LEARNING MODELS:

Supervised learning is one of the types of machine learning in which a machine is trained with well-labelled data and predicts the output based on the data provided. This learning aims to find a mapping function to map the input variable(x) with the labelled or output variable (y).

The working of this learning is shown by the following figure:

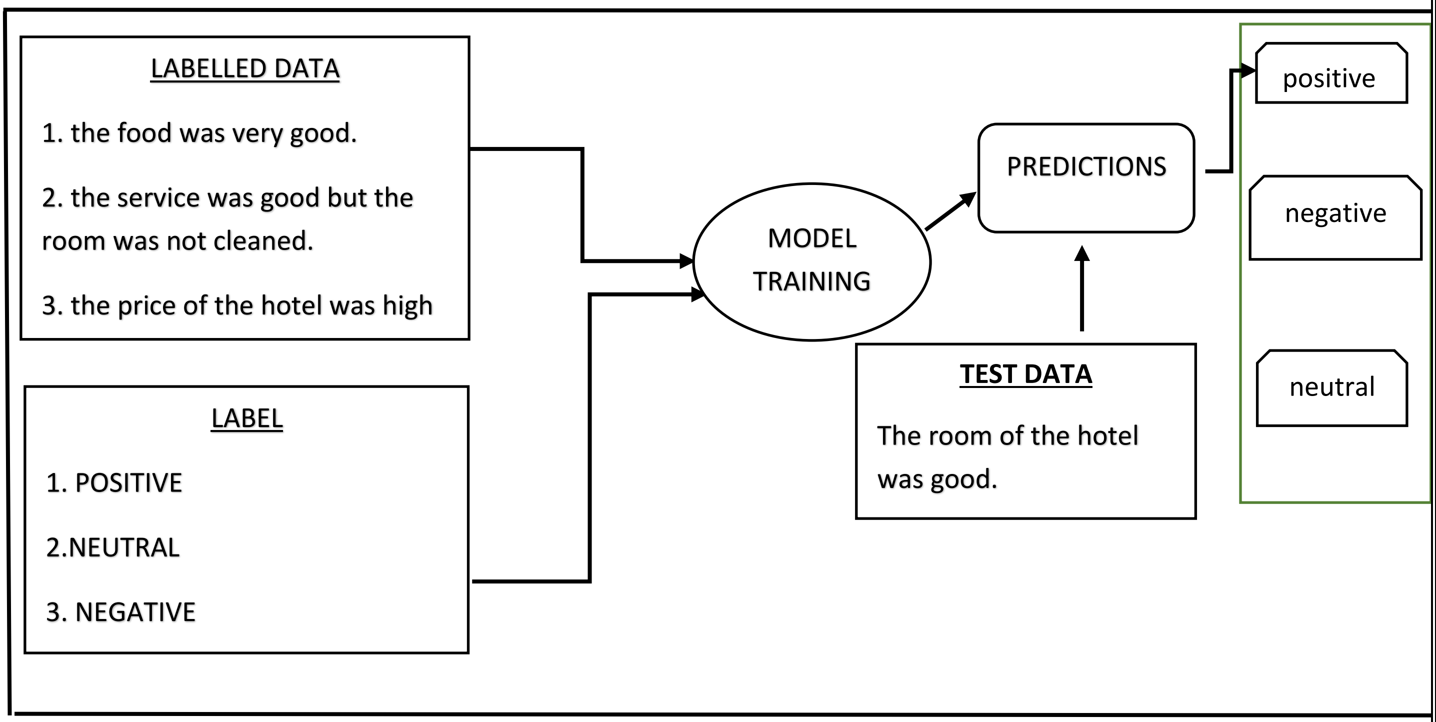


**Figure 3.1: Working Principle of Supervised Learning**

There are two types of supervised learning: Classification and Regression. **In this project**, the classification models used are Support Vector Machine and Naive Bayes Classifiers.

# 3.2.1.1 SUPPORT VECTOR MACHINE:

Support Vector Machine is one of the most popular supervised learning models used for classification and regression problems. The SVM model aims to create a hyperplane or a line that separates the data into different classes to put the new data point in the correct category in the future. SVM classifier can be used for face detection, image classification, text categorization, etc. There are two types of SVM models: Linear SVM and Non-linear SVM.

## ❍ WORKING PRINCIPLE OF SVM

- This algorithm helps to find the best decision boundaries to separate the classes in N-dimensional space which can be helpful for classifying the data points. This best boundary is called Hyperplane of SVM.
- This classifier finds the data points or vectors closest to the hyperplane. These points are called support vectors, which affect the position of the hyperplane.
- The distance between the vectors and hyperplane is called margin.
- The goal of SVM is to maximize this margin.

The following figure shows the working principle:

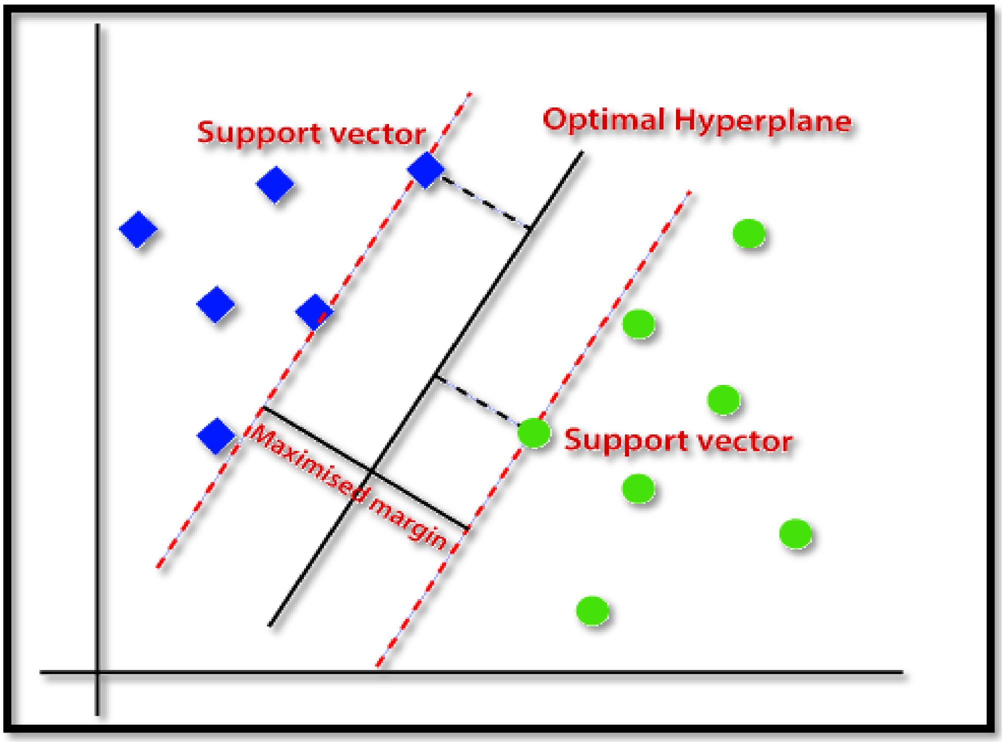


**Figure 3.2: Working Principle of Support Vector Machine (SVM)**

For determining the aspect category and its respective polarity, SVM creates a hyperplane that separates the review into different Ratings ranging between 1 to 20. The range of rating is given according to the aspect category and its respective polarity (like there is five aspect category and each category has the possibility of having four sentiments).

### 3.2.1.2 NAÏVE BAYES CLASSIFIER:

The Naive Bayes classifier is also a supervised learning model based on the Bayes theorem of probability. This learning model is the probabilistic classifier, mainly used for text classification. It is one of the most effective classification algorithms that help to predict the output rapidly by building a fast machine learning model. This classifier assumes that the input feature fed in the model is independent of each other, i.e. changing one input feature will not affect any other feature. This assumption may or may not be true, therefore called Naive. since it is based on the principle of Bayes Theorem, this is known as the Bayes. This classifier is used in sentiment analysis, spam detection and classifying articles. There are three types of Naïve Bayes Classifier: **Multinomial, Bernoulli and Gaussian Naïve Bayes**.

The disadvantage of this classifier is the assumption that the input feature must be independent, which is impossible in real-time.

#### ❍ WORKING PRINCIPLE OF NAÏVE BAYES:

- This classifier is based on principle of Bayes Theorem and Bayes theorem is written as:

$$P(A|B) = \frac{P(B|A)P(A)}{P(B)}$$

  Where A is the hypothesis and B is the evidence and P(A|B) is the Probability of A happening when B occurred.

- Now we rewritten the Bayes Theorem as:

$$\mathrm{P}(\mathrm{y}|\mathrm{X}) = \frac{\mathrm{P}(\mathrm{X}|\mathrm{y})\mathrm{P}(\mathrm{y})}{\mathrm{P}(\mathrm{X})}$$

where ‘X’ are the features and ‘y’ are the label.

X can be written as:

$$X = \left(x_1, x_{2,\dots}, x_n\right)$$

- Here, $x_1, x_{2,\dots}, x_n$ represent the features, and now substituting for ‘X’, we get :

$$\mathrm{P}\left(\mathrm{y}|x_1, x_{2,\dots}, x_n\right) = \frac{\mathrm{P}(x_1|\mathrm{y})\mathrm{P}(x_2|\mathrm{y})\dots\mathrm{P}(x_n|\mathrm{y})\mathrm{P}(\mathrm{y})}{\mathrm{P}(x_1)\mathrm{P}(x_2)\dots\mathrm{P}(x_n)}$$

- Now the proportionality as the denominator remains constant for all the features introduced in the dataset in the above equations. So the above equation becomes:

$$\mathrm{P}\left(\mathrm{y}|x_1, x_{2,\dots}, x_n\right) \propto \mathrm{P}(\mathrm{y}) \prod_{i=1}^{n} P(x_i|\mathrm{y})$$

- Since the classification may be multivariate, we find the label ‘y’ with maximum probability i.e.

$$\mathrm{y} = argmax_y\ \mathrm{P}(\mathrm{y}) \prod_{i=1}^{n} P(x_i|\mathrm{y})$$

- Thus, we obtain the label for given features.

Using this Classifier, we are predicting the aspect category and its polarity of the Restaurant by giving the ratings ranging from 1 to 20 (according to aspect category and polarity) as the output feature and input feature is the matrix of noun, adjective and verbs extracted from the review using POS tagger.

## *3.2.2 DEEP LEARNING MODELS:*

Deep learning is a machine learning process and artificial intelligence that behaves identically as a human brain gathers information or knowledge. Neural Networks are the type of Deep Learning which uses interconnected nodes or neurons in a layered structured identical to the human brain. Neural Network is a method in artificial intelligence which make the computer learn to process the data similar to the human brain.

The general Artificial Neural Network is shown in the figure:

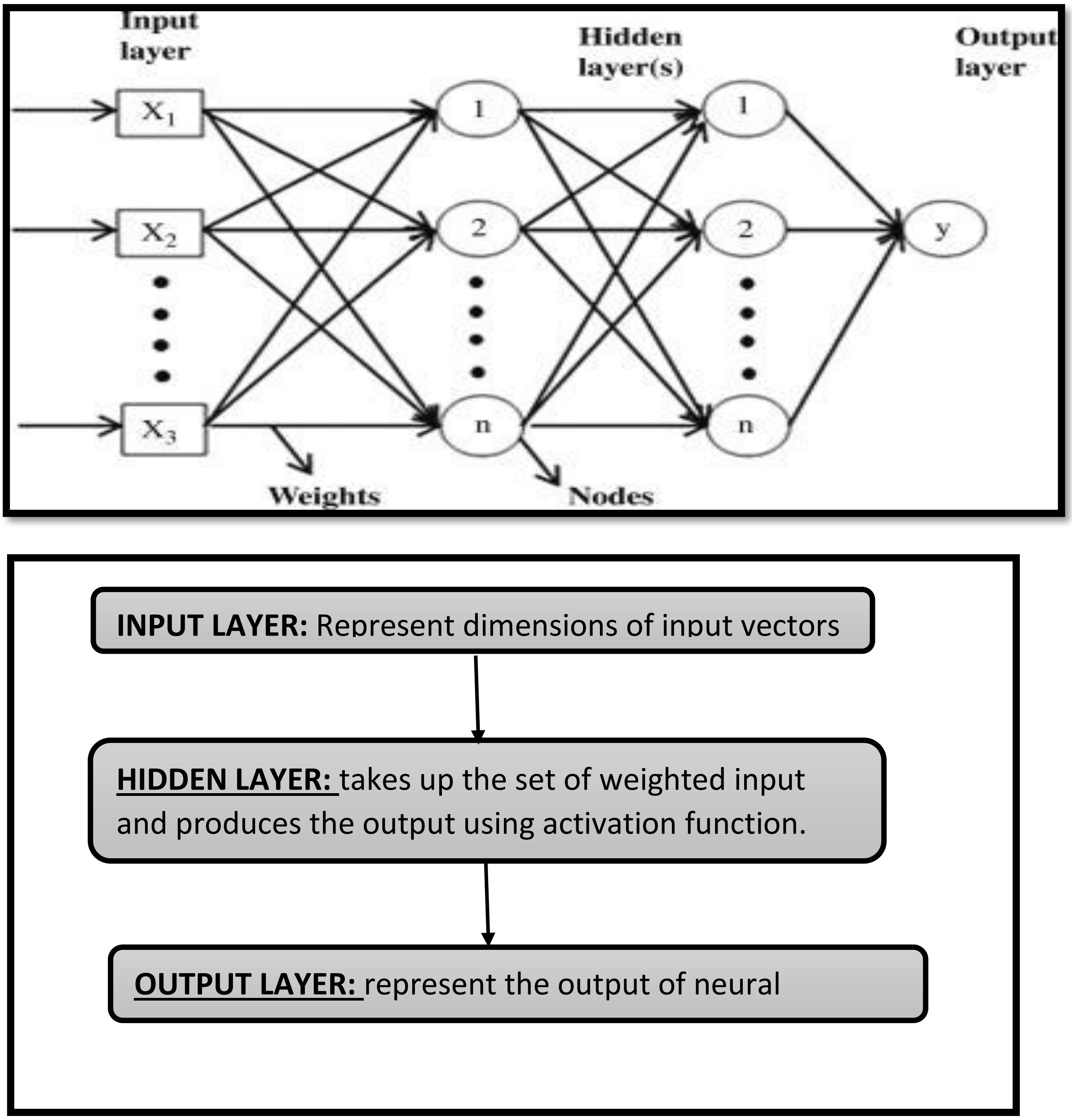


**FIGURE 3.3: WORKING ARCHITECTURE OF NEURAL NETWORK**

## *3.2.2.1 RECURRENT NEURAL NETWORK:*

In typical neural networks, the input is not dependent on output and vice versa. Still, when we have to predict the next word in a text, we need to remember the previous terms and these phenomena are solved by a Recurrent Neural Network in the form of a hidden layer. The previous output step is used as input for the current step in the Recurrent Neural Network. So, at a point in time, two inputs are fed into the current step in the Recurrent Neural Network; one is the current step input and the other is the previous step output. The most important feature of a Recurrent Neural Network is the memory cell, and it stores the information of the previous calculations. Unlike the other neural networks in Recurrent Neural Network, the same parameters are used for all the inputs, and the same task is performed. Thus, the complexity is less here.

Here first, the input is given, and then the current state is calculated with the help of current input and previous state output, and now the current state becomes the previous step for the next step. After all the time steps are done, the final state calculates the result.

## ❍ ***WORKING PRINCIPLE OF RNN:***

- The information cycles in loop i.e. when the output is predicted, the model takes into the account, the current input and the output learned from previous inputs.
  - The input is fed to the model.
  - Then, the current state is predicted with the help of current input and previous state output.
  - Now, the current output states become the previous output for next step.
  - After all the steps, the final state circulates the result.

The following figure shows the working of RNN:

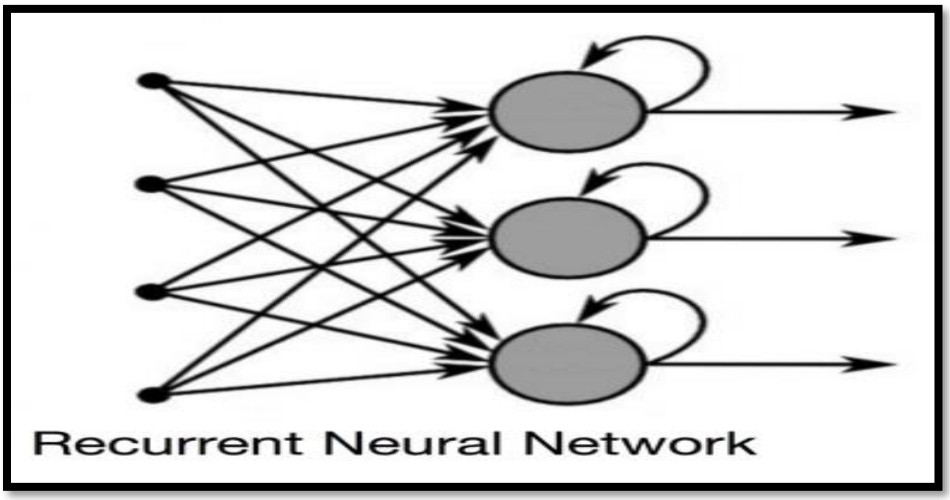


**FIGURE 3.4: WORKING OF RNN**

**In this project**, for determing the aspect category and its polarity, I have created a model which has embedding layer of 50 dimension (embedding layer is created using GloVe). The next layer is the SimpleRNN layer which has 100 hidden neurons and at last we have Dense layer with sigmoid functions. The ouput will be in the form of bag of words i.e it will have 1 in place of aspect category and its polarity (as both are appended) or else 0.

