##### A Project report on

**Hotel Review Analysis For The Prediction Of Business Using Deep Learning Approach**

###### A Dissertation submitted to JNTU Hyderabad in partial fulfillment of the academic requirements for the award of the degree.

**Bachelor of Technology**

**in**

**Computer Science and Engineering**

Submitted by

PODUTURI SAVAN REDDY

(19H51A05E4)

PULI SHABARISH

(19H51A05H8)

NAKKALA THOMAS REDDY

(19H51A0550)

Under the esteemed guidance of

Ms. P. Sravanthi

(Assistant Professor)



**Department of Computer Science and Engineering**

**CMR COLLEGE OF ENGINEERING & TECHNOLOGY**

(UGC Autonomous)

\*Approved by AICTE \*Affiliated to JNTUH \*NAAC Accredited with A+ Grade

KANDLAKOYA, MEDCHAL ROAD, HYDERABAD - 501401.

2019- 2023

**CMR COLLEGE OF ENGINEERING & TECHNOLOGY**

KANDLAKOYA, MEDCHAL ROAD, HYDERABAD – 501401

**DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING**



#### CERTIFICATE

This is to certify that the Major Project report entitled **"Hotel Review Analysis For The Prediction Of Business Using Deep Learning Approach"** being submitted by Poduturi Savan Reddy (19H51A05E4), Puli Shabarish (19H51A05H8), Nakkala Thomas Reddy (19H51A0550) in partial fulfillment for the award of **Bachelor of Technology in Computer Science and Engineering** is a record of bonafide work carried out her under my guidance and supervision.

###### The results embodies in this project report have not been submitted to any other University or Institute for the award of any Degree.

**Ms. P. Sravanthi Dr. Siva Skandha Sanagala**

**Assistant Professor Associate Professor and HOD**

**Dept. of CSE Dept. of CSE**

#### ACKNOWLEDGEMENT

With great pleasure we want to take this opportunity to express my heartfelt gratitude to all the people who helped in making this project work a grand success.

We are grateful to **Ms. P. Sravanthi, Assistant Professor**, Department of Computer Science and Engineering for her valuable technical suggestions and guidance during the execution of this project work.

We would like to thank **Dr. Siva Skandha Sanagala,** Head of the Department of Computer Science and Engineering, CMR College of Engineering and Technology, who is the major driving forces to complete my project work successfully.

We are very grateful to **Dr. Vijaya Kumar Koppula**, Dean-Academics, CMR College of Engineering and Technology, for his constant support and motivation in carrying out the project work successfully.

We are highly indebted to **Major Dr. V A Narayana,** Principal, CMR College of Engineering and Technology, for giving permission to carry out this project in a successful and fruitful way.

We would like to thank the **Teaching & Non- teaching** staff of Department of Computer Science and Engineering for their co-operation

We express our sincere thanks to **Shri. Ch. Gopal Reddy**, Secretary, CMR Group of Institutions, for his continuous care.

Finally, We extend thanks to our parents who stood behind us at different stages of this Project. We sincerely acknowledge and thank all those who gave support directly and indirectly in completion of this project work.

Poduturi Savan Reddy 19H51A05E4

Puli Shabarish 19H51A05H8

Nakkala Thomas Reddy 19H51A0550

Hotel Review Analysis For The Prediction Of Business Using Deep Learning Approach

**TABLE OF CONTENTS**

**CHAPTER**

**NO. TITLE PAGE NO.**

LIST OF FIGURES ii

LIST OF TABLES iii

ABSTRACT iv

**1** **INTRODUCTION** **2**

1.1 Problem Statement 3

1.2 Research Objective 3

1.3 Project Scope and Limitations 4

**2** **BACKGROUND WORK** **5**

2.1. Naïve Bayes Bernoulli Classifier 6

2.1.1.Introduction 6

2.1.2.Merits,Demerits and Challenges 6

2.1.3.Implementation of Naïve bayes model 9

2.2. SVM 13

2.2.1.Introduction 13

2.2.2.Merits,Demerits and Challenges 14

2.2.3.Implementation of SVM Model 16

2.3. Decision Tree Classifier 19

2.3.1.Introduction 19

2.3.2.Merits,Demerits and Challenges 21

2.3.3.Implementation of Decision Tree Classifier 22

**3 PROPOSED SYSTEM 24**

3.1. Objective of Proposed Model 25

3.2. Algorithms Used for Proposed Model 27

3.3. Designing 34

3.3.1.UML Diagram 34

3.3.2.Architecture Diagram 35

3.3.3.Data Flow Diagram 36

3.3.4.Class Diagram 37

3.4. Stepwise Implementation and Code 38

3.4.1 Stepwise Implementation of LSTM 38

3.4.2 Code of LSTM 40

**4 RESULTS AND DISCUSSION 44** 4.1 Performance metrics 45

4.2 Comparison of Existing Solutions 45

4.3 Data Collection and Performance metrics 46

4.3.1 Data Collection 46

**5** **CONCLUSION 47**

5.1 Conclusion and Future Enhancement 448

**REFERENCES** **49**

**GitHub Link 51**

CMRCET B. Tech (CSE) Page No ii

Hotel Review Analysis For The Prediction Of Business Using Deep Learning Approach

**List of Figures**

**FIGURE**

**NO. TITLE PAGE NO.**

2.2.1.1 Graph of hyper plane 13

2.2.3.1 SVM algorithm 15

2.3.3.1 Architecture of Decision Tree algorithm 21

3.1 LSTM Gate Structure 26

3.2.2 Visualization Of LSTM 33

3.3.1 UML Diagram 34

3.3.2 Architecture Diagram 35

3.3.3 Data Flow Diagram 36

3.3.4 Class Diagram 37

CMRCET B. Tech (CSE) Page No ii

Hotel Review Analysis For The Prediction Of Business Using Deep Learning Approach

**List of Tables**

**FIGURE**

**NO. TITLE PAGE NO.**

4.1.1 Performance metrics of each classifier on test data 46

CMRCET B. Tech (CSE) Page No iii

Hotel Review Analysis For The Prediction Of Business Using Deep Learning Approach

**ABSTRACT**

Sentiment analysis is a widely used topic in Natural Language Processing that allows identifying the opinions or sentiments from a given text. Social media is the scope for the customers to share their opinion over the products or services as part of customer reviews. Dissect this review has become an important factor for business analysis since online business is exponentially growing in today’s techno-friendly competitive market. A large number of algorithms have been found in recent articles. Among those deep learning is an important approach. In the proposed methodology, long short-term memory (LSTM) and Gated recurrent units (GRUs) have been used to train the hotel review data where the accuracy rate of identifying customer opinion is 86%, and 84% respectively.

The dataset is also tested by using Naïve Bayes, Decision Tree, Random Forest, and SVM. For Naïve Bayes obtains an accuracy of 75%, for Decision Tree obtains an accuracy of 71%, for Random Forest the accuracy is 82% and for SVM our accuracy result is 71%. Deep learning is used to obtain better business performance and also get the review from customers and also to predict the sentiment about customer review. Our algorithm works properly and gives better accuracy.

CMRCET B. Tech (CSE) Page No iv

Hotel Review Analysis For The Prediction Of Business Using Deep Learning Approach

# **CHAPTER** **1**

**INTRODUCTION**

CMRCET B. Tech (CSE) Page No 1

Hotel Review Analysis For The Prediction Of Business Using Deep Learning Approach

**CHAPTER** **1**

**INTRODUCTION**

In the age of modern science, everything is based on online and on the internet. Internet-based shopping has become easier and more popular because of better quality, and fast logistic systems. Internet-based shopping and booking are very comfortable. People can easily make a booking without going outside. The most effective side part of online-based work is that people can give a review. Recognizing reviews allows others to easily understand the emotions of others and obtain the rationality result of different products.

In the hotel review, the prediction of business using Deep Learning was analysed. Many start-up businesses became failure due to lack of analysis and the sentiment of the customer. Sentiment Analysis is the most significant to improve a business site. Here, different type of data from social media as well as from the Hotel Management Website was collected using Unamo tools. And also some supervised and unsupervised data is used to predict the best result. This article will help to improve the business.

At present, online-based opinions can easily analysis with the help of Sentiment Analysis (SA). It is the management of sentiments, different opinions, subjective text, and different emoji used for giving reviews. People can easily get the comprehension information related to people reviews. Mainly Sentiment analysis is one kind of tool that helps to get the public sentiment. By capturing reviews of product or location or person might be found from a different internet-based site like Face book, Amazon. Sentiment Analysis is used to increase the requirement of analysing and structuring hidden information which comes from social media in the form of unstructured data. A huge amount of data is used due to the capability of automation and can handle a huge amount of data. A different type of font of review are further classified.

