

## **Chapter One**

### **Introduction**

#### **1.1 Background to the Study**

Mental health, a crucial component of a satisfying existence, is influenced by emotional, psychological, and social well-being. Nonetheless, a considerable number of individuals continue to encounter difficulties in accessing prompt and personalised mental healthcare services. A significant proportion of the global population, nearly one billion individuals, have encountered a range of mental health disorders, including but not limited to anxiety, depression, psychotic, and personality disorders. Mental disorders result in a notable decline in the affected person's standard of living, as well as their societal and economic productivity. The world economy incurs an estimated annual cost of USD 2.5 trillion due to the collective negative impact, which is expected to increase to USD 6 trillion by 2030<sup>1</sup>. The World Health Organisation has underscored the pressing need for mental healthcare and treatment through its Special Initiative for Mental Health (2019-2023). The initiative seeks to provide access to high-quality and cost-effective care for mental health disorders to an additional 100 million individuals in 12 priority countries<sup>2</sup>.

A wide range of pharmacological and non-pharmacological interventions are commonly recommended for the treatment of mental illnesses. These interventions may include surgical procedures, in-patient and out-patient care, medication, support groups, counselling, psycho-social interventions, behavioural therapies, and alternative therapies<sup>1</sup>. The selection of appropriate intervention is contingent upon the classification of the ailment and its level of its severity. The World Health Organisation (WHO) has produced the International Classification of Diseases (ICD) and the Diagnostic and Statistical Manual of Mental Disorders (DSM-5) as formal

taxonomies for the classification of mental illnesses. These systems employ standardised criteria and common language for this purpose<sup>1,3</sup>.

The International Classification of Diseases (ICD) and the Diagnostic and Statistical Manual of Mental Disorders, Fifth Edition (DSM-5) respectively identify 11 and 20 principal classifications of psychiatric disorders<sup>2</sup>. Anxiety and depression are the most frequently occurring mental disorders, particularly among individuals of working age. The employment of psycho-social and behavioural interventions is widely acknowledged as the preferred non-pharmacological approach for managing anxiety and depression<sup>4</sup>.

Cognitive Behavioural Therapy (CBT) is a type of therapy that is focused on addressing problems and taking action. It has been extensively researched and is considered superior to other forms of psychotherapy. This is due to its alignment with the information processing condition of the human brain<sup>5</sup>. The utilisation of technological advancements has become more prevalent in expanding the implementation of cognitive-behavioral therapy (CBT) interventions that address psycho-social and behavioural concerns among a significant number of individuals.

Artificial intelligence is a field of technology that involves the study and development of theories, methodologies, processes, and systems that are capable of emulating human cognition<sup>6,7</sup>. This branch of study develops intelligent machines that function and provide responses comparable to those of the responses provided by the human intellect. Machine learning and deep learning are regarded as crucial subdomains of artificial intelligence<sup>8,9</sup>.

The integration of artificial intelligence (AI) in the healthcare industry has brought about a significant transformation. It has revolutionised the approach to patient care

by altering the methods of diagnosis, treatment, and monitoring. The implementation of this technology has significantly enhanced healthcare research and outcomes through the provision of precise diagnoses and the facilitation of individualised treatments. The capacity of AI in the healthcare sector to rapidly scrutinise extensive clinical documentation facilitates the identification of disease markers and patterns that may have otherwise gone undetected by medical practitioners<sup>10</sup>.

Conversational AI has received considerable attention in the field of natural user interface design. The field of machine learning has garnered significant attention from three research communities, namely natural language processing (NLP), information retrieval (IR), and machine learning (ML), owing to its rapid expansion. Despite this, the utilisation of deep learning (DL) and reinforcement learning (RL) methodologies in the context of conversational AI is widely regarded as a noteworthy accomplishment<sup>11</sup>. Consequently, researchers have redirected its focus towards enhancing end-to-end models for conversational AI systems.

Chatbots, an exemplar application of artificial intelligence, have demonstrated significant advantages in the healthcare sector by offering tailored support. Conversational agents, also known as chatbots, represent a novel technological advancement that is accessible around the clock via smartphones, facilitates communication with multiple users, and is immune to cognitive biases<sup>1</sup>. In contrast to computer-aided tools that necessitate human expertise, chatbots operate entirely online and in an automated manner, rendering them efficacious in delivering primary support for mental health disorders.