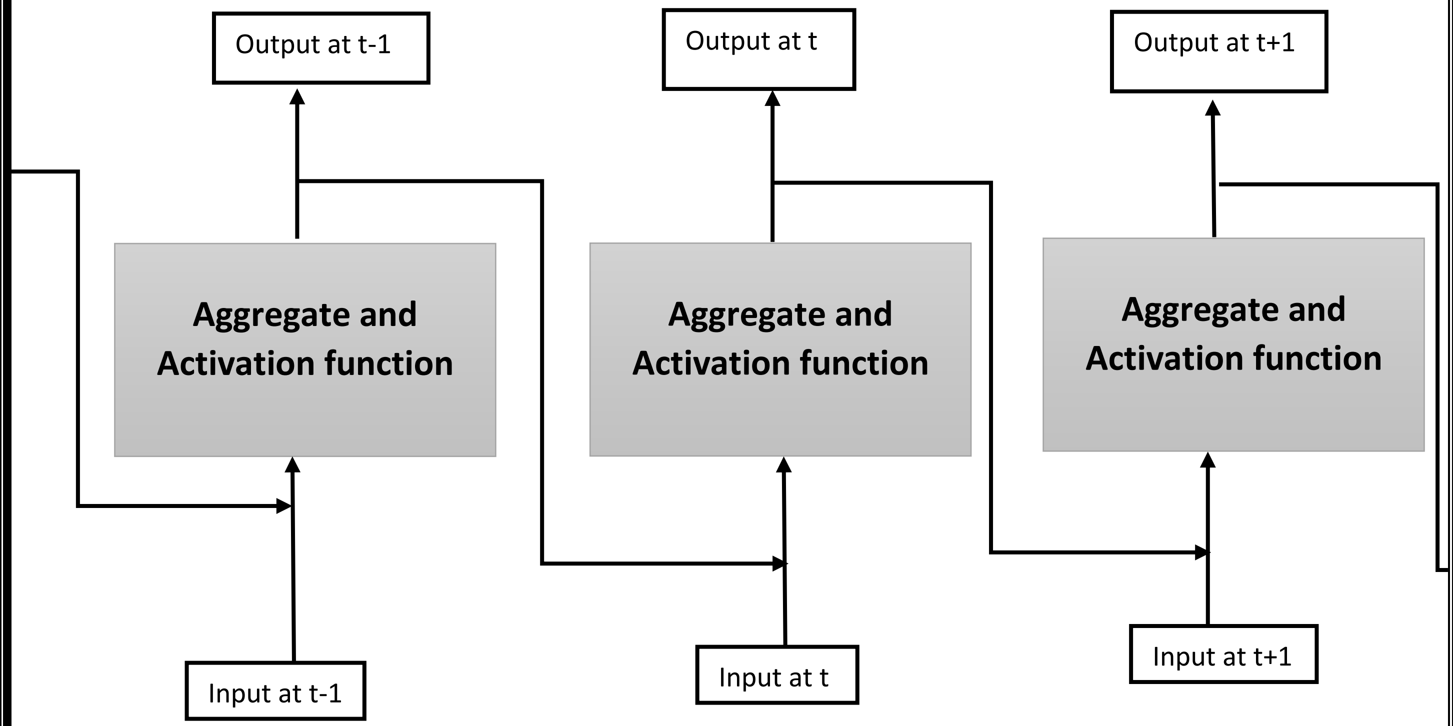

FIGURE 3.5: UNROLLED RNN OVER TIME

## *3.2.2.2 LSTM MODEL:*

In a Recurrent Neural Network, some previous information is preserved with the help of a memory cell. Still, suppose there is a large chunk of text; a recurrent neural network may omit some essential details because it has difficulty passing information from previous time steps to later time steps. This is called Short Term Memory, and Recurrent Neural Network suffers from it. Here comes the LSTM or Long Short-Term Memory, a special kind of Recurrent Neural Network. It can remember information for a longer period because it only remembers relevant information and forgets all the information that is not relevant.

The key feature of Long Short-Term Memory is Cell State, and it acts as the memory of the network. The Cell State carries only the important information while processing the sequence, and new relevant information gets added to it. The old non-relevant information gets chopped by which the effects of short-term memory are reduced, and all this happens by using some gates. These gates are neural networks that decide which information is to be kept and which to forget in the cell state while training.

In Long Short-Term Memory, there is a forget gate which decides if the information is important enough or not. Here, the information from the previous step and the current input are fed into the sigmoid function, giving the output between zero and one. If the value is closer to 0, the information is meant to forget, and if the value is more relative to 1, then the information is kept.

## ❍ ***WORKING PRINCIPLE OF LSTM:***

- The only difference in the workflow of LSTM and RNN is that the Internal Cell State ($c_i$) is passed with the hidden state ($h_i$).

- The basic workflow is shown in the below figure:

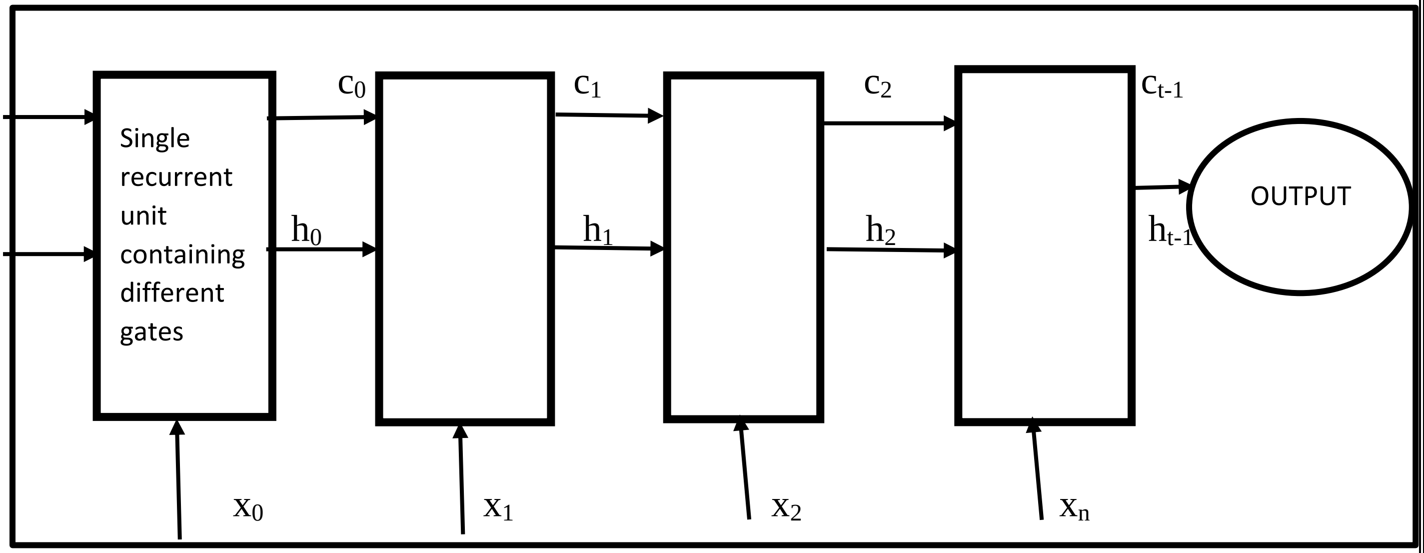


**Figure 3.6: Basic Workflow Of LSTM**

- The Input Gate is used to update the cell state. The hidden state information of a current state is passed in the tanh activation function, which gives the output between -1 and 1 and then the output of the sigmoid and the output of tanh are multiplied. Thus, the sigmoid output becomes the deciding factor in keeping the tanh output. A hidden state does the prediction and preserves previous states' information. The previous hidden state and current state input are fed into the sigmoid activation function. Modified cell states are passed into the tanh activation function. Then, the tanh and sigmoid output multiply and give the hidden state output. The modified cell state and hidden form are carried over in the next time step.

- The above stated working is shown below:

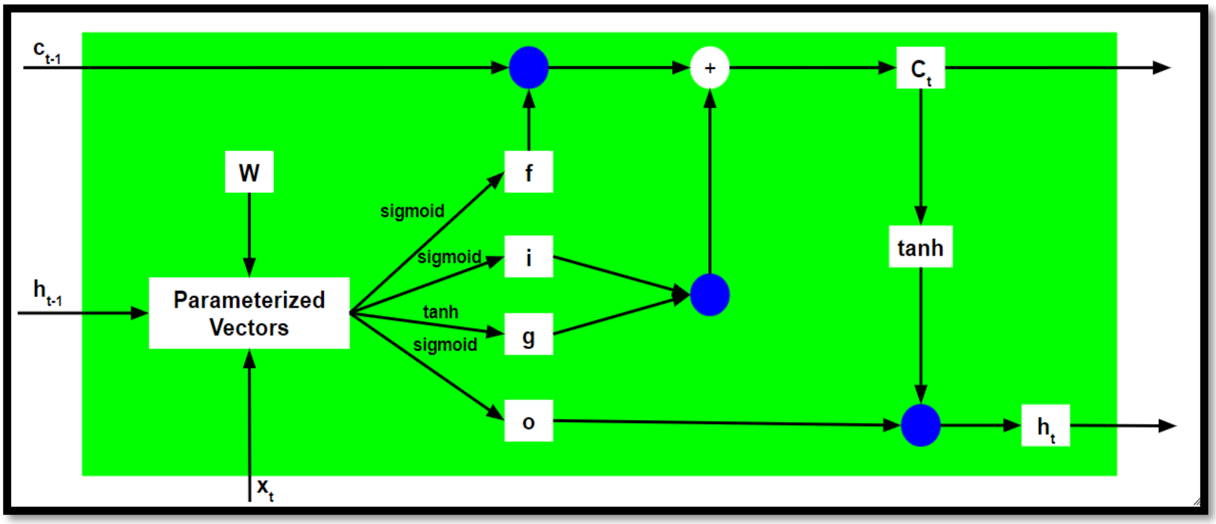


**Figure 3.7: Working Principle Of LSTM**

**In this project**, I have constructed a model with a 50-dimensional embedding layer to determine the aspect category and its polarity (embedding layer is created using GloVe). The LSTM layer, which contains 100 hidden neurons, is the next layer, followed by the Dense layer, which has sigmoid functions. The output will be in the form of a bag of words, with 1 in place of the aspect category and its polarity (as both are added) or 0 in lieu of the aspect category and its polarity.

## *3.2.2.3 CONVOLUTIONAL NEURAL NETWORK MODEL:*

In Convolutional Neural Network, variations of multilayer perceptron are used, and here the connection between the nodes does not form a cycle. It is a class of deep neural networks. Convolutional Neural Networks' applications mainly revolve around visual classification like face recognition, object detection, etc. But it proved to be good enough for performing Natural Language Processing tasks.

Convolutional Neural Network has convolutional and down sampling layers. The neurons in the convolutional layer can scan inputs for patterns, and Down sampling layers reduce the feature map dimensionality, which eventually improves actual performance. Down sampling layers are placed after convolutional layers in a ConvNet or Convolutional Neural Network, and after that, one or more multilayer perceptrons are placed.

The words in a text can be represented as vectors in a vector space based upon the entire vocabulary. In-text classification with Convolutional Neural Network, the sequential data is processed by CNN, so the one-dimensional convolutions are used. It can identify patterns in a text by changing the size of kernels and merging their outputs, and the output of each convolution is fired when a particular pattern is found. That is how patterns can be identified in a sentence, even if they are present anywhere.

