Project Report on

"User v/s Bot : Analyzing Sentiment from Legal Chats"

Project submitted
in partial fulfilment of the necessities for the degree of
MASTER OF COMPUTER APPLICATION

By

Abhilash Majumdar

Roll No: **001910503007**
Registration No: **149870** of **2019-20**

Under the supervision of

Dr. Dipankar Das

Department of Computer Science and Engineering
Faculty of Engineering and Technology

Jadavpur University
Kolkata – 700032, India

Jadavpur University
Faculty of Engineering and Technology
Department of Computer Science and Engineering

Certificate of Recommendation

This is to certify that **Abhilash Majumdar** (Reg. No.: 149870 of 2019-2020, Roll No: 001910503007) is a student in Master of Computer Application course and the project entitled "**User v/s Bot: Analyzing Sentiment from Legal Chats**" is a bonafide record of work carried out by him, is accepted in partial fulfilment of the requirement for the degree of **Master of Computer Application** from the **Department of Computer Science and Engineering, Jadavpur University** during the academic year **2021-2022**. He has been able to follow all the instructions in a calm and responsible way and successfully carried out his research work. Wish him all the best for his future endeavors.

Dr. Dipankar Das (Project Supervisor)
Assistant Professor, Dept. of Comp. Science & Engineering
Jadavpur University, Kolkata-700032

Prof. Anupam Sinha
Head of the Department, Dept. of Comp. Science & Engineering
Jadavpur University, Kolkata-700032

Prof. Chandan Majumdar
Dean, Faculty Council of Engineering & Technology
Jadavpur University, Kolkata-700032

Jadavpur University
Faculty of Engineering and Technology
Department of Computer Science and Engineering

CERTIFICATE OF APPROVAL

This is to clarify that the project entitled "**User v/s Bot : Analyzing Sentiment from Legal Chats**" has been completed by **Abhilash Majumdar.** This work is applied under the supervision of **Dr. Dipankar Das** in partial fulfilment for the award of the degree of **Master of Computer Applications** of the **Department of Computer Science and Engineering**, **Jadavpur University**, during the academic year **2021-2022**. The project report has been approved because it satisfies the tutorial requirements in respect of project work prescribed for the said degree.

……………………………………………………

Signature of Examiner 1

Date :-

……………………………………………………

Signature of Examiner 2

Date :-

Jadavpur University
Faculty of Engineering and Technology
Department of Computer Science and Engineering

Declaration of Originality and Compliance of Academic Ethics

I hereby declare that this project contains original work by the undersigned candidate, as a part of his Master of Computer Applications (MCA) studies.

All information during this document is obtained and presented in accordance with academic rules and ethical conduct.

I declare that, as required by these rules and conduct, I have got fully cited and referenced all material results that don't seem to be original to the current work.

I also declare that this can be a real copy of my thesis, including any final revisions, which this thesis has not been submitted for higher degree to the other University or Institution.

Name :- **Abhilash Majumdar**
Registration No :- **149870** of **2019-20**
Class Roll No :- **001910503007**
Project Title :- **"User v/s Bot : Analyzing Sentiment from Legal Chats"**

____
(Signature of the Candidate)

Jadavpur University
Faculty of Engineering and Technology
Department of Computer Science and Engineering

Acknowledgement

With my most sincere and gratitude, I would like to thank **Dr. Dipankar Das**, **Department of Computer Science & Engineering**, my supervisor, for his overwhelming support throughout the duration of the project. His motivation always gave me the required inputs and momentum to continue with my work, without which the project work would not have taken its current shape. His valuable suggestion and numerous discussions have always inspired new ways of thinking. I feel deeply honored that I got this opportunity to work under him.

I would also prefer to thank the experts involved within the validation survey for this project, **Prof. Anupam Sinha**, Head of the Department of Computer Science and Engineering and every one the university faculty members of the **Department of Computer Science and Engineering**.

Finally, I need to 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 method of my project work and writing this project report. This accomplishment wouldn't be possible without them.

Thank you.

(Signature of the Candidate)

Abstract

Sentiment analysis, a subfield of Natural Language Processing used to mingle its arms with Machine Learning (ML) is becoming popular in recent trends as we have huge amount of data in text, audio or video in WWW. In global spectrum, chatbots are becoming the essential assistants in our daily activities and the developments of such bots are being carried out under the umbrella of discourse and pragmatic sections of NLP. The intelligence is added and fuelled into the bots while it tries to determine emotions from such textual or voice data. The goal of sentiment analysis is to define automatic tools able to extract subjective information, like opinions and sentiments from language texts and pictures. One of our objectives is to study sentiment from chat data from the perspectives of both user and bots so that to shape the response of the bots based on user sentiments and vice versa.

On the other hand, it's one in every of the key chatbot features is to analyze customer data by mining thoughts, opinions, or sentiments. The employment of conversational sentiment analysis enables a chatbot to grasp the mood of the customer by sentence structures and verbal cues. Bots can use sentiment analysis to switch responses in tune with customer's emotions and thus help segment the audience.

User-bot is an interactive conversational agent. User-bots are employed in different domains like gaming, customer service, information provider etc. Siri, Alexa, Cortana (given as examples for such conversational agents). Sentiment analysis from natural language texts is the process of identifying subjective clues and categorizing the text or statement into positive, negative or neutral classes. The present work describes about incorporating sentiment analysis on User-bot interactive dataset. Our goal is to review dialogues of user and bot datasets and find out the sentiments and analyze those sentiments. We have used supervised as well as deep learning models to analyze the sentiments. By using Logistic Regression method, we have achieved 77% accuracy and using RNN and LSTM models, we have obtained 80% and 85% accuracies, respectively.

Keywords :- User, bot, Sentiment Analysis, Natural Language Processing, Machine Learning.

TABLE OF CONTENTS

1. **INTRODUCTION** -- 1
    - 1.1 CHALLENGES -- 2
    - 1.2 GAP IDENTIFICATION -- 2
    - 1.3 MOTIVATION -- 2
    - 1.4 PROBLEM STATEMENT --- 3
    - 1.5 OBJECTIVES -- 3
    - 1.6 CONTRIBUTIONS --- 3

2. **RELATED WORK** -- 4

3. **DATASET PREPARATION** --- 6
    - 3.1 DESCRIPTION OF THE DATASET -- 6
    - 3.3 STATISTICS -- 6
    - 3.3 DATASET USED IN EXPERIMENT -- 7
    - 3.4 DIALOGUE PRE-PROCESSING --- 7

4. **METHODOLOGY** --- 9
    - 4.1. SENTIMENT ANALYSIS MODELS -- 9
        - 4.1.1. SUPERVISED LEARNING MODEL ---------------------------------- 9
            - 4.1.1.1. LOGISTIC REGRESSION ------------------------------- 10
        - 4.1.2. DEEP LEARNING MODEL -- 11
            - 4.1.2.1. RNN MODEL --- 12
            - 4.1.2.2. LSTM MODEL -- 14

5. **EXPERIMENT AND RESULTS** --- 16
    - 5.1. LOGISTIC REGRESSION MODEL -- 18
    - 5.2. RNN MODEL -- 20
    - 5.3. LSTM MODEL --- 23

6. **OBSERVATIONS AND DISCUSSIONS** -- 26
    - 6.1. SUPERVISED LEARNING MODEL -- 26
        - 6.1.1. FEATURE ANALYSIS --- 26
    - 6.2. DEEP LEARNING MODEL -- 26
    - 6.3. DISCUSSIONS ON OUTPUTS --- 27

7. **ERROR ANALYSIS** -- 29

8. **CONCLUSION** -- 31

REFERENCES -- 32

LIST OF FIGURES & TABLES

TITLES :-

Figure 1: Working Principle of Supervised Learning --- 9
Figure 2: Types of Supervised Learning --- 10
Figure 3: Working Principle of Logistic Function --- 10
Figure 4: Working Architecture of Neural Network -- 12
Figure 5: Recurrent Neural Network --- 13
Figure 6: Working Architecture of Recurrent Neural Network ------------------------------- 13
Figure 7: Workflow of LSTM -- 15
Figure 8: Working Principle of LSTM -- 15

For Logistic Regression Experiments :-

Table 1: Experiment results for sentiment classification based on User-Bot data -------- 18
Table 2: Experiment results for sentiment classification based on User data ------------ 18
Table 3: Experiment results for sentiment classification based on Bot data -------------- 19
Table 4: Experiment results for sentiment classification training on User data and testing on Bot data --- 19
Table 5: Experiment results for sentiment classification training on Bot data and testing on User data -- 20

For RNN Experiments :-

Table 6: Experiment results for sentiment classification based on User-Bot data -------- 20
Table 7: Experiment results for sentiment classification based on User data ------------ 21
Table 8: Experiment results for sentiment classification based on Bot data -------------- 21
Table 9: Experiment results for sentiment classification training on User data and testing on Bot data --- 22
Table 10: Experiment results for sentiment classification training on Bot data and testing on User data -- 22

For LSTM Experiments :-

Table 11: Experiment results for sentiment classification based on User-Bot data -------23
Table 12: Experiment results for sentiment classification based on User data ------------ 23
Table 13: Experiment results for sentiment classification based on Bot data ------------- 24
Table 14: Experiment results for sentiment classification training on User data and testing on Bot data --- 24
Table 15: Experiment results for sentiment classification training on Bot data and testing on User data -- 25

1. INTRODUCTION

All citizens of every country are required to abide by the law. Despite the fact that breaking the law occasionally has consequences, it is very necessary to obtain legal guidance from an expert in these circumstances. From the perspective of a provider of legal services, serving a large number of clients at once can be difficult and tiresome. Due to the lack of a trained authorized automated agent, it is highly sought for. Natural Language Processing (NLP) techniques are being used in recent trends to train an automated agent by utilizing a variety of legal precedents.