Neural networks are used to mimic the basic functioning of the human brain and are inspired by how the human brain interprets information. It is used to solve various real-time tasks because of its ability to perform computations quickly and its fast responses. Artificial Neural Network has a huge number of interconnected processing elements, also known as Nodes. These nodes are connected with other nodes using a connection link. The connection link contains weights, these weights contain the information about the input signal. Each iteration and input in turn leads to updating of these weights.

CMRCET B. Tech (CSE) Page No 2

Hotel Review Analysis For The Prediction Of Business Using Deep Learning Approach

* 1. **Problem statement**

If any business wants to sustain in the market for a longer period then their customer’s reviews are the key indicators for their business. With ever increasing hotels available to customers, the hotels are faced with an emerging need to analyse the customer needs and adapt accordingly. With Social Media and Review Platforms available , customers consider the reviews before booking a hotel. This project aims at solving the problem faced by hotels and provide an intelligent system that helps to analyse customer reviews and summarize them.

* 1. **Research objective**
* It can help in business.
* Gives accurate results.
* Can handle large data.
* Improving the business.
* Improving predicted classification.
* Unseen data will not impact the effectiveness of model considerably.
* Training Model faster.
* Provides Binary classification.
* Improve its performance of LSTM.
* Deeper understanding of how LSTMs work and why they are effective.
* LSTMs are often used as black-box models, meaning that it can be difficult to interpret how they arrive at their predictions.
* How it changes over time, as well as investigating how different hyperparameters affect performance.

CMRCET B. Tech (CSE) Page No 3

Hotel Review Analysis For The Prediction Of Business Using Deep Learning Approach

* 1. **Project** **Scope and Limitations**

**Project Scope**

* Our Problem falls under classification category.
* In our project, features in the database created for review websites are classified by determining the input and output parameters for the classification.
* Classification is to determine the class to which each data sample of the methods belongs, which methods are used when the outputs of input data are qualitative.
* The purpose is to divide the whole problem space into a certain number of classes. A wide range of classification methods are present. This is due to the fact that different classification methods have been constructed for different data as there is no perfect method that works on every data set. The method with highest accuracy for the entire data set is chosen.

**Limitations**

* Our Project can achieve an accuracy of 86%.
* Our Project can only classify data but cannot output what the customers like/dislike.
* Our project is only used to detect the Sentimentality of the review.
* Although LSTMs are designed to capture long-term dependencies, they can still have difficulty with sequences that are very long or have complex patterns.
* This can make them slow to train and difficult to deploy in real-world applications.
* LSTMs require large amounts of data to train effectively. This can be a challenge in domains where data is scarce or expensive to obtain, such as in medical imaging or scientific research.
* LSTMs require significant computational resources, especially when dealing with large datasets or complex models.

CMRCET B. Tech (CSE) Page No 4

Hotel Review Analysis For The Prediction Of Business Using Deep Learning Approach

**CHAPTER** **2**

**BACKGROUND** **WORK**

CMRCET B. Tech (CSE) Page No 5

Hotel Review Analysis For The Prediction Of Business Using Deep Learning Approach

**CHAPTER** **2**

**BACKGROUND** **WORK**

**2.1** **Review analysis using Naïve Bayes Bernoulli Classifier**

**2.1.1** **Introduction**

Naive Bayes Bernoulli is a binary independence model, which generates an indicator for each term of the vocabulary ,either 1 indicating presence of the term in the document or 0 indicating absence. Bernoulli model uses binary occurrence information, ignoring the number of occurrences whereas the multinomial model keeps track of multiple occurrences. It specifies that a review is represented by a vector of binary attributes

indicating which words appear in the review or not.

**2.1.2** **Merits, Demerits,** **And** **Challenges**

**Merits**

* It is easy and fast to predict class of test data set. It also perform well in multi class prediction
* When assumption of independence holds, a Naive Bayes classifier performs better compare to other models like logistic regression and you need less training data.

**Demerits**

* If categorical variable has a category (in test data set), which was not observed in training data set, then model will assign a 0 (zero) probability and will be unable to make a prediction. This is often known as “Zero Frequency”.
* Accuracy of Naïve bayes model is 75%.

**Challenges**

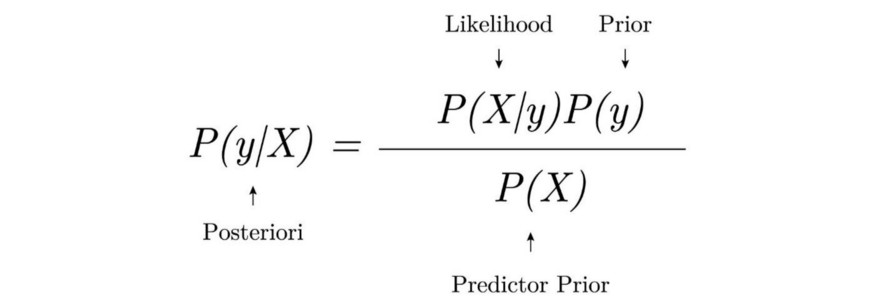
* Reviews may contain large number of unseen words and the model will not function as expected.

CMRCET B. Tech (CSE) Page No 6

Hotel Review Analysis For The Prediction Of Business Using Deep Learning Approach

**2.1.3. Implementation** **of** **Naïve bayes model**

* Assume that all features in data set are independent
* Choose the most appropriate dependent variable and split the data set into training, validation and testing.
* Build a frequency count of the training set and apply Naïve bayes on it



The Bayes’ theorem is used to determine the probability of a hypothesis when prior knowledge is available. It depends on conditional probabilities. The formula is given below :

Text

Description automatically generated with medium confidence

where P(A|B) is posterior probability i.e. the probability of a hypothesis A given the event B occurs. P(B|A) is likelihood probability i.e. the probability of the evidence given that hypothesis A is true. P(A) is prior probability i.e. the probability of the hypothesis before observing the evidence and P(B) is marginal probability i.e. the probability of the evidence

CMRCET B. Tech (CSE) Page No 7

Hotel Review Analysis For The Prediction Of Business Using Deep Learning Approach

When the Bayes’ theorem is applied to classify text documents, the class c of a particular document d is given by :

Text, letter

Description automatically generated

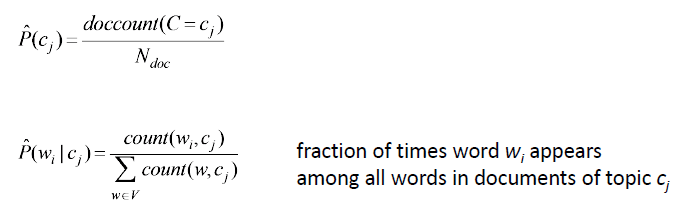
Where MAP is “maximum a posteriori” = most likely class.

Document d represented as features x1…xn.

Let the feature conditional probabilities P(x\_i | c) be independent of each other (conditional independence assumption). So,

P(x\_1, x\_2, …, x\_n | c) = P(x\_1 | c) X P(x\_2 | c) X … X P(x\_n | c)

Now, if we consider words as the features of the document, the individual feature conditional probabilities can be calculated using the following formula :



CMRCET B. Tech (CSE) Page No 8

Hotel Review Analysis For The Prediction Of Business Using Deep Learning Approach

**Code:**

import pandas as pd

import matplotlib.pyplot as plt

import plotly.express as px

from wordcloud import WordCloud

import nltk

import re

import string

from nltk.corpus import stopwords

nltk.download('punkt')

nltk.download('stopwords')

from nltk.tokenize import word\_tokenize

from nltk.stem import WordNetLemmatizer

stop\_words = stopwords.words()

df=pd.read\_csv('../input/imdb-dataset-of-50k-movie-reviews/IMDB Dataset.csv')

df.head()

df['sentiment'].value\_counts()

text = " ".join(i for i in df[df['sentiment']=='positive']['review'])

wordcloud = WordCloud( background\_color="white").generate(text)

plt.figure( figsize=(15,10))

plt.imshow(wordcloud, interpolation='bilinear')

plt.axis("off")

plt.title('wordcloud for positive review')

plt.show()

CMRCET B. Tech (CSE) Page No 9

Hotel Review Analysis For The Prediction Of Business Using Deep Learning Approach

df.rename(columns={'review':'text'}, inplace = True)

df

def cleaning(text):

# converting to lowercase, removing URL links, special characters, punctuations...

text = text.lower() # converting to lowercase

text = re.sub('https?://\S+|www\.\S+', '', text) # removing URL links

text = re.sub(r"\b\d+\b", "", text) # removing number

text = re.sub('<.\*?>+', '', text) # removing special characters,

text = re.sub('[%s]' % re.escape(string.punctuation), '', text) # punctuations

text = re.sub('\n', '', text)

text = re.sub('[’“”…]', '', text)

emoji\_pattern = re.compile("["

u"\U0001F600-\U0001F64F" # emoticons

u"\U0001F300-\U0001F5FF" # symbols & pictographs

u"\U0001F680-\U0001F6FF" # transport & map symbols

u"\U0001F1E0-\U0001F1FF" # flags (iOS)

u"\U00002702-\U000027B0"

u"\U000024C2-\U0001F251"