**In this project**, 1D convolutional layer of CNN model is used. The layer is followed by the Max-pooling Layer of size 2. The number of filters is set to be 64 with kernel size 5 and Rectified Liner Unit activation function. The layer of the model is Dense layer of 20 dimension with sigmoid activation function i.e. the output will be in the form of a bag of words, with 1 in place of the aspect category and its polarity (as both are added) or 0 in lieu of the aspect category and its polarity.

## *3.3 WORD EMBEDDINGS*:

Word Embedding is a word representation that allows having words with similar meanings to have similar representations. Word embedding is a numeric input vector representing the words in a dimensional space with approximate meaning. It extracts all the features from the text fed into the machine learning model as input to process the text data. It helps in reducing the dimensionality and capturing the word having a similar meaning.

The different approaches of word embedding used in this project are: GloVe and Bag of Words.

### *3.3.1 GloVe:*

GloVe stands for the Global Vectors for word embedding. It is an unsupervised learning algorithm used for obtaining vector representations for words. the basic rule of GloVe is that it uses statistics to find the link between the words. The co-occurrence matrix shows the frequency of word pairs appearing frequently.

Example:

Corpus: The food was very tasty and the room was clean.

Room was not clean.

| | THE | FOOD | WAS | VERY | TASTY | AND | ROOM | CLEAN | NOT |
|---|---|---|---|---|---|---|---|---|---|
| THE | 0 | | | | | | | | |
| FOOD | 1 | 0 | | | | | | | |
| WAS | $\frac{1}{2}+\frac{1}{2}$ | 1 | 0 | | | | | | |
| VERY | $\frac{1}{3}$ | $\frac{1}{2}$ | 1 | 0 | | | | | |
| TASTY | $\frac{1}{4}$ | $\frac{1}{3}$ | $\frac{1}{2}$ | 1 | 0 | | | | |
| AND | 1 | $\frac{1}{4}$ | $\frac{1}{3}+\frac{1}{3}$ | $\frac{1}{2}$ | 1 | 0 | | | |
| ROOM | 1 | $\frac{1}{6}$ | $\frac{1}{5}+1+1$ | $\frac{1}{4}$ | $\frac{1}{3}$ | $\frac{1}{2}$ | 0 | | |
| CLEAN | $\frac{1}{3}+\frac{1}{9}$ | $\frac{1}{8}$ | $1+\frac{1}{7}+\frac{1}{2}$ | $\frac{1}{6}$ | $\frac{1}{5}$ | $\frac{1}{4}$ | $\frac{1}{2}+\frac{1}{3}$ | 0 | |
| NOT | 0 | 0 | 1 | 0 | 0 | 0 | $\frac{1}{2}$ | 1 | 0 |

**TABLE 3.1: GloVe REPRESENTATION**

Here, we create a co-occurrence matrix for each word with each other word in the corpus. The word which occurs next to each other gets the value 1, the word which occurs one word apart gets the value **1/2**, word which occur two word apart get the value **1/3** and so on.

The upper matrix will be the reflection of lower matrix. Initially, they are assigned randomly, and then we take two vectors and compare them in the space. If they have a high value in the co-occurrence matrix, i.e. highly frequent and far from each other, they are brought close. And if two vectors are closed to each other but less frequent, they are moved apart from each other. After many iterations, we get the vector representation approximating the co-occurrence matrix.

## *3.3.2 BAG OF WORDS:*

A bag of words is a text modelling technique of Natural Language Processing. It is a very simple and flexible technique of extracting textual features from the corpus and converting them into numerical vectors.

The Bag of Words is the representation of text that notes the word's occurrences in the corpus. It creates a document term matrix where each column of the matrix represents the number of unique words present in the corpus, and each row represents the document/review of the corpus. The values in the matrix represent the occurrence of the word corresponding to the document. The sum of each column represents the total number of occurrences of the word in the corpus.

Example:

Review 1: The food was tasty.

Review 2: The room is not clean but the food was very tasty.

| | THE | FOOD | WAS | TASTY | ROOM | IS | NOT | CLEAN | VERY | BUT |
|---|---|---|---|---|---|---|---|---|---|---|
| REVIEW 1 | 1 | 1 | 1 | 1 | 0 | 0 | 0 | 0 | 0 | 0 |
| REVIEW 2 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 |
| TOTAL | 2 | 2 | 2 | 2 | 1 | 1 | 1 | 1 | 1 | 1 |

**TABLE 3.2: BAG OF WORD REPRESENTATION**

**In this project,** I have extracted the features for the supervised learning models for prediction of aspect category and its polarity. The features extracted are noun, adverb, adjective and verbs from the review after pre-processing. Nouns represents the aspects of the review and adverb/adjective represents its polarity. And using these features have created the bag of words models which is our input feature matrix for supervised learning models.

# 4. EXPERIMENT, RESULT AND ANALYSIS

## 4.1 CORPUS:

The dataset is the SemEval-2014 Restaurant Review for Aspect Based Sentiment Analysis. The dataset was in the xml format. After extraction each information from xml file, the dataset contains 5160 reviews. There are 5160 rows and five columns. The columns are ID, Review, Aspect, Aspect Category and Polarity. The Review column consists of a review of the Restaurant as stated by the customer. The Aspect_Category column has five unique categories, i.e. ambience, price, services, anecdotes/miscellaneous and food. The column Polarity tells the sentiment, i.e. positive, negative, conflict and neutral, of the corresponding aspect category. This corpus has 2023 unique reviews. We have used 70% of the dataset for the training purpose and 30% of the dataset for testing.