In order to avail such assisting technologies in emergencies even, researchers are being engaged in developing natural language bots. The natural language generation based techniques are incorporating to develop modules of such a QA system. Moreover, the inclusion of sentiments into the framework adds intelligence into the system. Therefore, the user and bot level sentiment analysis becomes the primal need to explore the possibility of including sentiments or emotions into the bot systems.

In the present work, the sentiment analysis has been carried out on the utterances of user and bot from the legal conversational texts. Several machine learning and deep learning techniques have been employed to review the involvements and impacts of the sentiments in the utterances both from the perspectives of users and bots. Finally, the system is trained and tested on user and bot data individually and jointly to show the roles of sentiments. The results show that the responses of the bots can be advanced in terms of quality and intelligence by identifying the sentiments of the users. In future, we have planned to explore the work by considering a long discourse to track the impacts of sentiments. Thus, this section contains challenges, research gaps, motivation, problem statement, objectives and contributions with respect to the theme of the work.

The rest of the draft is organized as follows. The literature review on current advancements in sentiment classification and related chatbot challenges are found in Section 2. The development of corpus on legal conversations and pre-processing are described in Section 3. Frameworks for machine learning and deep learning classifiers that are utilized for sentiment classifiers are described in Section 4. Section 5 describes the experimental results for each of the classifiers on the datasets, and Section 6 emphasizes significant findings related to the results. Error analysis is covered in Section 7 using several examples. The work is finally concluded with future directions in Section 8.

1.1 Challenges

The dataset consists of 560 conversations with a total of 3168 utterances containing 1594 user utterances and 1574 conversational agent (bot) utterances. Each of the user and bot utterances may fall under multiple intent classes out of 29 predefined intent classes and marked with a sentiment score within a range of -5 to +5 based on annotators' perception. We have employed several supervised and deep learning models to analyze the sentiments. By using Linear Regression model, we have achieved 77% accuracy and using RNN and LSTM we have achieved 83% and 85% accuracies, respectively. It took a long time to develop the dataset.

1.2 Gap Identification

Many researchers have had success recently in doing sentiment analysis. However, there has never been a successful implementation of a legal dataset based on user and bot discussions. Therefore, I have selected this project to work on a dataset that is constructed based on legal conversations from individual chats.

1.3 Motivation

In order to analyze the user's problems and recommend a likely remedy in exchange for sparse resources, the automated training agent (bots) needs be taught in legal contexts. To the best of our knowledge, such a huge conversational dataset is not available for that purpose. Additionally, based on the user's scenario, it should be able to effectively classify the user's intentions and generate the relevant responses. There is no prior work done on sentiment analysis based on user and bot specific utterances in legal domain. This gave me the idea of doing sentiment analysis on legal dialogues. A chat dataset with legal context is offered here, along with a comparison of various sentiment analysis techniques.

1.4 Problem Statement

This study aims to enhance a conversational bot interface to provide topically relevant responses and emotionally acceptable expressions during real-time interactions with users.
The present work discusses the impact of sentiment analysis on user-bot interactions. Our objective is to examine conversations from user and bot datasets, identify the sentiments, and analyze those sentiments from both the perspectives of users and bots.

1.5 Objectives

The work describes about investigating the roles of sentiment in user-bot interactive dataset especially in legal domain. Our goal is to review dialogues of user and bot datasets and find out the sentiments and analyze those sentiments.
One of the main objectives of the research is to identify that, training on the user data, the system is whether capable of predicting the same sentiment or not by testing on bot data. The sentiment will be either positive or negative. Similarly, we have to train on bot data and test on user data, to check user is capable of predicting the same sentiment or not. The idea is to generate interactive responses in a discussions thread to capture and utilize sentiments so as to make an intelligent agent.

1.6 Contributions

Overall, the dataset contains 3168 dialogue utterances. The dataset contains Speaker, Statements, Intents and Sentiments. There are 1694 utterances for the user and 1574 utterances for the bot. Sentiment scores are within the range of -5 to +5. We have used 80% of the dataset for the training purpose and 20% of the dataset for testing. We have used supervised and deep learning model to analyze the sentiments.

2. RELATED WORK

While surveying the related researches carried out so far in this area, some of the interesting works are found. In the article "CASA: Conversational Aspect Sentiment Analysis for Dialogue Understanding", the authors of (Song, Linfeng, et al. 2022) have presented the conversational aspect based sentiment analysis (CASA) task, which can offer beneficial fine-grained sentiment information for dialogue interpretation and planning. The overall goal of this job is to adapt the usual aspect-based sentiment analysis to the conversational scenario. They annotated 3,000 casual conversations (27,198 phrases) with fine-grained sentiment data, which included all sentiment expressions, their polarities, and the corresponding target mentions, to help with the training and evaluation of data-driven algorithms. Additionally, during a preliminary investigation, they created a number of baselines based on either pre-trained BERT or self-attention.

In the paper "End-to-end dialogue with sentiment analysis features" by (Rinaldi, Alex, et al. 2017), they have proposed an artificially intelligent, text-based conversational agent that produces responses resembling those of a counselor. As a stopgap before speaking with a licensed psychiatrist, the application will provide a momentary outlet for expression. By using sentiment analysis data, or ratings that rank the emotional content of users' text input, when producing responses, they improved upon prior methods. Additionally, using a dataset of transcripts from counseling and therapy sessions, they trained a response generation system. According to quantitative criteria of quality, they have shown that the integration of sentiment analysis data offers somewhat superior responses.

In another paper "Learning interaction dynamics with an interactive LSTM for conversational sentiment analysis" by (Zhang, Yazhou, et al. 2021), they have proposed an interactive long short-term memory (LSTM) network for conversational sentiment analysis to model interactions between speakers in a conversation. They have added a confidence gate before each LSTM hidden unit to gauge the veracity of the previous speakers and combining the output gate with the learned influence scores to account for their influences. The findings showed that their model outperforms a wide range of strong baselines and obtains competitive results using the most recent techniques.

In the article, "A Statistical Parsing Framework for Sentiment Classification" by (L. Dong, F. Wei, S. Liu, M. Zhou, and Ke. Xu's ,2005), the authors have proposed a statistical parsing framework for sentence-level sentiment categorization. They have demonstrated that complex sentiment analysis phenomena (such as negation, intensification, and contrast) which may be treated in the same way as plain and simple sentiment expressions in a unified and probabilistic manner. They have presented a formal explanation of the architectures for sentiment parsing and have built the sentiment grammar (sCFGs).

3. DATASET PREPARATION

3.1 Description of the Dataset

Chat bots, that translate important judgments to vernacular languages, can help in disseminating legal knowledge to the general public and improving the awareness of people about the functioning of the judiciary. So, we have developed a single dataset from a single source. The dataset consists of 29 intents with a total number of 3168 utterances collected from an online legal forum.

We focus on understanding the user's intention as well as the gravity of the situation. Due to that reason, each of the user utterances has been annotated with two attributes, intent and sentiment score within the range of -5 to +5. Similarly, each bot utterance has been annotated sentiment score within the range of -5 to +5.

3.2 Statistics

The dataset contains 3168 dialogues. There are 3168 rows and four columns. The columns are Speaker, Statements, Intents and Sentiments. There are 1694 rows for the user and 1574 rows for the bot. Sentiment scores are within the range of -5 to +5. We have used 80% of the dataset for the training purpose and 20% of the dataset for testing.

3.3 Dataset used in Experiment

The dataset is as follows.

Speaker	Statements	Intents	Sentiments
User	Hello.	@greet	0
Bot	hello, how can i help you?	@greet	0
User	I have been married for 6 years & having kids. After 6 months of my marriage my wife started mental harassment. Fighting with small issues & even some times she is creating dummy issues to fight with me. Now she is harassing my parents too.	@marriage@harassment@legal_enquiry	1
User	My mothers in law is giving fuel to that. Always after fight my wife used to say give me divorce.	@marriage@harassment@legal_enquiry	-1
Bot	You can not prove that your mothers in law is instigating your wife to misbehave with you, because as an adult lady like your wife should not be carried away by some one's instigation to misbehave with her own husband.	@marriage@harassment@suggestion	2
Bot	You can give her divorce and move on with your life. For that you have to collect some proofs against her to prove her guilty in court for divorce approval.	@marriage@harassment@suggestion	-2
User	Okay thank for your help.	@thank	2
Bot	It's my pleasure sir.	@thank	0

3.4 Dialogue Pre-processing

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

In the preliminary step, very first thing we did is to extract speaker and also the dialogue from the sentence into a columnar format where 1st column is that the name of speaker and 2nd column is for the dialogue said by that speaker. These sentences from the 2nd columns are further processed to get rid of and clean data with the assistance of following steps.

1. Removal of punctuations like '!" #$%&' ()*+,-./:;?@[\]^_`{|}~' and digits from the dialogue.
2. Converting the text of the dialogues 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 : - Suppose we have a set of words - **send**, **sent** and **sending**, after applying stem, we have only one word that is “**send**”.
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 : - Suppose we have a word “**better**”, after applying lemmatization, the word “**better**” becomes “**good**”.

Suppose, we have a word “**believes**”, after applying Stemming and lemmatization, the word “**believes**” becomes “**believ**”.

Original word	**After Stemming**	**After Lemmatization**
believes	believ	believ

The output of the word “**believes**” describes the major difference between stemming and lemmatization. We can say, stemming technique only finds at the form of the word whereas lemmatization technique finds at the meaning of the word. So applying lemmatization, we will always get a valid word.

In this project, dialogue pre-processing is done by removing punctuations and digits, changing the text of the dialogues to lower case, removing the stop words except for the not terms, and lemmatization are done using the spacy library[1].

---

[1] https://spacy.io/

4. METHODOLOGY

4.1 SENTIMENT ANALYSIS MODELS

In order to investigate the roles of sentiments from user and bot utterances, we have employed several learning algorithms and trained and test on user and bot data in various set up.