"]+", flags=re.UNICODE)

text = emoji\_pattern.sub(r'', text)

dt = df['text'].apply(cleaning)

dt['no\_sw'] = dt['text'].apply(lambda x: ' '.join([word for word in x.split() if word not in (stop\_words)]))

from collections import Counter

cnt = Counter()

CMRCET B. Tech (CSE) Page No 10

Hotel Review Analysis For The Prediction Of Business Using Deep Learning Approach

for text in dt["no\_sw"].values:

for word in text.split():

cnt[word] += 1

cnt.most\_common(10)

temp = pd.DataFrame(cnt.most\_common(10))

temp.columns=['word', 'count']

temp

px.bar(temp, x="count", y="word", title='Commmon Words in Text', orientation='h',

width=700, height=700)

FREQWORDS = set([w for (w, wc) in cnt.most\_common(10)])

def remove\_freqwords(text):

return " ".join([word for word in str(text).split() if word not in FREQWORDS])

dt["wo\_stopfreq"] = dt["no\_sw"].apply(lambda text: remove\_freqwords(text))

dt.head()

from sklearn.feature\_extraction.text import CountVectorizer

from nltk.tokenize import RegexpTokenizer

token = RegexpTokenizer(r'[a-zA-Z0-9]+')

cv = CountVectorizer(stop\_words='english',ngram\_range = (1,1),tokenizer = token.tokenize)

text\_counts = cv.fit\_transform(nb['review'])

from sklearn.model\_selection import train\_test\_split

X=text\_counts

y=nb['sentiment']

X\_train, X\_test, y\_train, y\_test = train\_test\_split(X, y, test\_size=0.20,random\_state=30)

from sklearn.naive\_bayes import ComplementNB

from sklearn.metrics import classification\_report, confusion\_matrix

CNB = ComplementNB()

CMRCET B. Tech (CSE) Page No 11

Hotel Review Analysis For The Prediction Of Business Using Deep Learning Approach

CNB.fit(X\_train, y\_train)

from sklearn import metrics

predicted = CNB.predict(X\_test)

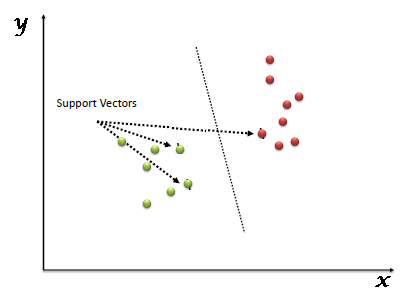
accuracy\_score = metrics.accuracy\_score(predicted, y\_test)

CMRCET B. Tech (CSE) Page No 12

Hotel Review Analysis For The Prediction Of Business Using Deep Learning Approach

**2.2 Review analysis using SVM**

**2.2.1** **Introduction**

****“Support Vector Machine” (SVM) is a supervised machine learning algorithm that can be used for both classification or regression challenges. However, it is mostly used in classification problems. In the SVM algorithm, we plot each data item as a point in n-dimensional space (where n is the number of features you have) with the value of each feature being the value of a particular coordinate. Then, we perform classification by finding the hyper-plane that differentiates the two classes very well (look at the below snapshot).

**Figure 2.2.1.1: Graph of hyper plane**

Support Vectors are simply the coordinates of individual observation. The SVM classifier is a frontier that best segregates the two classes (hyper-plane/ line). SVM can handle non-linearly separable data by using a kernel function that maps the original input space into a higher-dimensional feature space. This allows SVM to find a hyperplane that can separate the data points even when they are not linearly separable in the original input space. SVM tries to find the hyperplane that has the maximum margin between the closest data points from different classes.

CMRCET B. Tech (CSE) Page No 13

Hotel Review Analysis For The Prediction Of Business Using Deep Learning Approach

**2.2.2 Merits, Demerits,** **And** **Challenges**

**Merits**

* It works well with a clear margin of separation
* It is effective in high-dimensional spaces.
* It is effective in cases where the number of dimensions is greater than the number of samples.
* It uses a subset of training points in the decision function (called support vectors), so it is also memory efficient.

**Demerits**

* It doesn’t perform well when we have large data set because the required training time is higher
* It also doesn’t perform very well, when the data set has more noise i.e. target classes are overlapping
* SVM doesn’t directly provide probability estimates, these are calculated using an expensive five-fold cross-validation. It is included in the related SVC method of the Python scikit-learn library.

**Challenges**

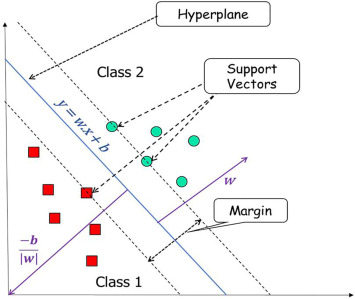
* Unbalanced data
* Multi-label classifications
* SVM cannot handle large data sets
* Semi-supervised learning.
* It is Complex to implement.
* Large Memory required to implement.

CMRCET B. Tech (CSE) Page No 14

Hotel Review Analysis For The Prediction Of Business Using Deep Learning Approach

**2.2.3 Implementation** **of** **SVM Model**

* Pre-process the data to remove stop words, unnecessary characters .
* Segment and normalize the data set.
* Extract the feature vector.
* Convert text to vector matrix
* Train SVMs based on the saved sample database.
* Analyze the Review by the set of SVMs trained in advance.
* If there are no more unclassified samples, then STOP. Otherwise, continue model training
* Add these test samples into their corresponding database for further training.



**Figure 2.2.3.1: SVM algorithm**

CMRCET B. Tech (CSE) Page No 15

Hotel Review Analysis For The Prediction Of Business Using Deep Learning Approach

**CODE:**

import numpy as np

import pandas as pd

from bs4 import BeautifulSoup

import matplotlib.pyplot as plt

import seaborn as sns

import nltk

from nltk.corpus import stopwords

from nltk.stem import SnowballStemmer

from nltk.tokenize import TweetTokenizer

from sklearn.feature\_extraction.text import CountVectorizer, TfidfTransformer

from sklearn.linear\_model import LogisticRegression

from sklearn.svm import SVC

from sklearn.model\_selection import train\_test\_split, StratifiedKFold, cross\_val\_score

from sklearn.pipeline import make\_pipeline, Pipeline

from sklearn.model\_selection import GridSearchCV

from sklearn.metrics import make\_scorer, accuracy\_score, f1\_score

from sklearn.metrics import roc\_curve, auc

from sklearn.metrics import confusion\_matrix, roc\_auc\_score, recall\_score, precision\_score

data = pd.read\_csv("../input/Tweets.csv")

data\_clean = data.copy()

data\_clean = data\_clean[data\_clean['airline\_sentiment\_confidence'] > 0.65]

data\_clean['sentiment'] = data\_clean['airline\_sentiment'].\

apply(lambda x: 1 if x=='negative' else 0)

CMRCET B. Tech (CSE) Page No 16

Hotel Review Analysis For The Prediction Of Business Using Deep Learning Approach

data\_clean['text\_clean'] = data\_clean['text'].apply(lambda x: BeautifulSoup(x, "lxml").text)

data\_clean['sentiment'] = data\_clean['airline\_sentiment'].apply(lambda x: 1 if x=='negative' else 0)

data\_clean = data\_clean.loc[:, ['text\_clean', 'sentiment']]

data\_clean.head()

train, test = train\_test\_split(data\_clean, test\_size=0.2, random\_state=1)

X\_train = train['text\_clean'].values

X\_test = test['text\_clean'].values

y\_train = train['sentiment']

y\_test = test['sentiment']

def tokenize(text):

tknzr = TweetTokenizer()

return tknzr.tokenize(text)

def stem(doc):

return (stemmer.stem(w) for w in analyzer(doc))

en\_stopwords = set(stopwords.words("english"))

vectorizer = CountVectorizer(

analyzer = 'word',

tokenizer = tokenize,

lowercase = True,

ngram\_range=(1, 1),

stop\_words = en\_stopwords)

kfolds = StratifiedKFold(n\_splits=5, shuffle=True, random\_state=1)

np.random.seed(1)