Chatbots employ natural language processing and have garnered significant attention due to their ability to offer a diverse range of services via a single interface. Chatbots

have been implemented in various industries such as education, banking, customer service, and healthcare to enhance their services and improve customer experience.

NLP is a subfield of AI and linguistic, devoted to make computers understand the statements or words written in human languages<sup>1</sup>. Chatbots have the potential to alleviate specific obstacles in the provision of mental health services, including the social stigma associated with seeking psychological assistance and the physical distance that may impede in-person counselling attendance<sup>12</sup>.

The field of mental health has witnessed a recent surge in the development of digital interventions that can serve as a complementary or alternative approach to conventional face-to-face mental health services<sup>12,13</sup>. The existing body of literature indicates that the utilisation of chatbots yields extensive advantages. It has been observed that certain individuals exhibit a preference towards utilising chatbots for the execution of specific tasks that are conventionally carried out on web pages or mobile applications.

Chatbots have the potential to alleviate specific obstacles in the provision of mental health services, including the social stigma associated with seeking psychological assistance and the physical distance that may impede in-person counselling attendance<sup>14</sup>. Individuals who work non-traditional work schedules may encounter challenges in obtaining mental health care services. The utilisation of chatbots presents a viable avenue for addressing these issues, given that mobile mental health interventions are unobtrusive and readily available to individuals possessing a mobile device<sup>13</sup>. Insufficient mental health workforce and limited access to mental health services are persistent issues globally. In this regard, automated procedures have

emerged as a promising solution for mental healthcare providers and individuals seeking mental health services<sup>13</sup>.

Numerous chatbots are currently accessible for utilisation in the provision of mental health care. Woebot, a chatbot therapist, is a well-known mental health chatbot<sup>15</sup>. The approach utilises cognitive behavioural therapy and employs various techniques to facilitate the enhancement of clients' mental well-being. A randomised controlled trial conducted among students revealed that the utilisation of Woebot led to a significant reduction in symptoms of depression within a span of two weeks<sup>15</sup>.

### **1.2 Statement of the Problem**

Despite the importance of mental health, many people still have trouble getting the timely and individualized mental health treatment they need which may be caused by social stigma, lack of available resources, or living in a remote location. Existing studies presented a chatbot system with low-complexity models and intelligence and were mainly about the user experience and were rarely evaluated with regards to the chatbot's performance. Also, most studies used various machine learning techniques, GD algorithms and NLP algorithms. Therefore, to overcome this low-complexity problem and low intelligence, this study seeks to use Recurrent Neural Network in developing a chatbot which offers an improved intelligent system.

### **1.3 Research Aim and Objectives**

This study aims to develop a web based chatbot for personalized therapy system for mental health. The specific objectives are to:

- i. Design a chatbot system that can effectively understand and interpret user inquiries related to mental health.
- ii. Implement the designed chatbot; and

iii. evaluation performance for the developed chatbot.

#### **1.4 Significance of the Study**

Development of a web based a chatbot for personalized therapy for mental health will play a critical and helpful role in a variety of mental health issues, including mild anxiety, depression, stress, and addiction. It can help develop emotional wellbeing and coping strategies in the face of stressful situations. In this way, the bot can act as a coach, encouraging the user to step outside their comfort zone or build helpful habits over time. Furthermore, the results of this study will open up more opportunities for further research on personalizing of chatbots and will contribute to the body of knowledge by proffering intelligent solutions to issues relating to mental health. Additionally, findings may result in the development of new theories regarding using artificial intelligence for health care.

#### **1.5 Scope of the Study**

The purpose of this thesis is to conceptualise, implement, and evaluate a mental health chatbot individualised therapy system. The purpose of this system is to provide those looking for mental health support with easy access to tailored advice and help. Using NLP and machine learning methods, the chatbot will be programmed to understand users' questions and provide helpful responses. The chatbot has the capability to offer a scalable machine learning pipeline for the models of the system, starting with raw data, and can provide real-time responses to users, aiding them with a subset of commonly asked inquiries. Also, the bot would be deployed on the web for user to interface with the AI mental bot. Before implementation, an assessment will be conducted to evaluate the chatbot technologies that are available. This will involve a careful consideration of the advantages and disadvantages of different frameworks.

The evaluation of the chatbot will encompass an assessment of its usage, performance, and capacity to augment user inquiries. The results will be presented and analysed in a descriptive manner. Furthermore, the bot's interactions will be analysed to deduce patterns that can be utilised to inform subsequent development and enhancement.