4.1.1 Supervised Learning Models

Supervised learning is the types of machine learning in which training the machines using well "labelled" training data, and on the basis of that data, machines can predict the output successfully. Supervised learning is a process where we can of provide input data as well as correct output data to the machine learning model. The goal of a supervised learning algorithm is to find a mapping function to map the input variable(x) with the output variable(y). It can be used for Risk assesment, Image classification, Fraud Detection etc. In this project, the classification model used is Logistic Regression.
The working of Supervised learning is shown by the following figure :-

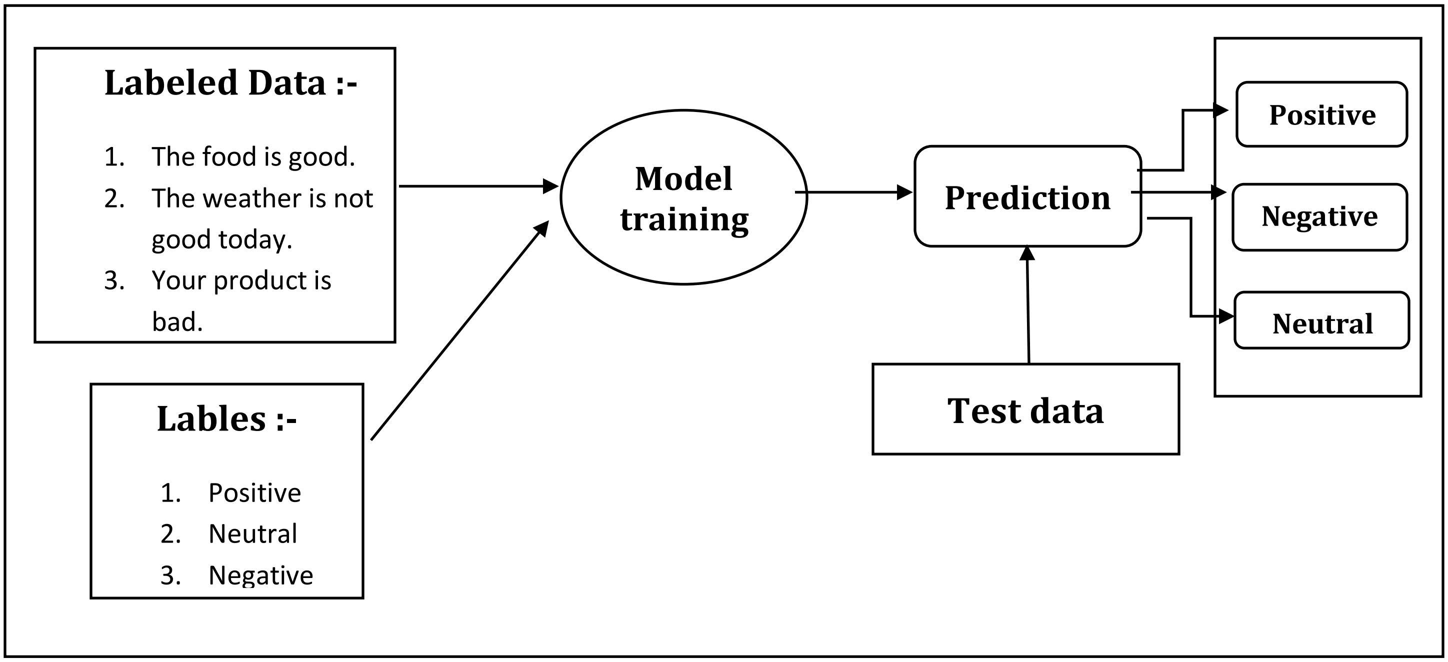


Figure 1: Working Principle of Supervised Learning

There are two types of supervised Learning :-

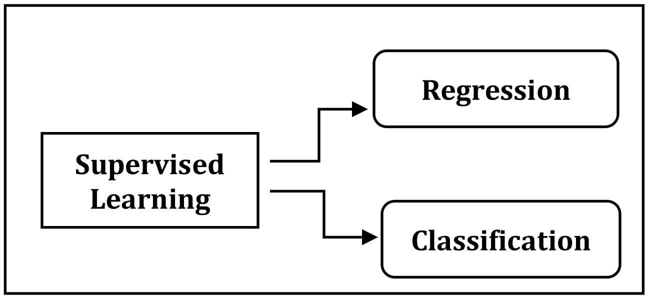


Figure 2: Types of Supervised Learning

4.1.1.1 Logistic Regression

One of the most well-known Machine Learning algorithms that falls under the Supervised Learning approach is logistic regression. It's used for predicting the explicit variable employing a given set of independent variables. Logistic regression predicts the output of a categorical variable quantity. Therefore the result must be a categorical or discrete value. It is either Yes or No, 0 or 1, true or False, etc. but rather than giving the precise value as 0 and 1, it gives the probabilistic values which lie between 0 and 1. Logistic regression uses the concept of predictive modeling as regression, therefore, it's called logistic regression, but is employed to classify samples. Therefore, it falls under the classification algorithm.

Working Principle of Logistic Function (Sigmoid Function)

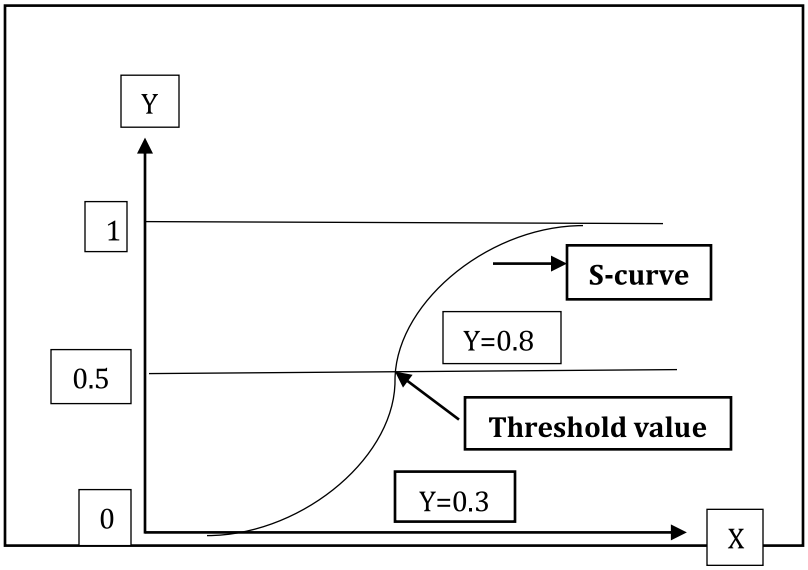


Figure 3 :- Working Principle of Logistic Function

logistic function = $1/(1+e^{-x})$

Here, x is the input variable.

The **sigmoid function** is a mathematical function used to generate the predicted values to probabilities. It predicts any real value into another value within a range of 0 and 1.

The value of the logistic regression must be between 0 and 1, which cannot transcend this limit, so it forms a curve like the **"S"** form. The S-form curve is termed the **Sigmoid function** or the **logistic function**. In logistic regression, we use threshold value, which defines the probability of either **0 or 1.** So the values above the threshold value tends to 1, and the value below the threshold values tends to 0.

4.1.2 DEEP LEARNING MODELS

Deep learning is a method of an artificial intelligence that aims to imitate the human brain's ability to process data and recognize patterns for making decisions. Deep learning is a machine learning process in artificial intelligence that uses networks capable of learning from data that is unstructured or unlabeled in an unsupervised manner. Deep learning uses a multi-layered structure of algorithms termed neural networks.

Large datasets of labelled data and neural network topologies that automatically extract characteristics from the data without the need for manual labour are used to train deep learning models. Layered neural networks are made up of a collection of interconnected nodes. There may be tens or even hundreds of hidden layers in networks.

Input Layer :- represent dimensions of input vector.

Hidden Layer :- takes up the set of weighted input and produces the output using activation function.

Output Layer :- represent output of a neural.