CMRCET B. Tech (CSE) Page No 17

Hotel Review Analysis For The Prediction Of Business Using Deep Learning Approach

pipeline\_svm = make\_pipeline(vectorizer,

SVC(probability=True, kernel="linear", class\_weight="balanced"))

grid\_svm = GridSearchCV(pipeline\_svm,

param\_grid = {'svc\_\_C': [0.01, 0.1, 1]},

cv = kfolds,

scoring="roc\_auc",

verbose=1,

n\_jobs=-1)

grid\_svm.fit(X\_train, y\_train)

grid\_svm.score(X\_test, y\_test)

grid\_svm.best\_params\_

def report\_results(model, X, y):

pred\_proba = model.predict\_proba(X)[:, 1]

pred = model.predict(X)

auc = roc\_auc\_score(y, pred\_proba)

acc = accuracy\_score(y, pred)

f1 = f1\_score(y, pred)

prec = precision\_score(y, pred)

rec = recall\_score(y, pred)

result = {'auc': auc, 'f1': f1, 'acc': acc, 'precision': prec, 'recall': rec}

return result

report\_results(grid\_svm.best\_estimator\_, X\_test, y\_test)def get\_roc\_curve(model, X, y):

pred\_proba = model.predict\_proba(X)[:, 1]

fpr, tpr, \_ = roc\_curve(y, pred\_proba)

return fpr, tpr

roc\_svm = get\_roc\_curve(grid\_svm.best\_estimator\_, X\_test, y\_test)

CMRCET B. Tech (CSE) Page No 18

Hotel Review Analysis For The Prediction Of Business Using Deep Learning Approach

**2.3 Review Analysis Using Decision Tree Classifier**

**2.3.1** **Introduction**

The main goal of the classification model for hotel reviews is to classify them as positive or negative, thus determining the opinion of the hotel guests expressed on the websites. Decision tree algorithm is used to generate a classification model for predicting the values of a target attribute (class or label) based on the values of several input attributes in the training data, used for classification of reviews. The attribute with the Label role is the target variable for prediction. It is used for classification of nominal and numeric data types. As a treelike model, it has root at the top and it grows downwards.

The Decision tree or subtree ends with leaf where a prediction about the target variable is made based on the conditions set forth. The decision tree is generated by recursive partitioning. In general, the recursion stops when all the examples have the same label value, i.e. the subset is pure, or if most of the examples are of the same label value, or when a certain condition is reached.

Information gain is selected for measuring the entropy and as a criterion for selecting the attribute for splitting the data. The attribute with the minimum entropy (the highest Information Gain) is selected for each split.

Decision trees are non-parametric, meaning that they do not make any assumptions about the underlying distribution of the data. This makes them suitable for a wide range of applications, including non-linear problems.

Overall, decision trees are a powerful and flexible algorithm that can be used in a wide range of applications. They are particularly useful when interpretability is important, or when the dataset has a mixture of categorical and continuous variables.

CMRCET B. Tech (CSE) Page No 19

Hotel Review Analysis For The Prediction Of Business Using Deep Learning Approach

* + 1. **Merits, Demerits,** **And** **Challenges**

**Merits**

* Compared to other algorithms decision trees requires less effort for data preparation during pre-processing.
* Missing values in the data also do NOT affect the process of building a decision tree to any considerable extent.

**Demerits**

* A small change in the data can cause a large change in the structure of the decision tree causing instability.
* Decision tree training is relatively expensive as the complexity and time has taken are more.

**Challenges**

* Large Data sets will not be evaluated properly thus leading to inaccurate results
* The training and evaluation of data is time consuming.
* Decision-tree learners can create over-complex trees that do not generalize the data well. This is called overfitting. Decision trees can be unstable because small variations in the data might result in a completely different tree being generated.

CMRCET B. Tech (CSE) Page No 20

Hotel Review Analysis For The Prediction Of Business Using Deep Learning Approach

* + 1. **Implementation Of Decision Tree Classifier**

While building a Decision tree, the main thing is to select the best attribute from the total features list of the dataset for the root node as well as for sub-nodes. The selection of best attributes is being achieved with the help of a technique known as the Attribute selection measure (ASM).

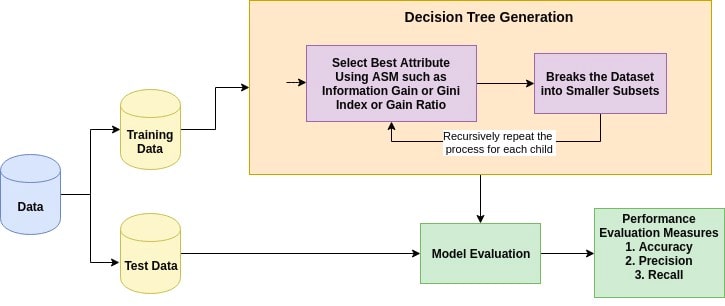
1.Select the best Feature using Attribute Selection Measures(ASM) to split the records.

2. Make that attribute/feature a decision node and break the dataset into smaller subsets.

3 Start the tree-building process by repeating this process recursively for each child until one of the following condition is being achieved :

a) All tuples belonging to the same attribute value.

b) There are no more of the attributes remaining.

c ) There are no more instances remaining.

**Figure 2.3.3.1 Architecture of Decision Tree algorithm**

CMRCET B. Tech (CSE) Page No 21

Hotel Review Analysis For The Prediction Of Business Using Deep Learning Approach

**CODE:**

from sklearn.feature\_extraction.text import TfidfVectorizer, CountVectorizer

from sklearn.metrics import confusion\_matrix

from sklearn.metrics import classification\_report

from sklearn.model\_selection import cross\_val\_score

from sklearn.model\_selection import cross\_val\_predict

from sklearn.model\_selection import train\_test\_split

from sklearn.svm import LinearSVC

from sklearn.tree import DecisionTreeClassifier

from sklearn import tree

with open("tickets.txt") as f:

tickets = f.read().strip().split("\n")

with open("labels\_4.txt") as f:

labels = f.read().strip().split("\n")

X\_train, X\_test, y\_train, y\_test = train\_test\_split(tickets, labels, test\_size=0.1, random\_state=1337)

vectorizer = CountVectorizer()

svm = LinearSVC()

X\_train = vectorizer.fit\_transform(X\_train)

X\_test = vectorizer.transform(X\_test)

\_ = svm.fit(X\_train, y\_train)

y\_pred = svm.predict(X\_test)

print(classification\_report(y\_test, y\_pred))

dt = DecisionTreeClassifier()

dt.fit(X\_train, y\_train)

y\_pred = dt.predict(X\_test)

CMRCET B. Tech (CSE) Page No 22

Hotel Review Analysis For The Prediction Of Business Using Deep Learning Approach

print(classification\_report(y\_test, y\_pred))

print(confusion\_matrix(y\_test, y\_pred))

from IPython.display import SVG

from graphviz import Source

from IPython.display import display

graph = Source(

tree.export\_graphviz(

dt,

out\_file=None,

feature\_names=vectorizer.get\_feature\_names(),

class\_names=['3', '1', '2', '0'],

filled = True)

)

display(SVG(graph.pipe(format='svg')))

tree.export\_graphviz(

dt,

out\_file="tree.dot",

feature\_names=vectorizer.get\_feature\_names(),

class\_names=dt.classes\_,

filled = True

)

dt.fit(X\_train, y\_train, max\_depth=4)

CMRCET B. Tech (CSE) Page No 23

Hotel Review Analysis For The Prediction Of Business Using Deep Learning Approach

**CHAPTER** **3**

**PROPOSED SYSTEM**

CMRCET B. Tech (CSE) Page No 24

Hotel Review Analysis For The Prediction Of Business Using Deep Learning Approach

**3.1 Objective of Proposed Model**

The objective of using LSTM (Long Short-Term Memory) is to effectively model and process sequential data by accounting for long-term dependencies and temporal patterns.

Unlike traditional feedforward neural networks, LSTMs can maintain an internal memory state that allows them to selectively remember or forget information from previous inputs. This ability makes LSTMs particularly useful for processing time-series data, natural language processing, speech recognition, and other tasks where sequential data is involved.

LSTMs can help capture the underlying structure of sequential data and make accurate predictions or generate new sequences. Therefore, LSTMs are commonly used in applications such as text prediction, machine translation, sentiment analysis, speech recognition, and more.

LSTM (Long Short-Term Memory) and RNN (Recurrent Neural Network) are both types of neural networks that are commonly used for sequential data analysis. However, LSTM has some advantages over traditional RNNs, which make it a preferred choice for some applications.

One of the main advantages of LSTM over RNN is its ability to handle long-term dependencies in sequential data. In traditional RNNs, the gradients tend to vanish or explode as they propagate through time, making it difficult for the network to learn long-term dependencies. LSTM solves this problem by introducing a gating mechanism that selectively retains or forgets information over extended periods of time, making it more effective in modelling long-term dependencies.