The dataset is as follows:

| Id | Review | Aspect | Aspect_Category | Polarity |
|---|---|---|---|---|
| 3121 | But the staff was so horrible to us. | {'staff': 'negativ e'}, | Service | negative |
| 2777 | To be completel y fair, the only r edeeming factor was the food, w hich was above average, but cou ldn't make up fo r all the other de ficiencies of Te odora." | {'food': 'positiv e'}, | Food | Positive |

| | | | | |
|---|---|---|---|---|
| 2777 | To be completel y fair, the only r edeeming factor was the food, w hich was above average, but cou ldn't make up fo r all the other de ficiencies of Te odora." | {'food': 'positiv e'}, | anecdotes/ miscellaneous | negative |
| 1634 | "The food is uni formly exceptio nal, with a very capable kitchen which will prou dly whip up wh atever you feel l ike eating, whet her it's on the m enu or not." | {'food': 'positiv e', 'kitchen': 'pos itive', 'menu': 'n eutral'} | Food | positive |
| 2846 | Not only was th e food outstandi ng, but the little 'perks' were gre at.", | {'food': 'positiv e', 'perks': 'posit ive'} | Food | positive |

**TABLE 3.3: DATASET USED IN EXPERIMENT**

# 4.2 SUPERVISED MODEL ALGORITHM:

In this, we have used Bag of Words approach for the feature selection procedure. Initially, I have imported all the libraries required for the experiment and then imported the X ml file. After importing, extraction of information required for Aspect based sentiment analysis was performed and stored them in a data frame and converting the data frame into pickle file for convenience.

After creating the dataset for the experiment, text pre-processing is carried out by removing all the punctuations, digits, stop words, emails and URLs, converting the review into lower case and lemmatization is done. Now extracted the POS tag for each word by tokenization and created a dictionary containing words as key and POS tag as value. Now, counted the frequency of each noun, adverb/adjective and verb words from the corpus

**(here, nouns are the aspects of the review, adjective/adverbs are the sentiments related to aspects)** and took top 500 words for each criterion i.e. noun adverb/adjective and verb for the formation of bag of words. This is our **Input Feature Matrix.**

For the label of supervised learning we created a column, Rating which ranges from 1 to 20. This range is given by concatenating the Apect_Category and Polarity columns, sorting them and store in dictionary with concatenated value as key and corresponding values ranging from 1 to 20 (Since we have five unique aspect category and each category has 4 possible polarity). Assigning the values of dictionary according to each review's aspect category and polarity to the new column, Rating.

Split the input feature matrix created and Rating column in the ratio 7:3, i.e. 70% training set and 30% testing set. Train the Support Vector Machine model and Naïve Bayes Model and predict the Rating. Note down the accuracy, precision, recall and F1 score measuring metrics.

## *4.3 DEEP LEARNING MODEL ALGORITHM:*

In this, we have used GloVe word embedding approach for the feature extraction. I began by importing all of the libraries needed for the experiment, followed by the X ml file. Following import, data for Aspect-based sentiment analysis was extracted and stored in a data frame, which was then converted to a pickle file for convenience.

**For Output Feature matrix**, I concatenated the Apect_Category and Polarity columns, sort them and store in list. Now created another dataframe,Y having the columns name as the items in the list. Now created the bag of words model by assigning 1 to the column the columns if the column name of Y is equal to name formed by concatenation of Aspect_Category and polarity for each review else assign 0. This feature matrix is of size $2023 \times 20$ where 2023 is the number of unique reviews in the dataset.

**For Input Feature matrix**, I created a corpus of unique reviews. Then mapped each word from the corpus and represented with their indexes and sequenced it as each review and pre-padded the sequence of embedding as it takes the definite size vector matrix. Maximum length of the sentence is taken for pre-padding. This feature matrix is of size $2023 \times 34$.

Then created an Embedding dictionary with the pretrained GloVe- 50d (50 dimension) where word is the key and corresponding array of values from GloVe -50 is the value of embedding dictionary and creating embedding matrix of 50 dimension for each word where values of the matrix are the values from the embedding dictionary if the word is present. This matrix is used in the Embedding layer while creating the deep learning models.

Splitting the input feature matrix and output feature matrix in the ratio 7:3, i.e. 70% training set and 30% testing set. Create the models: LSTM, RNN and CNN. Train the model and predict the aspect category and corresponding polarity. Note down the accuracy, precision, recall and F1 score measuring metrics.

# *4.4 EVALUATION:*

## *4.4.1 RESULT OF SVM MODEL:*

In determining the aspect category and its respective polarity, we have trained the learning model with linear kernel, its classification performance in the form of following metric is shown below:

| METRICS | PERFORMANCE (%) |
|---|---|
| Accuracy score | 45% |
| Precision | 38.03% |
| Recall | 33.93% |
| F1-score | 35.21% |

**TABLE 3.4: PERFORMANCE TABLE OF SVM**

## *4.4.2 RESULT OF NAÏVE BAYES MODEL:*

In determining the aspect category and its respective polarity, we have trained Multinomial Naïve Bayes learning model (one of the types of Naïve Bayes). Its classification performance in the form of following metric is shown below:

| METRICS | PERFORMANCE (%) |
|---|---|
| Accuracy score | 45% |
| Precision | 36.84% |
| Recall | 26.55% |
| F1-score | 28.28% |

**TABLE 3.5: PERFORMANCE TABLE OF NAÏVE BAYES CLASSIFIER**

## *4.4.3 RESULT OF LSTM MODEL:*

In determining the aspect category and its respective polarity, we have trained the LSTM model with epoch=80 and batch size =25. The training accuracy noted was 97% whereas test accuracy noted was 86%. The model was trained 5 times due to slight fluctuations in accuracy. Therefore, the final accuracy noted was the average of accuracies obtained. The performance of the model in the form of following metrics is shown below:

| METRICS | PERFORMANCE (%) |
|---|---|
| Accuracy score | 86% |
| Precision | 96.55% |
| Recall | 81.42% |
| F1-score | 87.5% |

**TABLE 3.6: PERFORMANCE TABLE OF LSTM**

## *4.4.4 RESULT OF RNN MODEL:*

In determining the aspect category and its respective polarity, we have trained the RNN model with epoch=80 and batch size =25. The training accuracy noted was 86.44% whereas test accuracy noted was 73%. The model was trained 5 times due to slight fluctuations in accuracy. Therefore, the final accuracy noted was the average of accuracies obtained. The performance of the model in the form of following metrics is shown below:

| METRICS | PERFORMANCE (%) |
|---|---|
| Accuracy score | 73% |
| Precision | 86.95% |
| Recall | 68.35% |
| F1-score | 74.62% |