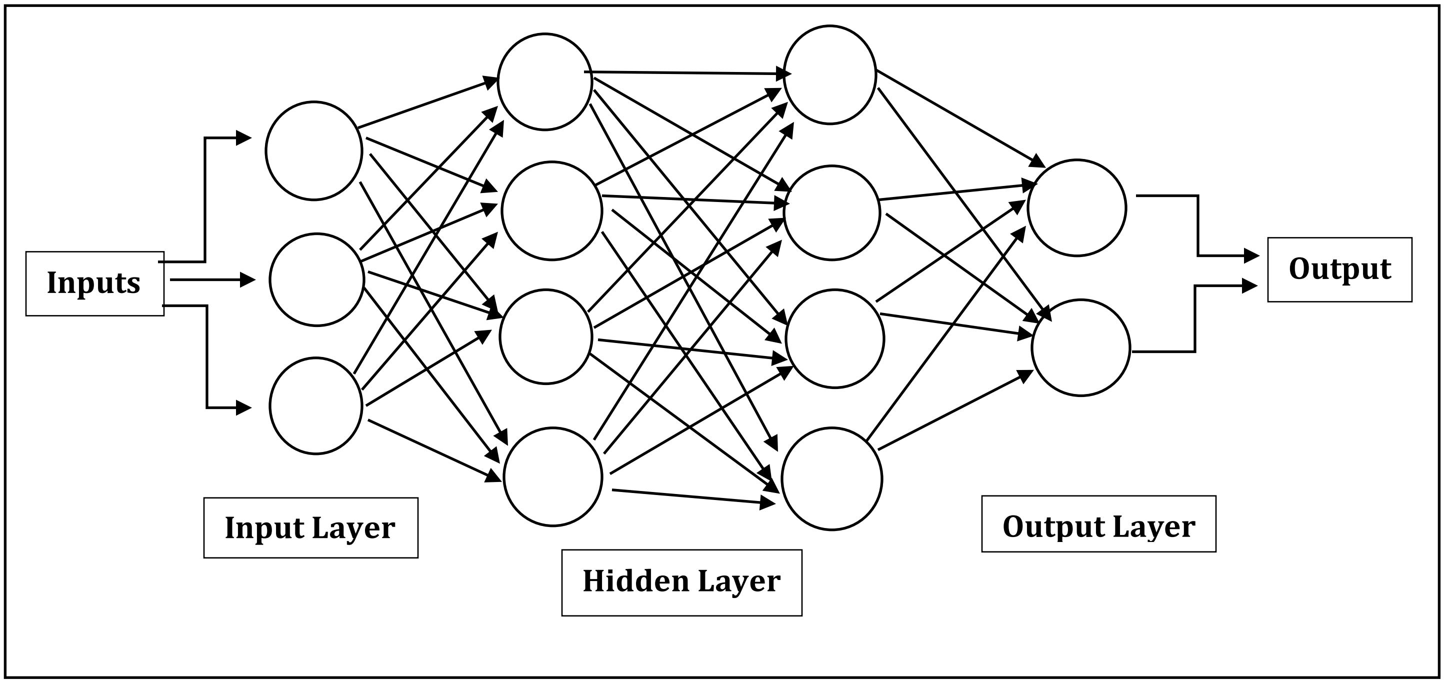


Figure 4 :- Working Architecture of Neural Network

4.1.2.1 Recurrent Neural Network

Recurrent Neural Network(RNN) are a kind of Neural Network where the output from previous step are fed as input to the current step. In traditional neural networks, all the inputs and outputs are independent of every other, but in cases like when it is required to predict the next word of a sentence, the previous words are required and hence there is a requirement to recollect the previous words. With the help of hidden layers, RNN solved this issue. The main and most significant feature of RNN is Hidden state, which remembers some information a couple of sequence.

RNN have a “memory” that is used to remember all the calculated information. It uses the identical parameters for every input because it performs the identical task on all the inputs or hidden layers to supply the output. The nodes in numerous layers of the neural network are compressed to make a single layer of recurrent neural networks. The parameters of the network are A, B and C, those are used to improve the output of the model. Here “X” is input layer, “H” is hidden layer and “Y” is output layer. Suppose at any given time t, the current input is a combination of input at X(t) and X(t-1). The output at any given time is fetched back to the network to enhance on the output.

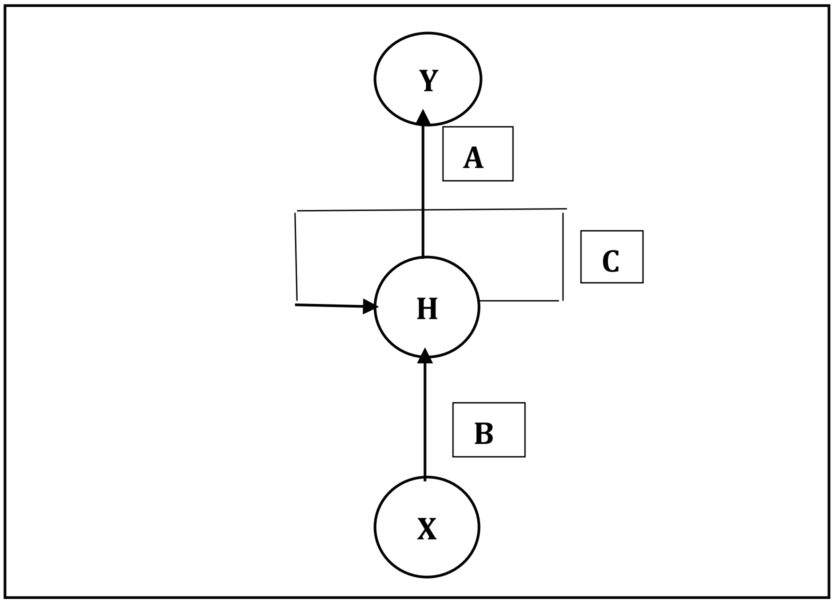


Figure 5:- Recurrent Neural Network

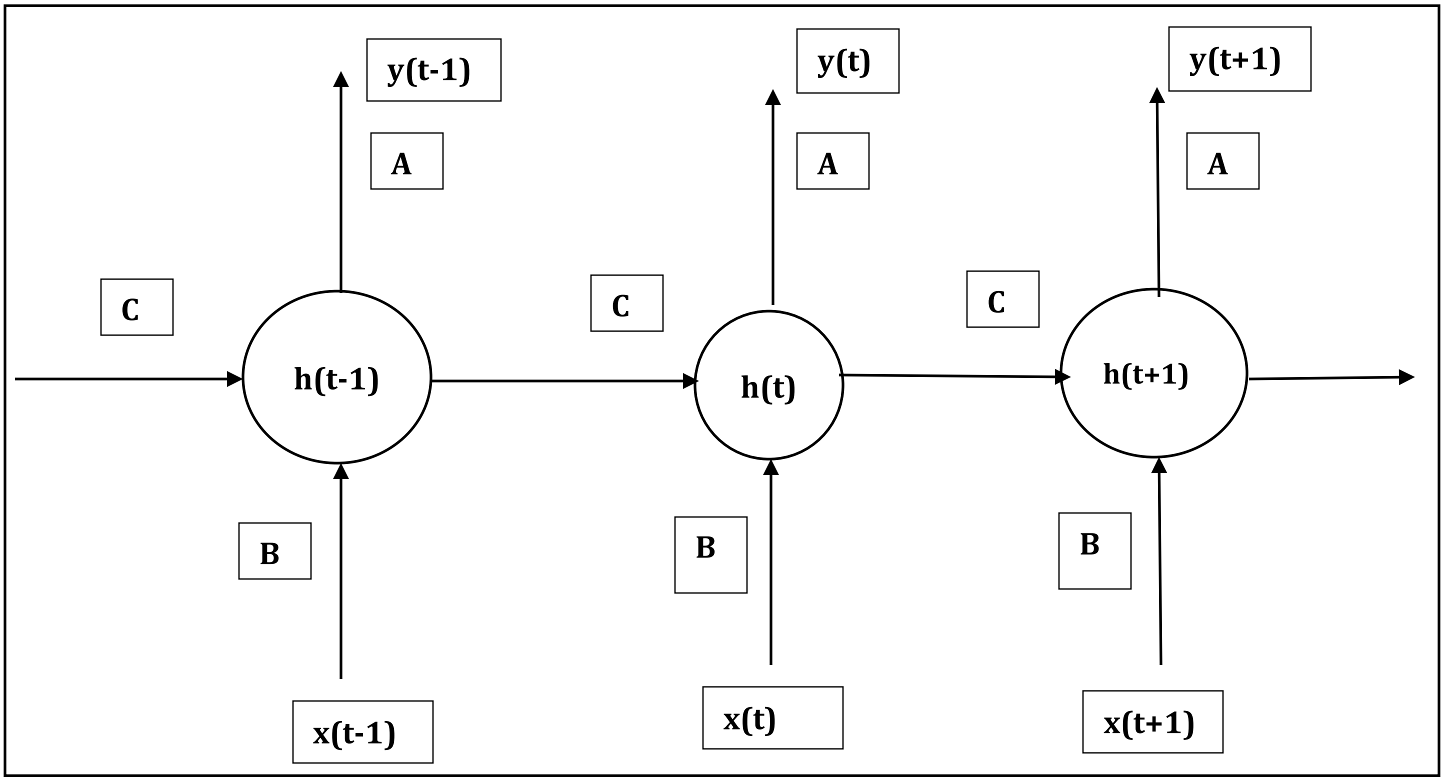


Figure 6:- Working Architecture of Recurrent Neural Network

$$\mathbf{h(t) = f_c(h(t-1), x(t))}$$

$h(t)$ = new state
f_c = function with parameter c
$h(t-1)$ = old state
$x(t)$ = input vector at time t

In Recurrent Neural networks, the information cycles through a loop to the center hidden layer. The input layer ‘x’ takes within the input to the neural network and processes it and passes it onto the center layer.

The middle layer ‘h’ can carries with it multiple hidden layers, each with its own activation functions and weights and biases. If you have a neural network where the various parameters of various hidden layers are not affected by the previous layer, ie: the neural network doesn't have memory, then you can use a recurrent neural network.

The Recurrent Neural Network will standardize the various activation functions and weights and biases in order that each hidden layer has the identical parameters. Then, rather than creating multiple hidden layers, it'll create one and loop over it as over and over as needed.

4.1.2.2 LSTM (Long Shot-Term Memory) Model :-

LSTM is a special kind of recurrent neural network, which is capable of handling long-term dependencies. Long Short Term Memory Network is an advanced RNN, a sequential network, that enables information to persist. Long Short-Term Memory (LSTM) networks are a modified version of recurrent neural networks, which makes it easier to recall past data in memory. It's capable of managing the vanishing gradient problem faced by RNN.

The difference in the workflow of LSTM and RNN is that LSTM uses an Internal State and the Internal Cell State is passed with the hidden state.