CMRCET B. Tech (CSE) Page No 25

Hotel Review Analysis For The Prediction Of Business Using Deep Learning Approach

Additionally, LSTMs have a cell state that allows them to preserve information over time, which makes them more effective in handling sequential data with variable-length input sequences. In contrast, traditional RNNs only have a hidden state, which is updated at each time step and may not be able to preserve information over a long period.

Furthermore, LSTMs are more robust to noisy data and can handle missing data more effectively than traditional RNNs. This is because the gating mechanism in LSTMs allows them to selectively filter out noise and fill in missing information, making them more reliable in handling real-world data.

LSTMs effectively store and access long-term dependencies using a special type of memory cell and gates. GRUs, a simplified version of LSTMs, use a single “update gate” and are easier to train and run, but may not handle long-term dependencies as well. The best type of RNN depends on the task at hand. It is often useful to try multiple types and see which performs best.

Diagram

Description automatically generated

**Figure 3.1: LSTM Gate Structure**

CMRCET B. Tech (CSE) Page No 26

Hotel Review Analysis For The Prediction Of Business Using Deep Learning Approach

**3.2 Algorithms Used for Proposed Model**

**3.2.1 Implementation of LSTM model**

There are 5 steps in the LSTM model

1. Define Network
2. Compile Network
3. Fit Network
4. Evaluate Network
5. Make Predictions

## Step 1. Define Network

The first step is to define your network.

Neural networks are defined in Kera’s as a sequence of layers. The container for these layers is the Sequential class

The first step is to create an instance of the Sequential class. Then you can create your layers and add them in the order that they should be connected. The LSTM recurrent layer comprised of memory units is called LSTM(). A fully connected layer that often follows LSTM layers and is used for outputting a prediction is called Dense().

But we can also do this in one step by creating an array of layers and passing it to the constructor of the Sequential.

* **Samples**. These are the rows in your data.
* **Timesteps**. These are the past observations for a feature, such as lag variables.
* **Features**. These are columns in your data.

CMRCET B. Tech (CSE) Page No 27

Hotel Review Analysis For The Prediction Of Business Using Deep Learning Approach

Assuming your data is loaded as a NumPy array, you can convert a 2D dataset to a 3D dataset using the reshape() function in NumPy. If you would like columns to become timesteps for one feature, you can use:

You can specify the input\_shape argument that expects a tuple containing the number of timesteps and the number of features. For example, if we had two timesteps and one feature for a univariate time series with two lag observations per row, it would be specified as follows:

LSTM layers can be stacked by adding them to the Sequential model. Importantly, when stacking LSTM layers, we must output a sequence rather than a single value for each input so that the subsequent LSTM layer can have the required 3D input. We can do this by setting the return\_sequences argument to True. For example:

This is a helpful container in Keras as concerns that were traditionally associated with a layer can also be split out and added as separate layers, clearly showing their role in the transform of data from input to prediction.

## Step 2: Compline Network

Once we have defined our network, we must compile it.

Compilation is an efficiency step. It transforms the simple sequence of layers that we defined into a highly efficient series of matrix transforms in a format intended to be executed on your GPU or CPU, depending on how Keras is configured.

Think of compilation as a precompute step for your network. It is always required after defining a model.

CMRCET B. Tech (CSE) Page No 28

Hotel Review Analysis For The Prediction Of Business Using Deep Learning Approach

Think of compilation as a precompute step for your network. It is always required after defining a model.

Compilation requires a number of parameters to be specified, specifically tailored to training your network. Specifically, the optimization algorithm to use to train the network and the loss function used to evaluate the network that is minimized by the optimization algorithm.

Alternately, the optimizer can be created and configured before being provided as an argument to the compilation step

The most common optimization algorithm is stochastic gradient descent, but Keras also supports a suite of other state-of-the-art optimization algorithms that work well with little or no configuration.

Perhaps the most commonly used optimization algorithms because of their generally better performance are:

Finally, you can also specify metrics to collect while fitting your model in addition to the loss function. Generally, the most useful additional metric to collect is accuracy for classification problems. The metrics to collect are specified by name in an array

## Step 3: Fit Network

Once the network is compiled, it can be fit, which means adapt the weights on a training dataset.

Fitting the network requires the training data to be specified, both a matrix of input patterns, X, and an array of matching output patterns, y.

The network is trained using the backpropagation algorithm and optimized according to the optimization algorithm and loss function specified when compiling the model.

CMRCET B. Tech (CSE) Page No 29

Hotel Review Analysis For The Prediction Of Business Using Deep Learning Approach

The backpropagation algorithm requires that the network be trained for a specified number of epochs or exposures to the training dataset.

Each epoch can be partitioned into groups of input-output pattern pairs called batches. This defines the number of patterns that the network is exposed to before the weights are updated within an epoch. It is also an efficiency optimization, ensuring that not too many input patterns are loaded into memory at a time.

Once fit, a history object is returned that provides a summary of the performance of the model during training. This includes both the loss and any additional metrics specified when compiling the model, recorded each epoch.

Training can take a long time, from seconds to hours to days depending on the size of the network and the size of the training data.

By default, a progress bar is displayed on the command line for each epoch. This may create too much noise for you, or may cause problems for your environment, such as if you are in an interactive notebook or IDE.

## Step 4: Evaluation Network

Once the network is trained, it can be evaluated.

The network can be evaluated on the training data, but this will not provide a useful indication of the performance of the network as a predictive model, as it has seen all of this data before.

We can evaluate the performance of the network on a separate dataset, unseen during testing. This will provide an estimate of the performance of the network at making predictions for unseen data in the future.

CMRCET B. Tech (CSE) Page No 30

Hotel Review Analysis For The Prediction Of Business Using Deep Learning Approach

The model evaluates the loss across all of the test patterns, as well as any other metrics specified when the model was compiled, like classification accuracy. A list of evaluation metrics is returned.

As with fitting the network, verbose output is provided to give an idea of the progress of evaluating the model. We can turn this off by setting the verbose argument to 0.

## Step 5: Make Prediction

Once we are satisfied with the performance of our fit model, we can use it to make predictions on new data. This is as easy as calling the predict() function on the model with an array of new input patterns.

The predictions will be returned in the format provided by the output layer of the network. In the case of a regression problem, these predictions may be in the format of the problem directly, provided by a linear activation function.

For a binary classification problem, the predictions may be an array of probabilities for the first class that can be converted to a 1 or 0 by rounding. For a multiclass classification problem, the results may be in the form of an array of probabilities (assuming a one hot encoded output variable) that may need to be converted to a single class output prediction using the argmax() NumPy function.

Alternately, for classification problems, we can use the predict\_classes() function that will automatically convert uncrisp predictions to crisp integer class values. As with fitting and evaluating the network, verbose output is provided to given an idea of the progress of the model making predictions. We can turn this off by setting the verbose argument to 0.

CMRCET B. Tech (CSE) Page No 31

Hotel Review Analysis For The Prediction Of Business Using Deep Learning Approach

**3.2.2 Internal execution of LSTM model**

LSTMs have many variations, but we’ll stick to a simple one. One cell consists of three gates (input, forget, output), and a cell unit. Gates use a sigmoid activation, while input and cell state is often transformed with tanh. LSTM cell can be defined with a following set of equations:

Gates:

it=g(Wxixt+Whiht−1+bi)it=g(Wxixt+Whiht−1+bi)

ft=g(Wxfxt+Whfht−1+bf)ft=g(Wxfxt+Whfht−1+bf)

ot=g(Wxoxt+Whoht−1+bo)ot=g(Wxoxt+Whoht−1+bo)

Input transform:

c\_int=tanh(Wxcxt+Whcht−1+bc\_in)

c\_int=tanh(Wxcxt+Whcht−1+bc\_in)

State update:

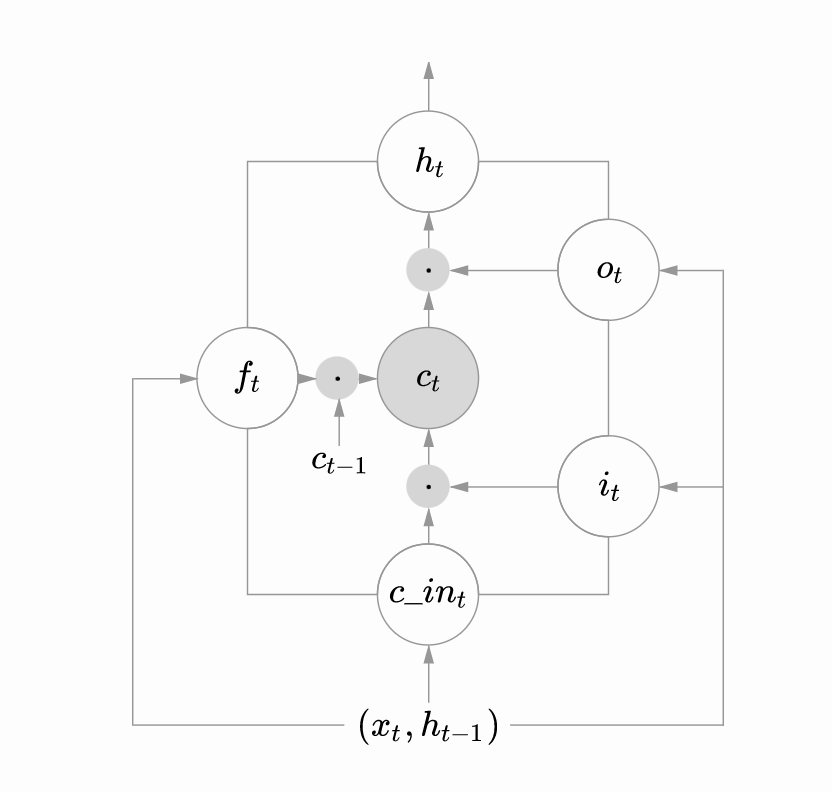
ct=ft⋅ct−1+it⋅c\_intct=ft⋅ct−1+it⋅c\_int

ht=ot⋅tanh(ct)