### **1.6 Operational Definition of Terms**

**Artificial Intelligence (AI):** Artificial intelligence is a field of technology that involves the study and development of theories, methodologies, processes, and systems that are capable of emulating human cognition.

**Chatbots:** Automated programs that make use of artificial intelligence technology to interact with human users of digital platforms such as web-based input forms and instant messengers. The chatbot employs machine learning to develop the ability to predict users' behaviour and respond to input stimuli in a manner like human-to-human conversation.

**Natural Language Processing:** NLP is a subfield of Artificial Intelligence and linguistic, devoted to make computers understand the statements or words written in human languages

## Endnotes

1. P Rathnayaka, N Mills, B Durnett, De Silva D, Alahakoon D, R Gray. *A mental health chatbot with cognitive skills for personalised behavioural activation and remote health monitoring*. Sensors. 2022 May 11;22(10):3653.
2. World Health Organization. *The WHO special initiative for mental health (2019-2023): universal health coverage for mental health*. World Health Organization; 2019.
3. D.A Regier, E.A Kuhl & D.J Kupfer. The DSM-5: Classification and criteria changes. *World psychiatry*, 2013;12(2)
4. M Oud, L De Winter, E Vermeulen-Smit, D Bodden, M Nauta, L Stone, M Den Heuvel, R Al Taher, I De Graaf, T Kendall, R Engels. *Effectiveness of CBT for children and adolescents with depression: A systematic review and meta-regression analysis*. **European psychiatry**. 2019 Apr;57:33-45.
5. D, Cristea, S.G Hofmann. *Why cognitive behavioral therapy is the current gold standard of psychotherapy*. Frontiers in psychiatry. 2018 Jan 29;9:4.
6. E Bryndin. *Creative communication safe ethical artificial intelligence in the era of technological development*. Software Engineering. 2020;8(3):13-23.
7. G.J Hwang, H Xie, B.W Wah, D Gašević. *Vision, challenges, roles and research issues of Artificial Intelligence in Education*. *Computers and Education: Artificial Intelligence*. 2020 Jan 1;1:100001.
8. P Ongsulee. *Artificial intelligence, machine learning and deep learning*. In 2017 15th international conference on ICT and knowledge engineering (ICT&KE) 2017 Nov 22 (pp. 1-6). IEEE.
9. M Hügler, P Omoumi, J.M van Laar, J Boedecker, T Hügler. *Applied machine learning and artificial intelligence in rheumatology*. *Rheumatology advances in practice*. 2020;4(1):rkaa005.
10. E.J Topol. *High-performance medicine: the convergence of human and artificial intelligence*. *Nature medicine*. 2019 Jan;25(1):44-56.
11. A Fuad, M -Yahya. *Recent developments in Arabic conversational AI: a literature review*. IEEE Access. 2022 Feb 28;10:23842-59.
12. K Kretschmar, H Tyroll, G Pavarini, A Manzini, I Singh, NeurOx Young People's Advisory Group. *Can your phone be your therapist? Young people's ethical perspectives on the use of fully automated conversational agents (chatbots) in mental health support*. *Biomedical informatics insights*. 2019 Feb;11:1178222619829083.
13. C Sweeney, C Potts, E Ennis, R Bond, M.D Mulvenna, S O'neill, M Malcolm, L Kuosmanen, C Kostenius, A Vakaloudis, G Mcconvey. *Can Chatbots help support a person's mental health? Perceptions and views from mental healthcare*

*professionals and experts*. ACM Transactions on Computing for Healthcare. 2021 Jul 15;2(3):1-5.

14. A Miner, A Chow, S Adler, I Zaitsev, Tero P, Darcy A, Paepcke A. *Conversational agents and mental health: Theory-informed assessment of language and affect*. In Proceedings of the fourth international conference on human agent interaction 2016 Oct 4 (pp. 123-130).
15. K.K Fitzpatrick, A Darcy, M Vierhile. *Delivering cognitive behavior therapy to young adults with symptoms of depression and anxiety using a fully automated conversational agent (Woebot): a randomized controlled trial*. **JMIR mental health**. 2017 Jun 6;4(2):e7785.
16. M Skjuve, P.B Brandtzæg. *Chatbots as a new user interface for providing health information to young people. Youth and news in a digital media environment–Nordic-Baltic perspectives*. 2018.
17. A.N Vaidyam, H Wisniewski, J.D Halamka, M.S Kashavan, J.B Torous. *Chatbots and conversational agents in mental health: a review of the psychiatric landscape*. **The Canadian Journal of Psychiatry**. 2019 Jul;64(7):456-64..