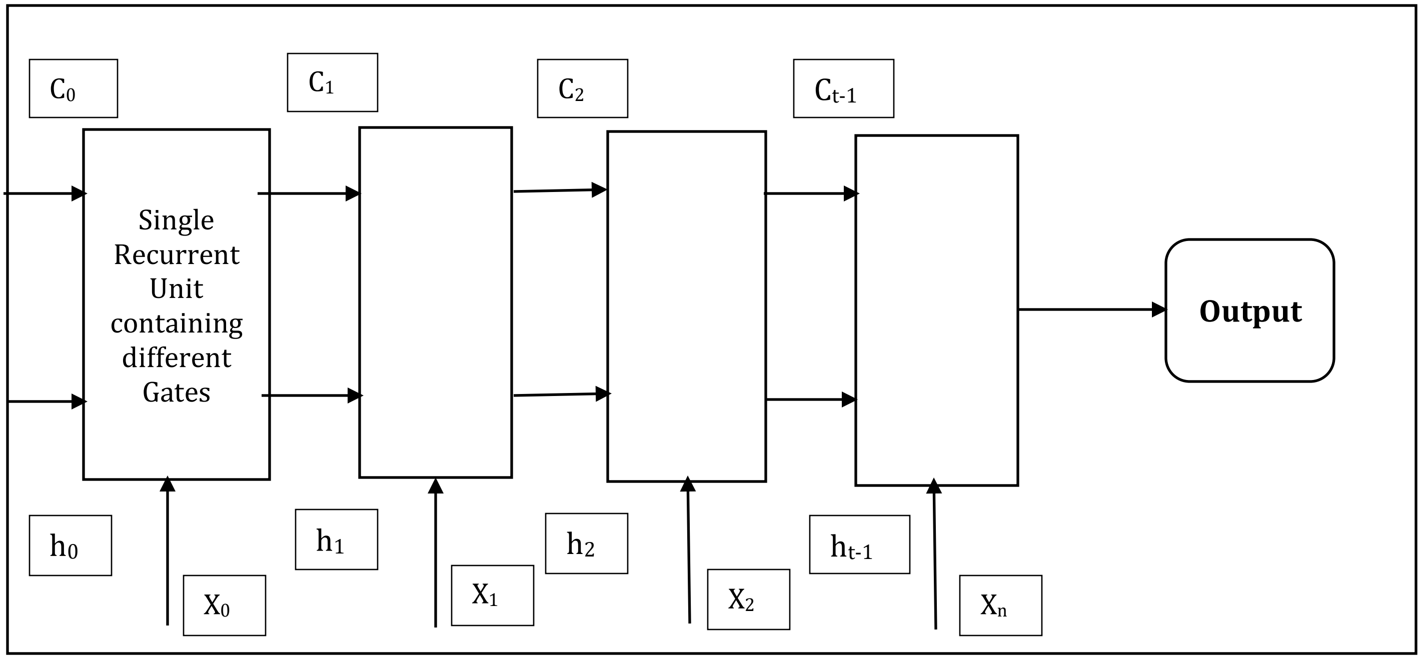


Figure 7 :- Workflow of LSTM

The input gate is used to update the state of the cell. A current state's hidden state information is passed within the tanh activation function, which produces an output between -1 and 1, multiplying the sigmoid output and the tanh output. The decision to keep the tanh output is made based on the sigmoid output. A hidden state makes predictions and keeps track of earlier states. The sigmoid activation function receives input from the previous hidden state and the present state. The tanh activation function receives modified cell states. The hidden state output is then produced by multiplying the tanh and sigmoid output. The hidden form and modified cell state are carried over to the following time step.

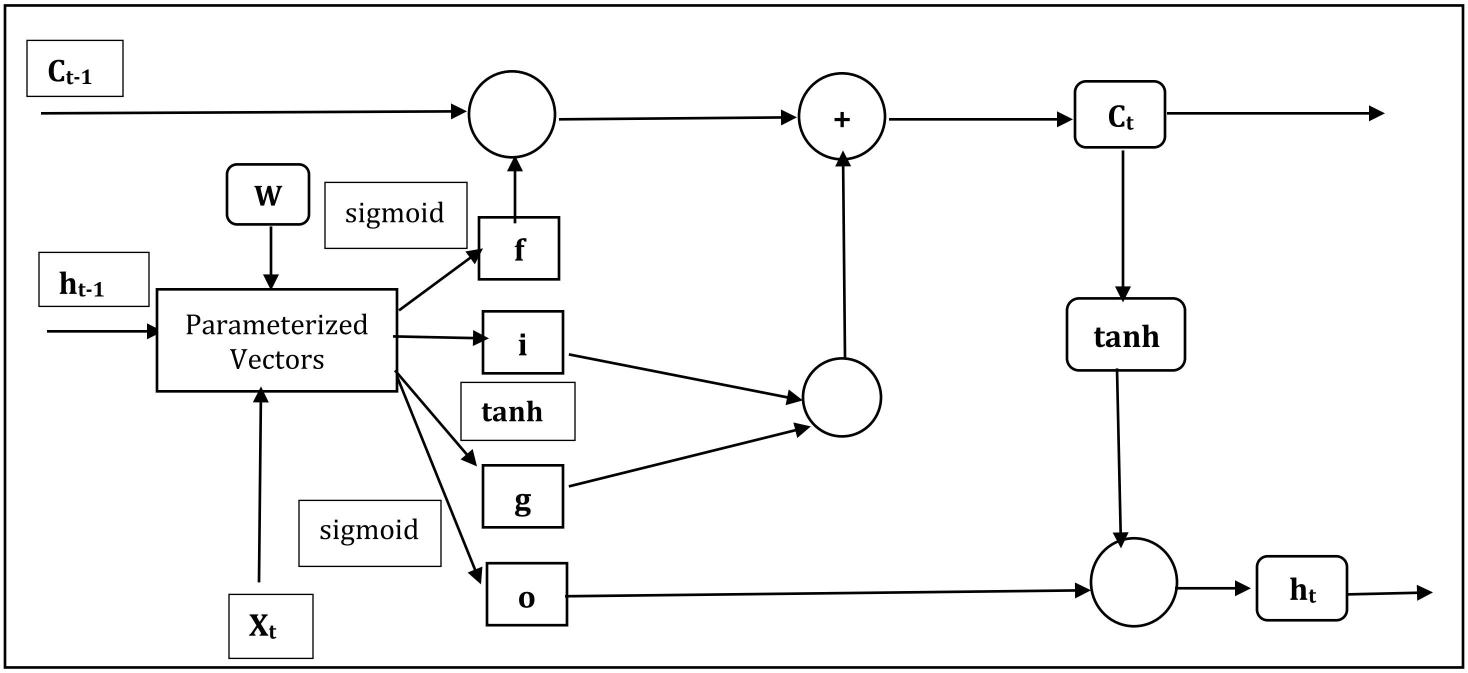


Figure 8 :- Working Principle of LSTM

5. EXPERIMENTS AND RESULTS

Here we have performed experiment based on **User and Bot dataset**. We have used 80% of the User and Bot dataset for training purpose and 20% of the User and Bot dataset for testing purpose. We also notice that **LSTM** performs best, with an **accuracy of 85%, precision score 0.82**, **recall score 0.83** and **f1 score 0.82**. The detail results are shown in below.

	Logistic Regression	RNN	LSTM
Precision	**0.45**	**0.78**	**0.82**
Recall	**0.36**	**0.78**	**0.83**
F1 score	**0.39**	**0.78**	**0.82**
Accuracy	**77%**	**83%**	**85%**

Here we have performed experiment based on **User dataset**. The result is shown in below :-

	Logistic Regression	RNN	LSTM
Precision	**0.35**	**0.76**	**0.78**
Recall	**0.28**	**0.75**	**0.79**
F1 score	**0.29**	**0.76**	**0.79**
Accuracy	**75%**	**81%**	**82%**

We have used 80% of the User dataset for training purpose and 20% of the User dataset for testing purpose. We also notice that **LSTM** performs best, **with an accuracy of 82%**, **precision score 0.78**, **recall score 0.79** and **f1 score 0.79**.

Here we have performed experiment based on **Bot dataset**. The result is shown in below :-

	Logistic Regression	RNN	LSTM
Precision	**0.52**	**0.76**	**0.77**
Recall	**0.36**	**0.75**	**0.79**
F1 score	**0.39**	**0.76**	**0.78**
Accuracy	**76%**	**81%**	**82%**

We have used 80% of the Bot dataset for training purpose and 20% of the Bot dataset for testing purpose. We also notice that **LSTM** performs best, **with an accuracy of 82%**, **precision score 0.77**, **recall score 0.79** and **f1 score 0.78**. Here we have performed experiment **training on User dataset** and **testing on Bot dataset**. The result is shown in below.

	Logistic Regression	**RNN**	**LSTM**
Precision	**0.09**	**0.60**	**0.67**
Recall	**0.10**	**0.57**	**0.62**
F1 score	**0.09**	**0.57**	**0.63**
Accuracy	**57%**	**68%**	**73%**

We notice that **LSTM** performs best, **with an accuracy of 73%**, **precision score 0.67**, **recall score 0.62** and **f1 score 0.63**. Here we have performed experiment **training on Bot dataset** and **testing on User dataset**. The result is shown in below :-

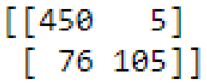
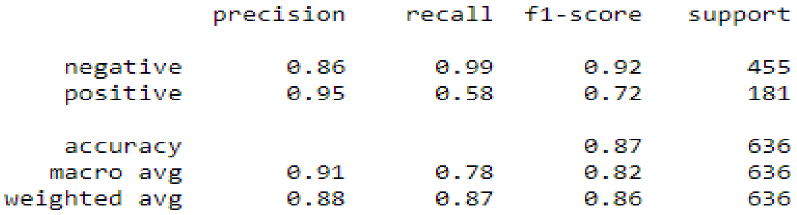
	Logistic Regression	**RNN**	**LSTM**
Precision	**0.09**	**0.66**	**0.68**
Recall	**0.09**	**0.63**	**0.66**
F1 score	**0.08**	**0.64**	**0.67**
Accuracy	**52%**	**72%**	**74%**

We notice that **LSTM** performs best, **with an accuracy of 74%**, **precision score 0.68**, **recall score 0.66** and **f1 score 0.67**.