CMRCET B. Tech (CSE) Page No 32

Hotel Review Analysis For The Prediction Of Business Using Deep Learning Approach

**3.2.2 Internal visualization of LSTM model gates**



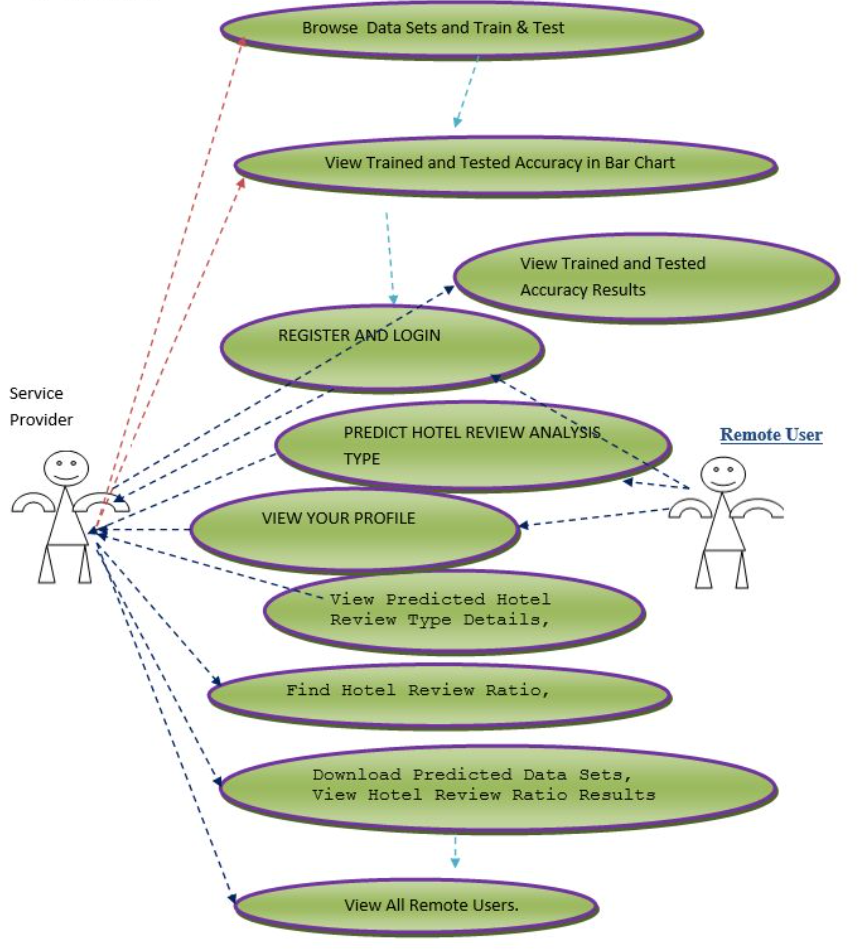
**Figure 3.2.2: Visualization Of LSTM**

CMRCET B. Tech (CSE) Page No 33

Hotel Review Analysis For The Prediction Of Business Using Deep Learning Approach

**3.3 Designing**

**3.3.1 UML Diagram**

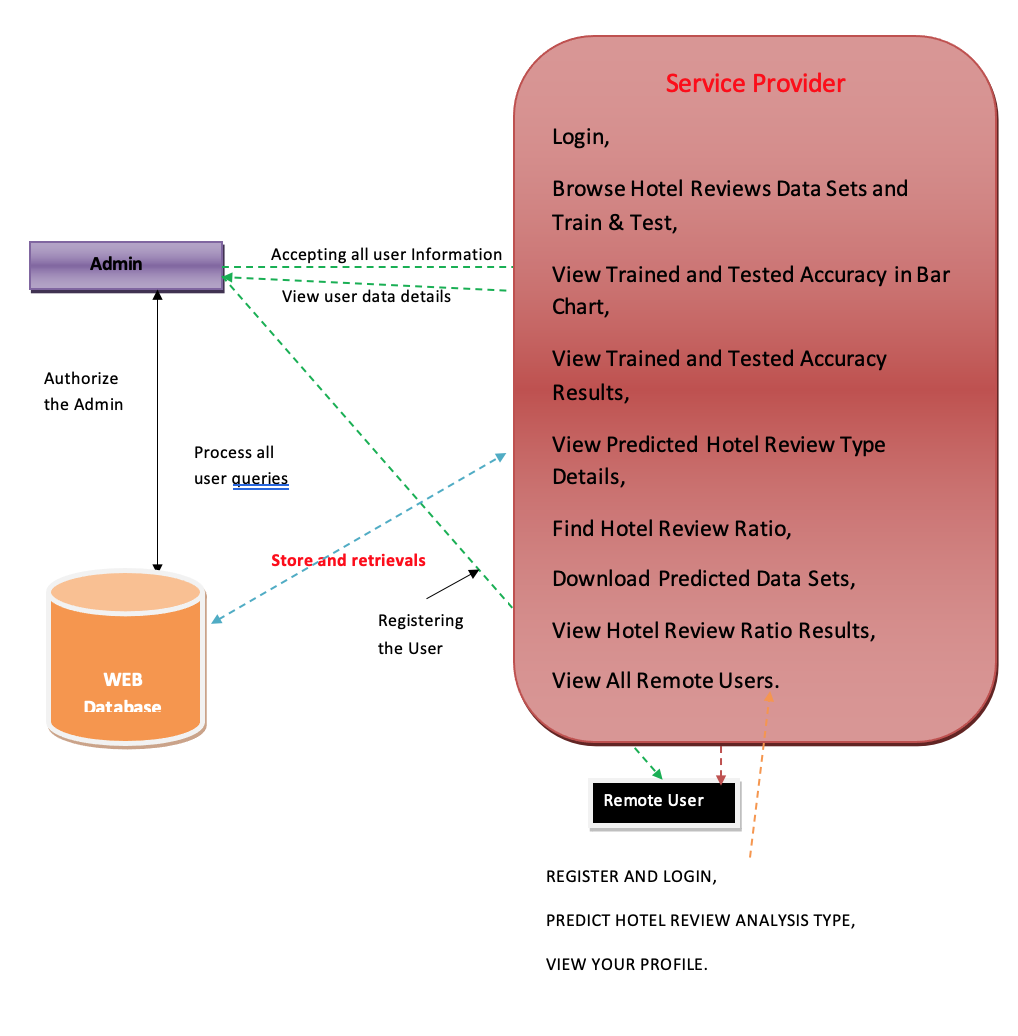
****

**Figure 3.3.1: UML Diagram**

CMRCET B. Tech (CSE) Page No 34

Hotel Review Analysis For The Prediction Of Business Using Deep Learning Approach