Do Not Copy, Lead City University, Nigeria

## **Chapter Two**

### **Literature Review**

#### **2.1 Conceptual Review**

This section presents a thorough exposition of the fundamental concepts employed in this study for the purpose of developing an intelligent chatbot. These concepts encompass various fields, such as mental health, artificial intelligence (AI), machine learning, sequential models, chatbots, natural language processing (NLP), cognitive architecture. Furthermore, a comprehensive literature review of prior works by various authors on chatbot development for mental health, psychiatric counselling, stress management, personalized health assistance, anxiety-depressive symptoms and work-related burnout, depression, anxiety, stress, insomnia, burnout, and work-related stress was performed. The Review also covered various approaches such as SAT counseling method, cognitive support through CBT-Bot, text-based chatbot, agent-based framework for chatbot, machine learning, Web-based text messaging application, high-level natural language understanding (NLU), and emotion recognition based on multi-modal approach, and mobile application with integrated chatbot that implements methods from cognitive behaviour therapy (CBT). This review has identified gaps in the existing literature and emphasised the contributions of this study to the field of knowledge.

##### **2.1.1 Mental Health**

Mental health is a multifaceted construct that encompasses emotional, social, and psychological well-being. Emotional well-being pertains to the attainment of a state of wellness, while social well-being pertains to the ability of an individual to function effectively within their community. Psychological well-being pertains to the effective functioning of an individual<sup>1</sup>.

The attainment of life satisfaction is dependent upon an individual's maintenance of a sound mental state. The World Health Organisation (WHO) has provided a definition of mental health that characterises it as a condition of optimal wellness, wherein an individual is capable of achieving their full potential, managing typical life stressors, engaging in productive and fruitful work, and making meaningful contributions to their community<sup>2</sup>.

Mental health refers to a condition of psychological wellness that empowers individuals to effectively manage the challenges of daily life, recognise and utilise their personal strengths, attain educational and occupational success, and make meaningful contributions to society<sup>3</sup>. The notion of health and well-being is intrinsically linked to a fundamental element that serves as a foundation for our capacity to form connections, make decisions and influence our environment.

Mental health is considered a fundamental entitlement of every individual. It is imperative for both individual and collective advancement, as well as for socio-economic progress. The concept of mental health extends beyond the mere absence of mental disorders. The phenomenon in question is situated on a multifaceted spectrum, which is subjectively perceived by individuals with varying levels of challenge and discomfort, and may result in divergent social and medical consequences<sup>4</sup>.

### **2.1.2 Artificial Intelligence**

Artificial intelligence refers to the emulation of cognitive processes that are typically associated with human intelligence, which is achieved through the use of computer systems and related technologies<sup>5</sup>. AI has various specific applications, such as expert systems, natural language processing, speech recognition, and machine vision.

Artificial intelligence (AI) systems operate through the process of assimilating substantial quantities of labelled training data, scrutinising the data for correlations and patterns, and subsequently utilising these patterns to generate forecasts

concerning forthcoming conditions<sup>6</sup>. Through this process, a chatbot can acquire the ability to produce realistic conversations with individuals by being exposed to textual examples, while an image recognition tool can develop the capacity to recognise and depict objects in images by analysing millions of instances. Recent advancements in generative artificial intelligence (AI) methodologies have demonstrated significant progress in generating authentic text, images, music, and other forms of media. The AI programming domain is primarily concerned with the development of cognitive abilities that includes the following<sup>6</sup>:

**Learning.** The area of AI programming that is concerned with the acquisition of data and the formulation of rules for its transformation into actionable information. Algorithms, also known as rules, furnish computing devices with a set of sequential instructions for accomplishing a particular task.

**Reasoning.** The facet of artificial intelligence programming that is concerned with the selection of an appropriate algorithm to achieve a specific objective.

**Self-correction.** The purpose of this aspect of AI programming is to iteratively optimise algorithms and guarantee their provision of the utmost precise outcomes.

**Creativity.** This aspect of artificial intelligence employs a combination of neural networks, rules-based systems, statistical methodologies, and other AI techniques to produce novel images, text, music, and concepts.