1.1. Logistic Regression Model

Table 1: Experiment results for sentiment classification based on User-Bot data :-

METRICS	PERFORMANCE
Accuracy score	77%
Precision	0.45
Recall	0.36
F1-score	0.39

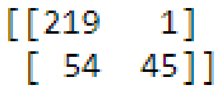
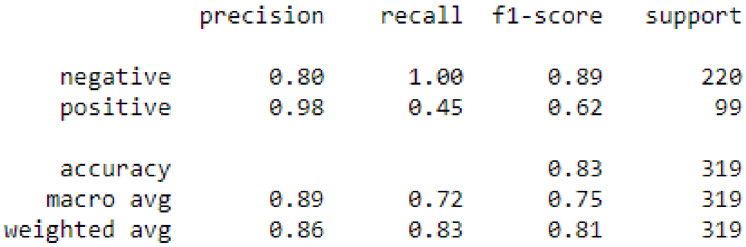
Confusion Matrix	Classification Report
	

```
[[450   5]
 [ 76 105]]
```

```
              precision    recall  f1-score   support

    negative       0.86      0.99      0.92       455
    positive       0.95      0.58      0.72       181

    accuracy                           0.87       636
   macro avg       0.91      0.78      0.82       636
weighted avg       0.88      0.87      0.86       636
```

Table 2: Experiment results for sentiment classification based on User data :-

METRICS	PERFORMANCE
Accuracy score	75%
Precision	0.35
Recall	0.28
F1-score	0.29

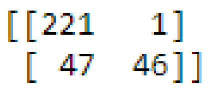
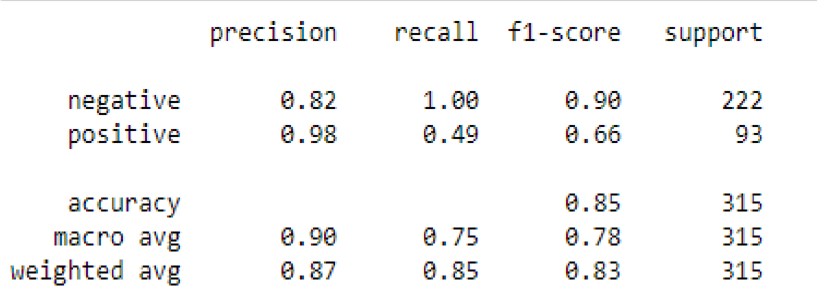
Confusion Matrix	Classification Report
	

```
[[219   1]
 [ 54  45]]
```

```
              precision    recall  f1-score   support

    negative       0.80      1.00      0.89       220
    positive       0.98      0.45      0.62        99

    accuracy                           0.83       319
   macro avg       0.89      0.72      0.75       319
weighted avg       0.86      0.83      0.81       319
```

Table 3: Experiment results for sentiment classification based on Bot data :-

METRICS	PERFORMANCE
Accuracy score	76%
Precision	0.52
Recall	0.36
F1-score	0.39

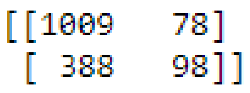
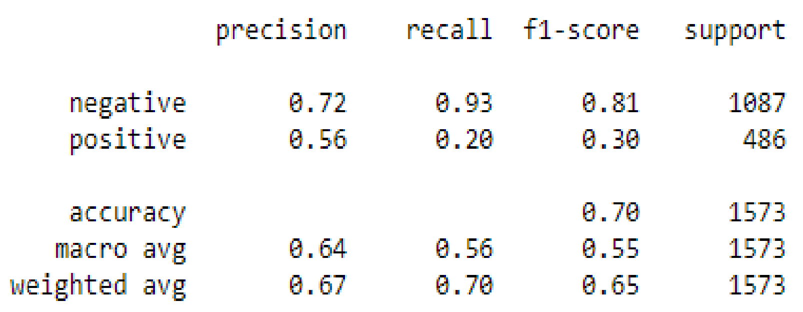
Confusion Matrix	Classification Report
	

```
[[221   1]
 [ 47  46]]
```

```
              precision    recall  f1-score   support

    negative       0.82      1.00      0.90       222
    positive       0.98      0.49      0.66        93

    accuracy                           0.85       315
   macro avg       0.90      0.75      0.78       315
weighted avg       0.87      0.85      0.83       315
```

Table 4: Experiment results for sentiment classification training on User data and testing on Bot data :-

METRICS	PERFORMANCE
Accuracy score	57%
Precision	0.09
Recall	0.10
F1-score	0.09

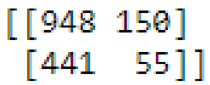
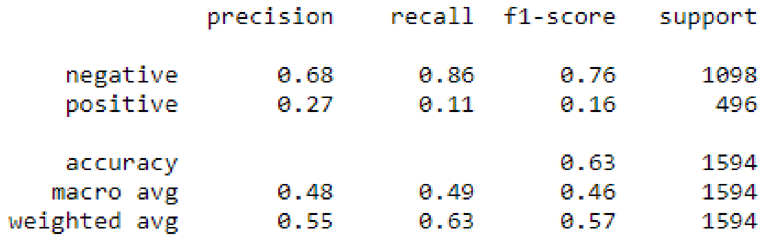
Confusion Matrix	Classification Report
	

```
[[1009   78]
 [ 388   98]]
```

```
              precision    recall  f1-score   support

    negative       0.72      0.93      0.81      1087
    positive       0.56      0.20      0.30       486

    accuracy                           0.70      1573
   macro avg       0.64      0.56      0.55      1573
weighted avg       0.67      0.70      0.65      1573
```

Table 5: Experiment results for sentiment classification training on Bot data and testing on User data :-

METRICS	PERFORMANCE
Accuracy score	52%
Precision	0.09
Recall	0.09
F1-score	0.08

Confusion Matrix	Classification Report
	

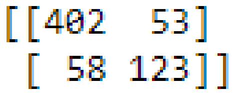
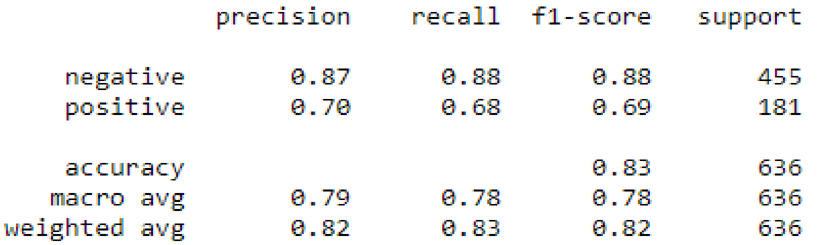
```
[[948 150]
 [441  55]]
```

```
              precision    recall  f1-score   support

    negative       0.68      0.86      0.76      1098
    positive       0.27      0.11      0.16       496

    accuracy                           0.63      1594
   macro avg       0.48      0.49      0.46      1594
weighted avg       0.55      0.63      0.57      1594
```

5.2. RNN Model

Table 6: Experiment results for sentiment classification based on User-Bot data :-

METRICS	PERFORMANCE
Accuracy score	83%
Precision	0.78
Recall	0.78
F1-score	0.78

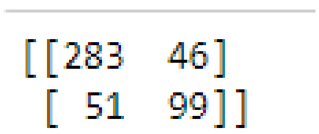
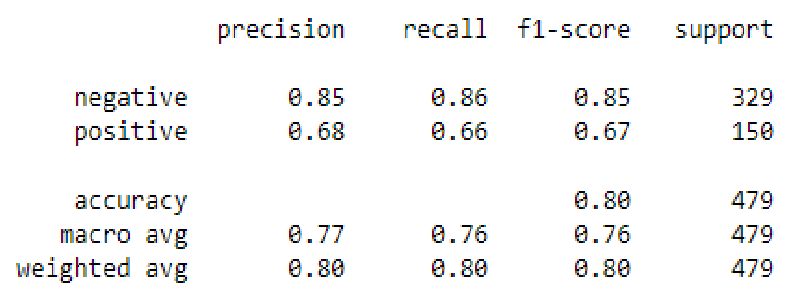
Confusion Matrix	Classification Report
	

```
[[402  53]
 [ 58 123]]
```

```
              precision    recall  f1-score   support

    negative       0.87      0.88      0.88       455
    positive       0.70      0.68      0.69       181

    accuracy                           0.83       636
   macro avg       0.79      0.78      0.78       636
weighted avg       0.82      0.83      0.82       636
```

Table 7: Experiment results for sentiment classification based on User data :-

METRICS	PERFORMANCE
Accuracy score	80%
Precision	0.76
Recall	0.76
F1-score	0.76

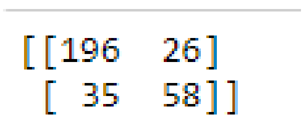
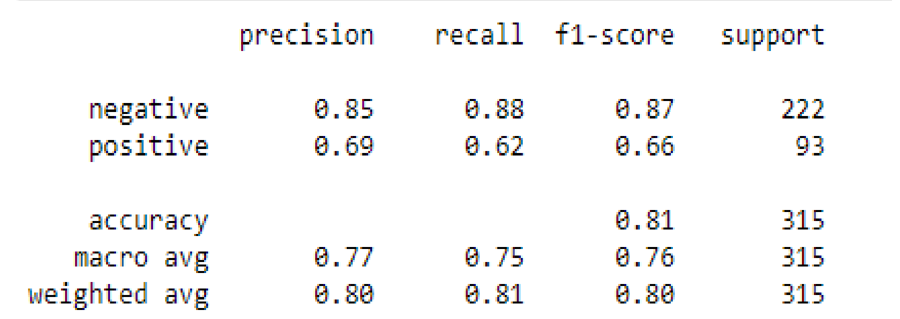
Confusion Matrix	Classification Report
	

```
[[283  46]
 [ 51  99]]
```

```
              precision    recall  f1-score   support

    negative       0.85      0.86      0.85       329
    positive       0.68      0.66      0.67       150

    accuracy                           0.80       479
   macro avg       0.77      0.76      0.76       479
weighted avg       0.80      0.80      0.80       479
```

Table 8: Experiment results for sentiment classification based on Bot data :-

METRICS	PERFORMANCE
Accuracy score	81%
Precision	0.76
Recall	0.75
F1-score	0.76

Confusion Matrix	Classification Report
	

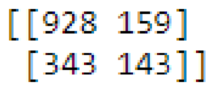
```
[[196  26]
 [ 35  58]]
```

```
              precision    recall  f1-score   support

    negative       0.85      0.88      0.87       222
    positive       0.69      0.62      0.66        93

    accuracy                           0.81       315
   macro avg       0.77      0.75      0.76       315
weighted avg       0.80      0.81      0.80       315
```