**TABLE 3.7: PERFORMANCE TABLE OF RNN**

## *4.4.5 RESULT OF CNN MODEL:*

Since we have sequential textual data, we are using one dimensional convolution. In determining the aspect category and its respective polarity, we have trained the CNN model with epoch=80 and batch size =25. The training accuracy noted was 99% whereas test accuracy noted was 85.2%. The model was trained 5 times due to slight fluctuations in accuracy. Therefore, the final accuracy noted was the average of accuracies obtained. The performance of the model in the form of following metrics is shown below:

| METRICS | PERFORMANCE (%) |
|---|---|
| Accuracy score | 85.2% |
| Precision | 90.47% |
| Recall | 82.92% |
| F1-score | 86.26% |

**TABLE 3.8: PERFORMANCE TABLE OF CNN**

## *4.5 ANALYSIS:*

From the above performance tables, we can observe that the performance of supervised learning models is very low in comparison to the deep learning models. I noted that the accuracy of supervised models, SVM and Naïve Bayes is 45% whereas the accuracies of deep learning models, LSTM, RNN and CNN are 86%,73% and 85.2% respectively. This may be due to the feature selection process. In supervised learning analyze the manually selected subset of features to determine the target whereas in Deep Neural Network, new features are emerged and the unwanted features are discarded during the learning process. In supervised learning, the features selected manually is top 500 noun, adjective/adverb and verb words from the corpus, and model is trained over it whereas in deep learning the important features are learned by the model itself during learning process. We also observe that the accuracy calculated by the deep learning models are taken as the average of accuracies (accuracies obtained by training the model 5 times), this is due fluctuations in accuracy as the data is very less in number whereas there are no fluctuations in supervised learning.

# 5. CONCLUSION

Based on several tests on different models of supervised learning, SVM and Naïve Bayes and Deep learning models, LSTM, RNN and CNN, it is concluded that the deep learning models performed quiet well in comparison to supervised learning models due to feature selection. We also observe that LSTM has highest performance with 86% accuracy which is better than Simple RNN whose accuracy is 73% because LSTM suffers much less from vanishing gradient problem (an effect which is similar to what is observed with non-recurrent networks which have many layers deep and when we keep adding layers to it the network becomes untrainable). We have also noted down that CNN is more powerful than Simple RNN on comparing their accuracies and other measures too. This may be due to CNN's feed-forward neural network using filters and pooling whereas RNN send the output back to the network. In comparison between supervised learning model, SVM and Naïve Bayes model, we see that the accuracy is same for both but the F1-measure and, precision and recall have difference. F1_score is considered to be more accurate metrics than accuracy because F1 is balancing the recall and precision on positive class whereas Accuracy looks at correctly classified observations positive and negative. It also concludes that SVM is better classifier than Naïve Bayes in this aspect as F1_score of SVM is higher than Naïve Bayes.

# REFERENCES


1. Caroline Brun, Diana Nicoleta Popa, and Claude Roux. 2014. XRCE: Hybrid Classification for Aspect-based Sentiment Analysis. In *Proceedings of the 8th International Workshop on Semantic Evaluation (SemEval 2014)*, pages 838–842, Dublin, Ireland. Association for Computational Linguistics.
2. Samuel Brody and Noemie Elhadad. 2010. An unsupervised aspect-sentiment model for online reviews. In Proceedings of NAACL, pages 804–812, Los Angeles, California.
3. S. Kiritchenko, X. Zhu, C. Cherry, and S. Mohammad, "NRC-Canada-2014: Detecting aspects and sentiment in customer reviews," in Proceedings of the 8th International Workshop on Semantic Evaluation (SemEval 2014) (Association for Computational Linguistics and Dublin City University, Dublin, Ireland, 2014), pp. 437–442.
4. T. Brychcin, M. Konkol, and J. Steinberger, "UWB: Machine learning approach to aspect-based sentiment analysis," in Proceedings of the 8th International Workshop on Semantic Evaluation (SemEval 2014) (Association for Computational Linguistics and Dublin City University, Dublin, Ireland, 2014), pp. 817–822.
5. Mubarok, Mohamad Syahrul & Adiwijaya, Kang & Aldhi, Muhammad. (2017). Aspect-based sentiment analysis to review products using Naïve Bayes. AIP Conference Proceedings. 1867. 020060. 10.1063/1.4994463.
6. Minqing Hu and Bing Liu. 2004. Mining and summarizing customer reviews. In Proceedings of the tenth ACM SIGKDD international conference on Knowledge discovery and data mining (KDD '04). Association for Computing Machinery, New York, NY, USA, 168–177. https://doi.org/10.1145/1014052.1014073
7. Soo-Min Kim and Eduard Hovy. 2004. Determining the sentiment of opinions. In Proceedings of the 20th international conference on Computational Linguistics (COLING '04). Association for Computational Linguistics, USA, 1367–es. https://doi.org/10.3115/1220355.1220555
8. Pang, Bo, Lillian Lee, and Shivakumar Vaithyanathan. 2002. Thumbs up?: Sentiment classification using machine learning techniques. In EMNLP '02: Proc. of the

conference on Empirical methods in natural language processing. ACL, Morristown, NJ, USA, pages 79–86.

9. Ganu, Gayatree, Noemie Elhadad, and Amelie Marian. 2009. Beyond the stars: Improving rating predictions using review text content. In WebDB.
10. Ana-Maria Popescu and Oren Etzioni. 2005. Extracting Product Features and Opinions from Reviews. In *Proceedings of Human Language Technology Conference and Conference on Empirical Methods in Natural Language Processing*, pages 339–346, Vancouver, British Columbia, Canada. Association for Computational Linguistics.
11. P. D. Turney. 2002. Thumbs up or thumbs down? semantic orientation applied to unsupervised classification of reviews. In Procs. of the 40th Annual Meeting of the Association for Computational Linguistics (ACL'02), pages 417–424.