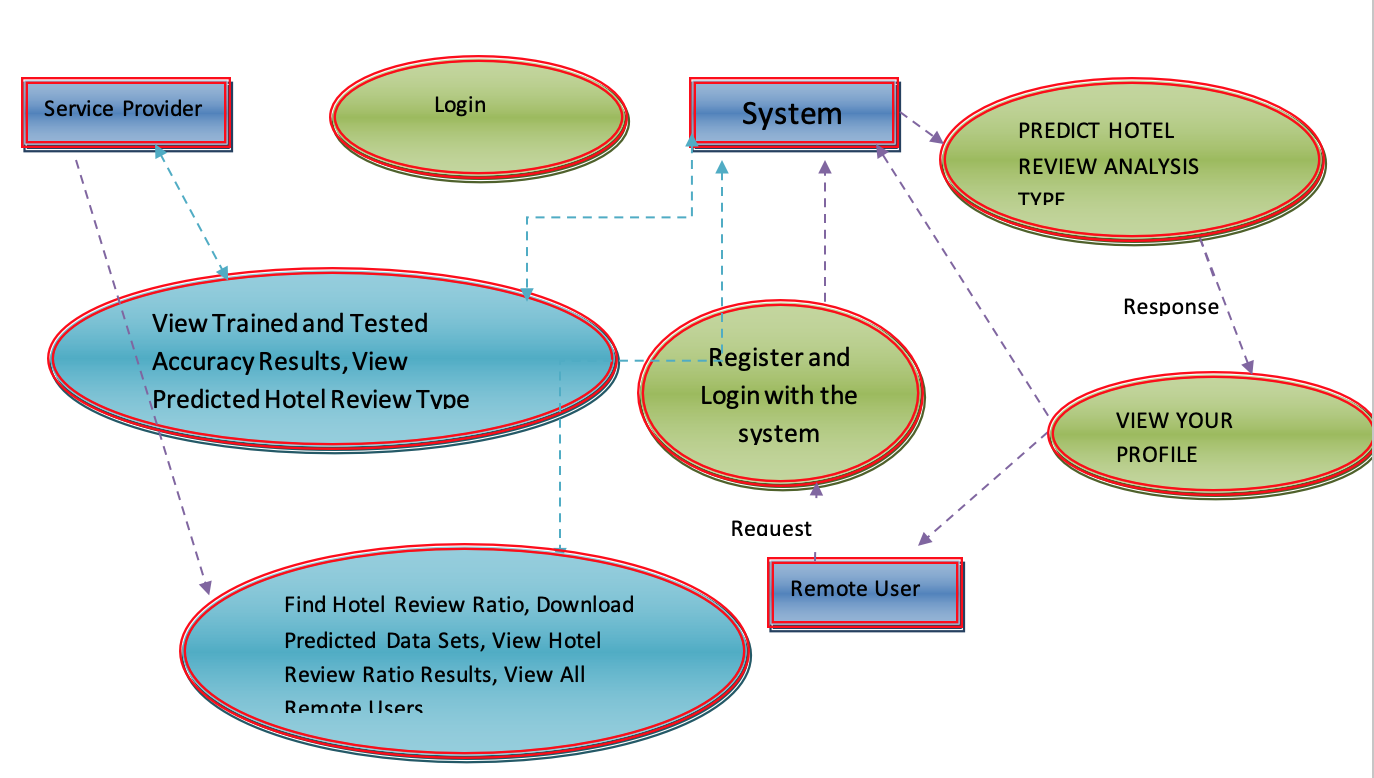
**3.3.2 Architecture Diagram**



CMRCET B. Tech (CSE) Page No 35

Hotel Review Analysis For The Prediction Of Business Using Deep Learning Approach

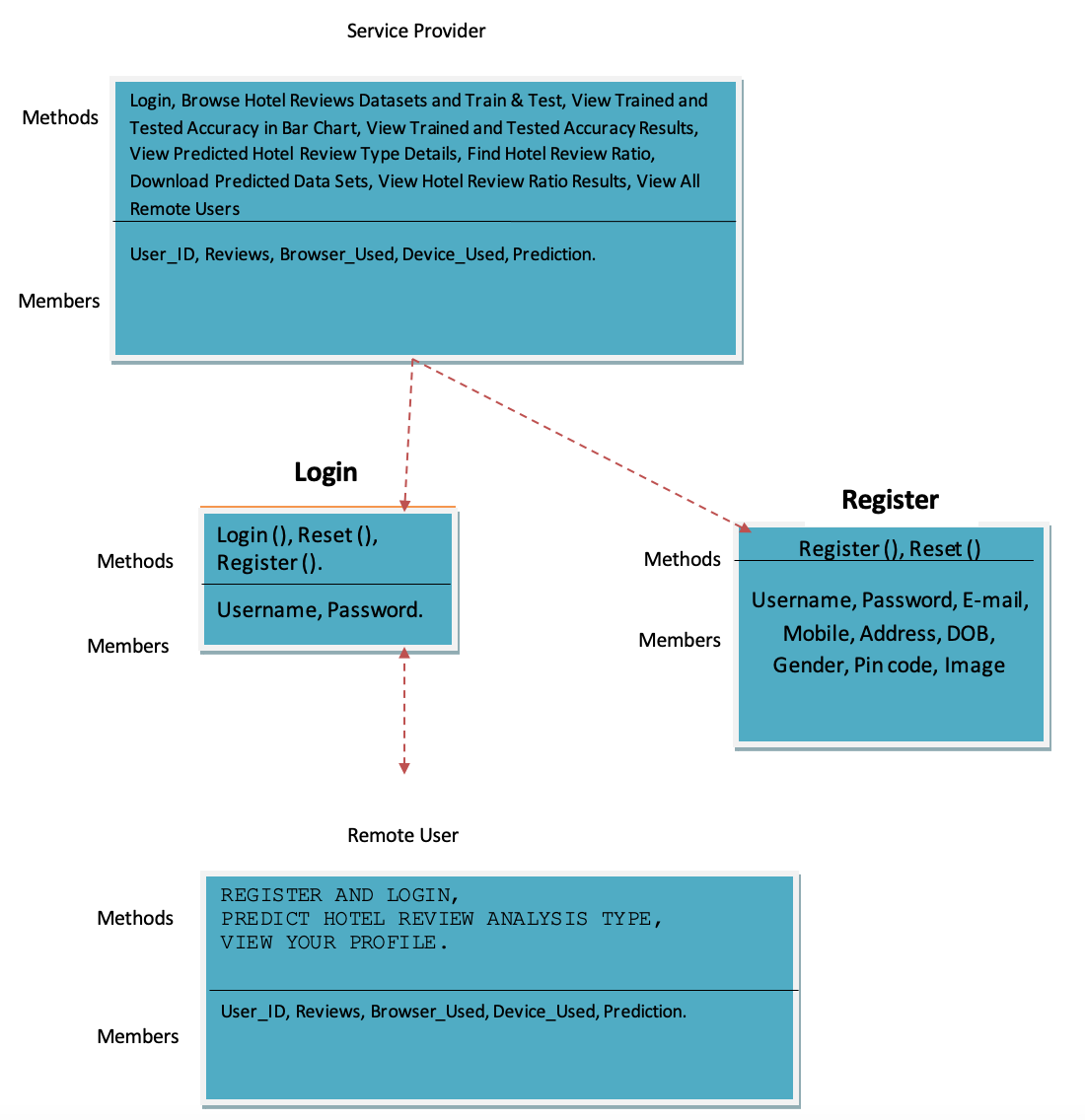
**3.3.3 Data Flow Diagram**



CMRCET B. Tech (CSE) Page No 36

Hotel Review Analysis For The Prediction Of Business Using Deep Learning Approach

**3.3.4 Class Diagram**

****

CMRCET B. Tech (CSE) Page No 37

Hotel Review Analysis For The Prediction Of Business Using Deep Learning Approach

**3.4 Stepwise Implementation and Code**

**3.4.1 Stepwise Implementation of LSTM.**

The following is a step-wise implementation of LSTM:

1. Import the required libraries:

|  |
| --- |
| import numpy as np from keras.models  import Sequential from keras.layers  import LSTM, Dense |

1. Define the input data:

|  |
| --- |
| data = np.array(  [ [0.1, 0.2, 0.3],  [0.2, 0.3, 0.4],  [0.3, 0.4, 0.5],  [0.4, 0.5, 0.6] ]  ) |

1. Define the output data:

|  |
| --- |
| labels = np.array([0.4, 0.5, 0.6, 0.7]) |

1. Reshape the input data to fit the LSTM layer:

|  |
| --- |
| data = np.reshape(data, (data.shape[0], 1, data.shape[1])) |

CMRCET B. Tech (CSE) Page No 38

Hotel Review Analysis For The Prediction Of Business Using Deep Learning Approach

1. Define the LSTM model:

|  |
| --- |
| model = Sequential()  model.add(LSTM(50, input\_shape=(1, 3)))  model.add(Dense(1)) |

In this example, the LSTM layer has 50 units and expects an input of shape (1, 3).

1. Compile the model:

|  |
| --- |
| model.compile(loss='mean\_squared\_error', optimizer='adam') |

1. Train the model:

|  |
| --- |
| model.fit(data, labels, epochs=100, batch\_size=1, verbose=2) |

In this example, the model is trained for 100 epochs with a batch size of 1.