### 2.1.2.1 Types of Conversational AI

#### 1. Rule Based Chatbot

In a rule-based approach, a bot answers questions based on some rules on which it is trained on. The rules defined can be very simple to very complex. The creation of these bots are relatively straightforward using some rule-based approach, but the bot is not efficient in answering questions, whose pattern does not match with the rules on which

the bot is trained. However, these systems aren't able to respond to input patterns or keywords that don't match existing rules. One of such languages is AIML (Artificial Intelligence Markup Language): The AIML language's purpose is to make the task of dialog modeling easy, according to the stimulus-response approach. Moreover, it is a XML-based markup language and it is a tag based. Tags are identifiers that are responsible to make code snippets and insert commands in the chatterbot. AIML defines a data object class called AIML objects, which is responsible for modelling patterns of conversation.

Example of AIML Code,

**Basic Tags:**

1. **<aiml>**: Defines the beginning and end of an AIML document
2. **<category>**: Defines the knowledge in a knowledge base.
3. **<pattern>**: Defines the pattern to match what a user may input.
4. **<template>**: Defines the response of an Alicebot to user's input.

```
<aiml version="1.0.1" encoding="UTF-8"?>
<category>
  <pattern> HELLO BOT </pattern>
  <template>
    Hello my new friend!
  </template>
</category>
</aiml>
```

## 2. Retrieval Based Conversational AI

When given user input, the system uses heuristics to locate the best response from its database of pre-defined responses. Dialogue selection is essentially a prediction problem, and using heuristics to identify the most appropriate response template may involve simple algorithms like keywords matching or it may require more complex processing with machine learning or deep learning. Regardless of the heuristic used, these systems only regurgitate pre-defined responses and do not generate new output. With massive data available, it is intuitive to build a retrieval based conversational system as information retrieval techniques are developing fast.

Given a user input utterance as the query, the system searches for candidate responses by matching metrics.

The core of retrieval based conversational systems is formulated as a matching problem between the query utterance and the candidate responses. A typical way for matching is to measure the inner-product of two representing feature vectors for queries and candidate responses in a transformed Hilbert space. The modelling effort boils down to finding the mapping from the original inputs to the feature vectors, which is known as representation learning. There is two-step retrieval technique to find appropriate responses from the massive data repository. The retrieval process consists of a fast ranking by standard TF-IDF measurement and the re-ranking process using conversation-oriented features designed with human expertise. The systems to select the most suitable response to the query from the question-answer pairs using a statistical language model as *cross-lingual information retrieval*. These methods are based on shallow representations, which basically utilises one-hot representation of words.

Most strong retrieval systems learn representations with deep neural networks (DNNs). DNNs are highly automated learning machines; they can extract underlying abstract features of data automatically by exploring multiple layers of non-linear transformation. Prevailing DNNs for sentence level modelling include convolution neural networks (C-NNs) and recurrent neural networks (RNNs). A series of matching methods can be applied to short-text conversations for retrieval-based systems. Basically, these methods model sentences using convolutional or recurrent networks to construct abstractive representations. Although not all of these methods are originally designed for conversation, they are effective for short-

text matching tasks and are included as strong baselines for retrieval-based conversational studies.

### **2.1.3 AI in Mental Health**

AI is gaining a foothold across clinical practice. The utilisation of machine learning (ML) and deep learning (DL) has been found to yield higher levels of precision in the diagnosis of mental health conditions and in the prediction of patient outcomes<sup>7</sup>. Additionally, computer vision has facilitated advancements in the analysis of imaging data and the interpretation of non-verbal cues, such as facial expressions, gestures, eye gaze, and human pose. Furthermore, natural language processing (NLP) has contributed to the improvement of speech recognition and text analysis, which has been applied in the development of a chatbot software for simulating human conversations, as well as in the creation and comprehension of clinical documentation<sup>8</sup>. AI in health care is affordability, accessibility, efficiency, privacy and ease to open up and support for therapists. Some revolutionalization of AI in mental health include ;

- i. The process of scrutinising patient data to evaluate the likelihood of developing mental health disorders, categorise the disorders, and enhance the efficacy of treatment plans.
- ii. Conducting self-assessment and therapy sessions by keyword-triggered and NLP chatbots. The most popular AI-powered virtual therapists include Woebot, Replika, Wysa, Ellie, Elomia, and Tess.
- iii. Enhancing patient engagement to improve and personalize patient experience.
- iv. Equipping therapists with technology to automate daily workflows.