Table 9: Experiment results for sentiment classification training on User data and testing on Bot data :-

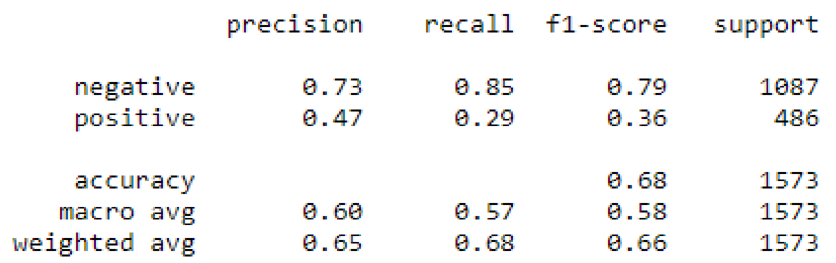
METRICS	PERFORMANCE
Accuracy score	68%
Precision	0.60
Recall	0.57
F1-score	0.57

Confusion Matrix



```
[[928 159]
 [343 143]]
```

Classification Report

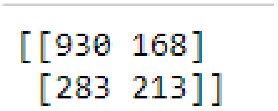


```
              precision    recall  f1-score   support

    negative       0.73      0.85      0.79      1087
    positive       0.47      0.29      0.36       486

    accuracy                           0.68      1573
   macro avg       0.60      0.57      0.58      1573
weighted avg       0.65      0.68      0.66      1573
```

Table 10: Experiment results for sentiment classification training on Bot data and testing on User data :-

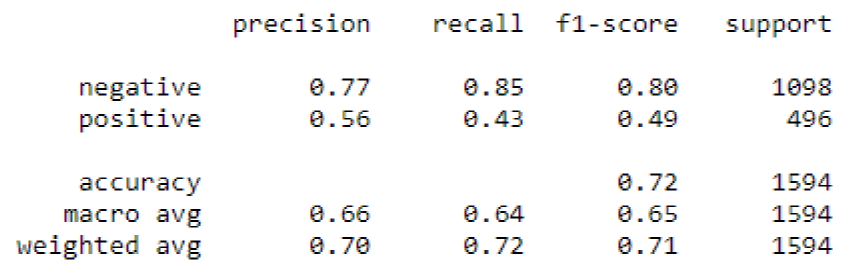
METRICS	PERFORMANCE
Accuracy score	72%
Precision	0.66
Recall	0.63
F1-score	0.64

Confusion Matrix



```
[[930 168]
 [283 213]]
```

Classification Report

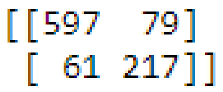
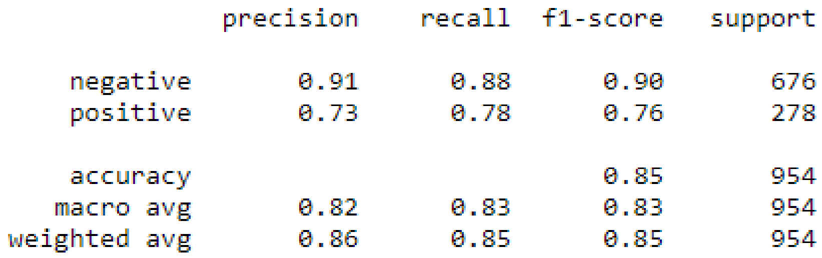


```
              precision    recall  f1-score   support

    negative       0.77      0.85      0.80      1098
    positive       0.56      0.43      0.49       496

    accuracy                           0.72      1594
   macro avg       0.66      0.64      0.65      1594
weighted avg       0.70      0.72      0.71      1594
```

5.3 LSTM Model

Table 11: Experiment results for sentiment classification based on User-Bot data :-

METRICS	PERFORMANCE
Accuracy score	85%
Precision	0.82
Recall	0.83
F1-score	0.82

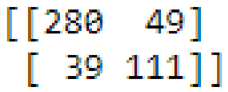
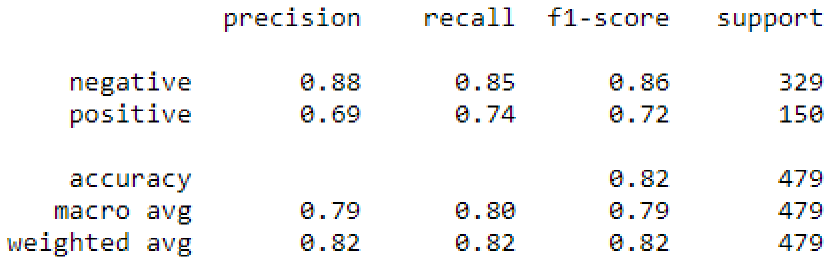
Confusion Matrix	Classification Report
	

```
[[597  79]
 [ 61 217]]
```

```
              precision    recall  f1-score   support

    negative       0.91      0.88      0.90       676
    positive       0.73      0.78      0.76       278

    accuracy                           0.85       954
   macro avg       0.82      0.83      0.83       954
weighted avg       0.86      0.85      0.85       954
```

Table 12: Experiment results for sentiment classification based on User data :-

METRICS	PERFORMANCE
Accuracy score	82%
Precision	0.78
Recall	0.79
F1-score	0.79

Confusion Matrix	Classification Report
	

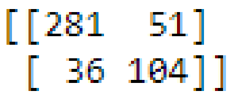
```
[[280  49]
 [ 39 111]]
```

```
              precision    recall  f1-score   support

    negative       0.88      0.85      0.86       329
    positive       0.69      0.74      0.72       150

    accuracy                           0.82       479
   macro avg       0.79      0.80      0.79       479
weighted avg       0.82      0.82      0.82       479
```

Table 13: Experiment results for sentiment classification based on Bot data :-

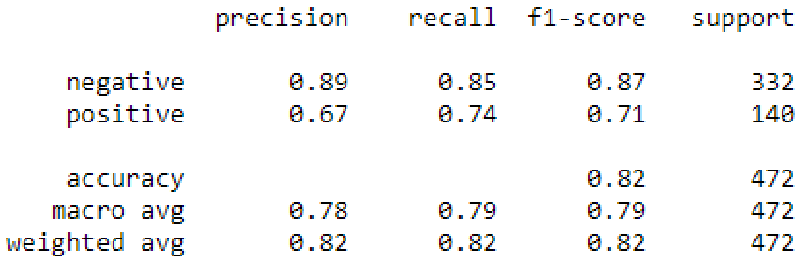
METRICS	PERFORMANCE
Accuracy score	82%
Precision	0.77
Recall	0.79
F1-score	0.78

Confusion Matrix



```
[[281  51]
 [ 36 104]]
```

Classification Report

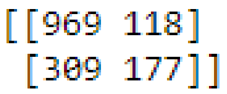


```
              precision    recall  f1-score   support

    negative       0.89      0.85      0.87       332
    positive       0.67      0.74      0.71       140

    accuracy                           0.82       472
   macro avg       0.78      0.79      0.79       472
weighted avg       0.82      0.82      0.82       472
```

Table 14: Experiment results for sentiment classification training on User data and testing on Bot data :-

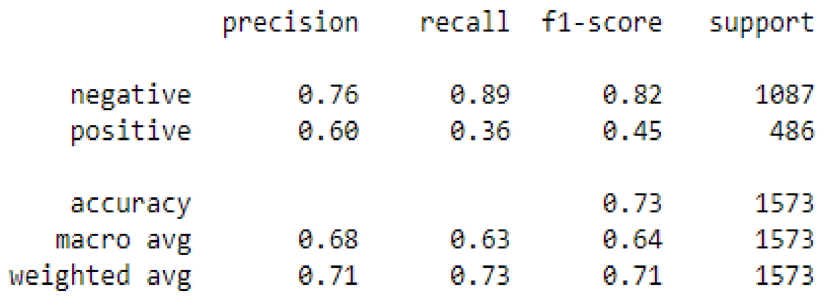
METRICS	PERFORMANCE
Accuracy score	73%
Precision	0.67
Recall	0.62
F1-score	0.63

Confusion Matrix



```
[[969 118]
 [309 177]]
```

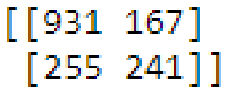
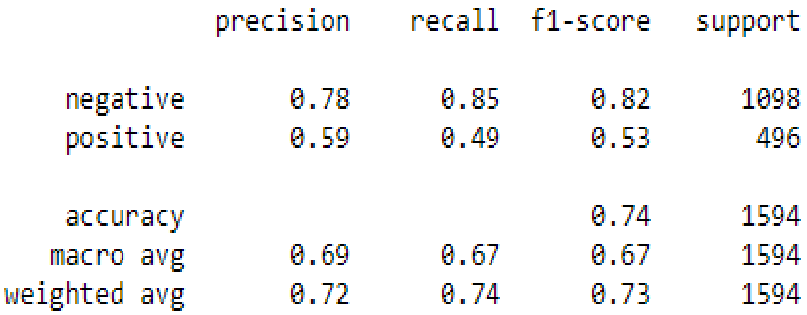
Classification Report



```
              precision    recall  f1-score   support

    negative       0.76      0.89      0.82      1087
    positive       0.60      0.36      0.45       486

    accuracy                           0.73      1573
   macro avg       0.68      0.63      0.64      1573
weighted avg       0.71      0.73      0.71      1573
```

Table 15: Experiment results for sentiment classification training on Bot data and testing on User data :-

METRICS	PERFORMANCE
Accuracy score	74%
Precision	0.68
Recall	0.66
F1-score	0.67

Confusion Matrix	Classification Report
	

Confusion Matrix

```
[[931 167]
 [255 241]]
```

Classification Report

```
              precision    recall  f1-score   support

    negative       0.78      0.85      0.82      1098
    positive       0.59      0.49      0.53       496

    accuracy                           0.74      1594
   macro avg       0.69      0.67      0.67      1594
weighted avg       0.72      0.74      0.73      1594
```

6. OBSERVATIONS AND DISCUSSIONS

6.1. **Supervised Learning Model**

6.1.1. Feature Analysis

The experiment's dataset is initially constructed. To preprocess the dialogue, lemmatization using the spacy library is performed along with the removal of punctuation and numbers, transforming the dialogue's text to lower case, and eliminating all stop words except for the not phrases.