1. Make predictions:

|  |
| --- |
| predictions = model.predict(data) |

1. Evaluate the model:

|  |
| --- |
| score = model.evaluate(data, labels, verbose=0) |

CMRCET B. Tech (CSE) Page No 39

Hotel Review Analysis For The Prediction Of Business Using Deep Learning Approach

**3.4.2 Code of LSTM:**

import nltk

import re

import string

from nltk.corpus import stopwords

from sklearn.feature\_extraction.text import CountVectorizer

from nltk.stem.wordnet import WordNetLemmatizer

import pandas as pd

from wordcloud import WordCloud, STOPWORDS

from sklearn.feature\_extraction.text import CountVectorizer

from sklearn.metrics import accuracy\_score, confusion\_matrix, classification\_report

from sklearn.metrics import accuracy\_score

from sklearn.metrics import f1\_score

import keras

import math

import nltk

def Train\_Test\_DataSets(request):

detection\_accuracy.objects.all().delete()

data = pd.read\_csv("Hotel\_Reviews\_Datasets.csv")

def clean\_text(text):

'''Make text lowercase, remove text in square brackets,remove links,remove punctuation

and remove words containing numbers.'''

text = text.lower()

text = re.sub('\[.\*?\]', '', text)

text = re.sub('https?://\S+|www\.\S+', '', text)

CMRCET B. Tech (CSE) Page No 40

Hotel Review Analysis For The Prediction Of Business Using Deep Learning Approach

text = re.sub('<.\*?>+', '', text)

text = re.sub('[%s]' % re.escape(string.punctuation), '', text)

text = re.sub('\n', '', text)

text = re.sub('\w\*\d\w\*', '', text)

return text

data['text'] = data['Description'].apply(lambda x: clean\_text(x))

def remove\_emoji(text):

emoji\_pattern = re.compile("["

u"\U0001F600-\U0001F64F" # emoticons

u"\U0001F300-\U0001F5FF" # symbols & pictographs

u"\U0001F680-\U0001F6FF" # transport & map symbols

u"\U0001F1E0-\U0001F1FF" # flags (iOS)

u"\U00002702-\U000027B0"

u"\U000024C2-\U0001F251"

"]+", flags=re.UNICODE)

return emoji\_pattern.sub(r'', text)

data['text'] = data['text'].apply(lambda x: remove\_emoji(x))

data['text'].apply(lambda x: len(str(x).split())).max()

mapping = {'not happy':0,

'happy':1}

data['Review\_Analysis'] = data['Is\_Response'].map(mapping)

x = data['Description']

CMRCET B. Tech (CSE) Page No 41

Hotel Review Analysis For The Prediction Of Business Using Deep Learning Approach

y = data['Review\_Analysis']

cv = CountVectorizer()

x = cv.fit\_transform(x)

models = []

from sklearn.model\_selection import train\_test\_split

X\_train, X\_test, y\_train, y\_test = train\_test\_split(x, y, test\_size=0.20)

X\_train.shape, X\_test.shape, y\_train.shape

model = keras.Sequential([

keras.layers.Embedding(vocab\_size, embedding\_dim, input\_length=max\_length),

keras.layers.Bidirectional(keras.layers.LSTM(64)),

keras.layers.Dense(24, activation='relu'),

keras.layers.Dense(1, activation='sigmoid')

])

model.compile(loss='binary\_crossentropy',

optimizer='adam',

metrics=['accuracy'])

model.summary()

num\_epochs = 5

history = model.fit(train\_padded, train\_labels,

epochs=num\_epochs, verbose=1,

validation\_split=0.1)

prediction = model.predict(test\_padded)

pred\_labels = []

for i in prediction:

CMRCET B. Tech (CSE) Page No 42

Hotel Review Analysis For The Prediction Of Business Using Deep Learning Approach

if i >= 0.5:

pred\_labels.append(1)

else:

pred\_labels.append(0)

CMRCET B. Tech (CSE) Page No 43

Hotel Review Analysis For The Prediction Of Business Using Deep Learning Approach

**CHAPTER** **4**

**RESULTS**  **AND** **DISCUSSION**

CMRCET B. Tech (CSE) Page No 44

Hotel Review Analysis For The Prediction Of Business Using Deep Learning Approach

**CHAPTER** **4**

**RESULTS** **AND** **DISCUSSION**

**4.1 Performance Metrics**

In the proposed methodology, long short-term memory (LSTMs) have been used to train the hotel review data where the accuracy rate of identifying customer opinion is 86%. The dataset is also tested by using Naïve Bayes, Decision Tree, Random Forest, and SVM. For Naïve Bayes obtains an accuracy of 75%, for Decision Tree obtains an accuracy of 71%, for Random Forest the accuracy is 82% and for SVM our accuracy result is 71%. Deep learning is used to obtain better business performance and also get the review from customers and also to predict the sentiment about customer review. Our algorithm works properly and gives better accuracy.

It shows the performance metrics of each classifier on the test data. The metrics were calculated using the following formulas:

Accuracy = Number of samples correctly classified

Total number of samples

Precision = Number of +ve samples classified as +ve

Total number of samples classified as +ve

Recall = Number of +ve samples classified as +ve

Total number of +ve samples

CMRCET B. Tech (CSE) Page No 45

Hotel Review Analysis For The Prediction Of Business Using Deep Learning Approach

**Table 4.1.1 Performance metrics of each classifier on test data**

|  |  |
| --- | --- |
| **CLASSIFIER:** | **ACCURACY(%):** |
| Navive Bayes | 75 |
| SVM | 71 |
| Decision Tree | 71 |
| LSTM | 86 |

**4.2 Comparison of Existing Solutions**

* In the existing work, the system did not find sentiment timeline and finds difficulties in topic modeling.
* In the existing system the algorithm used is SVM, Naïve Bayes, Decision tree which are less accurate which results in the wrong information.
* The existing methods fail to detect feedback for a large amount of data.
* If a large amount of data is found then the existing system fails to give accurate results

**4.3 Data Collection and Performance metrics**

**4.3.1 Data Collection**

The Data set contains a total of 20623 records and 7 features. The positive Reviews in set are 14066 and Negative reviews are 6557. The positive Reviews are labeled 1 and negative as 0. The dependent or Target variable is the Binary classification i.e 1 or 0. The independent variable is processed Text which will be the value fed to the model.

CMRCET B. Tech (CSE) Page No 46

Hotel Review Analysis For The Prediction Of Business Using Deep Learning Approach

CHAPTER 5

**CONCLUSION**

CMRCET B. Tech (CSE) Page No 47

Hotel Review Analysis For The Prediction Of Business Using Deep Learning Approach

**CHAPTER** **5**

**CONCLUSION**

**5.1 Conclusion and Future Enhancements**

In this project, we defined features of Review analysis and we proposed a classification model in order to classification of the reviews. This method consists of reviews extraction from websites and classification section.

We aim to increase the accuracy of the prediction model and produce better analysis using LSTM.

The model could obtain an accuracy of 86% even on large data sets but the added advantage of the proposed model over other existing models is that our model can handle unseen or new data without effecting the evaluation or classification considerably.

The model can be enhanced to perform additional functions such as identifying improvement suggestions ,categorising customers and determine the sentiment type and score.

CMRCET B. Tech (CSE) Page No 48

Hotel Review Analysis For The Prediction Of Business Using Deep Learning Approach

**CHAPTER** **6**

**REFERENCES**

CMRCET B. Tech (CSE) Page No 49

Hotel Review Analysis For The Prediction Of Business Using Deep Learning Approach

**CHAPTER** **6**

## REFERENCES

[1] R. K. Bakshi, N. Kaur, R. Kaur and G. Kaur, "Opinion mining and sent iment analysis," 2016 3rd Internat ional Conference on Comput ing for Sustainable Global Development

(INDIACom), New Delhi, 2016, pp. 452-455.

[2] L. Yang, Y. Li, J. Wang and R. S. Sherrat t , "Sent iment Analysis for E-Commerce

Product Reviews in Chinese Based on Sent iment Lexicon and Deep Learning," in IEEE Access, vol. 8, pp. 23522-23530, 2020, doi: 10.1109/ACCESS.2020.2969854.

[3] H. S. and R. Ramathmika, "Sentiment Analysis of Yelp Reviews by Machine Learning," 2019 Internat ional Conference on Intelligent Comput ing and Cont rol Systems (ICCS), Madurai, India, 2019, pp. 700-704, doi: 10.1109/ICCS45141.2019.9065812. Z. Singla, S. Randhawa and S. Jain, "Stat ist ical and sent imentanalysis of consumer product reviews," 2017 8th Internat ional Conference on Comput ing, Communicat ion and Networking Technologies (ICCCNT), Delhi, 2017, pp. 1-6, doi: 10.1109/ICCCNT.2017.8203960.

https://ieeexplore.ieee.org/document/9498965.

CMRCET B. Tech (CSE) Page No 50

Hotel Review Analysis For The Prediction Of Business Using Deep Learning Approach

**Github Link**: https://github.com/savanp08/Hotel-Review-Analysis

CMRCET B. Tech (CSE) Page No 51

# **Thank** **you!**