Here, an object is created using the TfidfVectorizer() method, and we can then use that object to apply the fit transform() method to the "Statements/Dialogue" column. Therefore, we are creating the tfidf matrix, which is simply a sparse matrix, for the "Dialogue" column. Our input feature matrix looks like this.

We have sentiment values which are in the range from -5 to +5, which is our output value. Splitting the newly formed input feature matrix and 'Sentiment' column in the ratio **8:2**, or 80% training set and 20% testing set. Predict the Sentiment by training the Logistic Regression model. Making a note of the metrics for measuring **accuracy**, **precision**, **recall**, and **F1 score**.

6.2. Deep Learning Model

We started by importing every library required for the experiment, then I imported the xl file. Data for sentiment analysis was retrieved after import and stored in a data frame, which was afterwards converted to a pickle file for ease of use.

We developed a corpus of distinctive reviews for the **input feature matrix**. Following that, each word from the corpus was mapped, represented by their indexes, and then ordered as per dialogue, with pre-padded embedding sequences since it required a vector matrix of a specific size. Sentences are pre-padded to a maximum length. Sized at 3177 * 113, this feature matrix. This matrix is used in the Embedding layer while creating the deep learning models. Our output is the '**Sentiment**' column. Splitting the training set by 80% and the test set by 20% using a **8:2** split of the input feature matrix and output feature matrix. Create the Deep learning models - LSTM, RNN models. Predict the sentiment by training the model. Making a note of the metrics for measuring **accuracy**, **precision**, **recall**, and **F1 score**.

For the **RNN** model-based classification framework, we chose an **embedding layer with a size of 113**, **a SIMPLE RNN layer with a size of 100**, and **a size of 32 for the embedding vector features**. Finally, there is a **dense layer** with **sigmoid activation function**, which forecasts a value between **0 and 1.**

In the instance of the **LSTM** model-based classification framework, we chose **an embedding layer of size 113 as the first layer**, followed by **an LSTM layer of size 100**, with **a size of 32 for the embedding vector features**. The last layer is thick and has **sigmoid activation function**, which forecasts values between **0 and 1**.

Then, based on the actual and predicted values, we estimated the positive and negative sentiment, where a value of 0 indicates a negative sentiment and a value of 1 indicates a positive sentiment. and finally determined the result value, which is based on whether the actual and predicted numbers both are positive or negative.

6.3. Discussion on outputs

We can measure how much the results we achieved after making predictions, which can be either true value or false value, as we only measure positive and negative sentiments after training the models.

To calculate true value, we used the formula as :-

$$\textbf{True value} = \frac{\text{total number or “true” values in result}}{\text{total number or “true” values in result + total number or “false” values in result}}$$

$$\textbf{False value} = \frac{\text{total number or “false” values in result}}{\text{total number or “true” values in result + total number or “false” values in result}}$$

Using Logistics Regression we have calculated true and false value in various experiments, those are shown in below :-

Experiments	True Value (%)	False Value (%)
User-Bot Data	87.26%	12.73%
User Data	82.75%	17.24%
Bot Data	84.76%	15.23%
Training on User Data and Testing on Bot Data	63.69%	33.30%
Training on Bot Data and Testing on User Data	63.55%	36.44%

Using RNN we have calculated true and false value in various experiments, those are shown in below :-

Experiments	True Value (%)	False Value (%)
User-Bot Data	84.74%	15.25%
User Data	79.12%	20.87%
Bot Data	85.07%	14.92%
Training on User Data and Testing on Bot Data	71.01%	28.98%
Training on Bot Data and Testing on User Data	75.09%	24.90%

Using LSTM we have calculated true and false value in various experiments, those are shown in below :-

Experiments	True Value (%)	False Value (%)
User-Bot Data	86.37%	13.62%
User Data	82.04%	17.95%
Bot Data	82.62%	17.37%
Training on User Data and Testing on Bot Data	74.12%	25.87%
Training on Bot Data and Testing on User Data	72.83%	27.16%

7. ERROR ANALYSIS

When we perform prediction using the RNN model, then we notice that the actual sentiment is in "positive", but after prediction, the predicted sentiment is "negative" and vice versa. In the experiment of training on User data and testing on Bot data, we have got this kind of error.

Statements	Actual	Predicted	Results	Actual_new	Predicted_new
Submitting her photograph in the office does not seem to be a good idea. You do not need any of these options. These will do nothing but to warn her. Simply at the time of trial summon her employer with proofs of her employment and salary .	2	0	False	positive	negative
No, You cannot take any legal action against your mothers in law. Only Two options are possible.	0	1	False	negative	positive

In the experiment of training on Bot data and testing on User data, we have got this kind of error.

Statements	Actual	Predicted	Results	Actual_new	Predicted_new
Thanks this was really helpful.	1	0	False	positive	negative
Now we have all the docs with her maiden name and even our passports were updated accordingly.I was recently given an opportunity to work overseas(UAE) and in order to sponsor my family, I had to submit my marriage registration acquire_certificate.	0	1	False	negative	positive

Similarly, when we perform prediction using LSTM model, then we notice that the actual sentiment is in "positive", but after prediction, the predicted sentiment is "negative" and vice versa. In the experiment of training on User data and testing on Bot data, we have got this kind of error.

Statements	Actual	Predicted	Results	Actual_new	Predicted_new
You can not prove that your mothers in law is instigating your wife to misbehave with you, because as an adult lady like your wife should not be carried away by some one's instigation to misbehave with her own husband.	1	0	False	positive	negative
Apart from the above, you may file an application cum complaint U/s 340 of Cr.P.C. before the court where the matter is pending for her concealment of material facts and the court is bound to initiate preliminary inquiry against her. The court may even dismiss the main petition filed by her.	0	1	False	negative	positive

In the experiment of training on Bot data and testing on User data, we have got this kind of error.

Statements	Actual	Predicted	Results	Actual_new	Predicted_new
I don't want to give her divorce as it will damage my kids life.I want to get rid from this mental harassment.Can you please suggest me suitable steps?	1	0	False	positive	negative
Me & my wife filed a mutual divorce FIR.I have few questions on that , please help me out.	-1	1	False	negative	positive

8. CONCLUSION

We examine the sentiments for user and bot utterances in this project work utilizing machine learning and deep learning algorithms. Preprocessing is done by eliminating numerals and punctuation, turning the dialogue's text to lower case, deleting all stop words other than the not phrases, and lemmatization using the spacy library.

The deep learning models performed noticeably better than the supervised learning model due to feature selection, according to many tests on several supervised learning model like Logistic Regression and Deep learning models, LSTM, RNN. We also notice that LSTM performs best, with an accuracy of 85 percent, outperforming Simple RNN, whose accuracy is only 83 percent, since LSTM experiences considerably less vanishing gradient problem. We have achieved accuracy of 77 percent using Logistic Regression.

The following could be our future area of study :-

- Deep learning models will produce more effective and precise results if there are more utterances or dialogues, as well as sentiments, in the range of 10,000 or more.
- It is clear that neutral sentiment carries the most weight, followed by negative sentiment. It produces an unbalanced classification problem that can affect the effectiveness of machine learning or deep learning based models. We would like to use random sampling, oversampling, and under sampling in order to initially eliminate biases to get better result.

REFERENCES

1. Song L, Xin C, Lai S, Wang A, Su J, Xu K. CASA: Conversational Aspect Sentiment Analysis for Dialogue Understanding. Journal of Artificial Intelligence Research. 2022 Feb 9;73:511-33.

2. Rinaldi A, Oseguera O, Tuazon J, Cruz AC. End-to-end dialogue with sentiment analysis features. InInternational Conference on Human-Computer Interaction 2017 Jul 9 (pp. 480-487). Springer, Cham.

3. Li W, Shao W, Ji S, Cambria E. BiERU: Bidirectional emotional recurrent unit for conversational sentiment analysis. Neurocomputing. 2022 Jan 7;467:73-82.

4. Wang J, Wang J, Sun C, Li S, Liu X, Si L, Zhang M, Zhou G. Sentiment classification in customer service dialogue with topic-aware multi-task learning. InProceedings of the AAAI Conference on Artificial Intelligence 2020 Apr 3 (Vol. 34, No. 05, pp. 9177-9184).

5. An SH, Jeong OR. A study on the psychological counseling AI Chatbot system based on sentiment analysis. Journal of Information Technology Services. 2021;20(3):75-86.

6. Zhang Y, Tiwari P, Song D, Mao X, Wang P, Li X, Pandey HM. Learning interaction dynamics with an interactive LSTM for conversational sentiment analysis. Neural Networks. 2021 Jan 1;133:40-56.

7. Huddar MG, Sannakki SS, Rajpurohit VS. Attention-based multi-modal sentiment analysis and emotion detection in conversation using RNN.

8. Majid R, Santoso HA. Conversations Sentiment and Intent Categorization Using Context RNN for Emotion Recognition. In2021 7th International Conference on Advanced Computing and Communication Systems (ICACCS) 2021 Mar 19 (Vol. 1, pp. 46-50). IEEE.

9. Monika R, Deivalakshmi S, Janet B. Sentiment analysis of US airlines tweets using LSTM/RNN. In2019 IEEE 9th International Conference on Advanced Computing (IACC) 2019 Dec 13 (pp. 92-95). IEEE.

10. Wang X, Jiang W, Luo Z. Combination of convolutional and recurrent neural network for sentiment analysis of short texts. InProceedings of COLING 2016, the 26th international conference on computational linguistics: Technical papers 2016 Dec (pp. 2428-2437).