#### **2.1.4 Machine Learning**

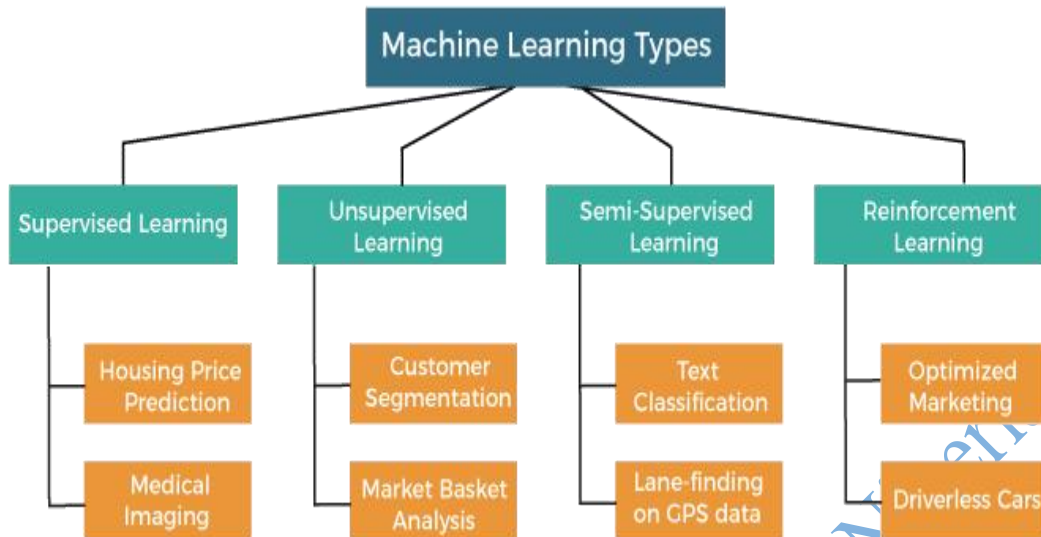
Machine learning is a subfield of AI that focuses on the study of computer algorithms designed to learn and improve automatically. Supervised learning, unsupervised learning, and reinforcement learning are the three categories that make up machine learning<sup>9,10</sup>. The goal of supervised learning is to produce a model that can predict an output by using historical observations of that output that have been labeled. Additionally, regression and classification are the two types of supervised learning that are categories based on the output value, which can be continuous or discrete<sup>11</sup>.

The process of regression involves fitting a model to the data that is provided, and it produces a continuous output. On the other hand, classification sorts the input data in order to produce the most useful output and generates discrete (or what is more commonly referred to as a class) output<sup>11,12</sup>. An unlabeled feature serves as the input for unsupervised learning, which then returns newly organized data by grouping, clustering, or organizing based on similarity measures such as distances<sup>13</sup>. Reinforcement learning is again different from both supervised and unsupervised learning. It learns by continuously optimizing an unknown reward function and updating its internal state based on some performance criterion<sup>14</sup>.

##### **2.1.4.1 Types of Machine Learning**

Machine learning is divided into four main types which are:

- i. Supervised Machine Learning
- ii. Unsupervised Machine Learning
- iii. Semi-Supervised Machine Learning
- iv. Reinforcement Learning



**Figure 2.1: Types of Machine Learning** <sup>13</sup>.

### 1. Supervised Machine Learning

In the context of supervised learning, machines undergo training using a dataset that has been labelled, and subsequently utilize this training to generate predictions. The labelled data denotes that certain inputs have already been mapped to their respective outputs. Initially, the machine is subjected to training using input and corresponding output data. Subsequently, the machine is tasked with predicting the output by utilizing the test dataset. The primary objective of the supervised learning methodology is to establish a correlation between the input variable (x) and the output variable (y)<sup>12</sup>. Several practical implementations of supervised learning include Risk Assessment, Fraud Detection, and Spam Filtering, among others. There are two distinct categories of problems in supervised machine learning, as follows:

Classification and Regression

**Classification:** Classification algorithms are utilised to address classification problems where the output variable is categorical in nature, such as binary categories like "Yes" or "No," or nominal categories like "Male" or "Female"<sup>13</sup>.

The classification algorithms are utilised to make predictions regarding the categories that are present within a given dataset. Instances of classification algorithms in practical applications include Spam Detection and Email Filtering, among others. The Random Forest Algorithm, Decision Tree Algorithm, Logistic Regression Algorithm, and Support Vector Machine Algorithm are among the commonly employed classification algorithms<sup>13</sup>.

**Regression:** Regression algorithms are frequently utilised to tackle regression problems that show a linear correlation between input and output variables. Regression models are employed for the purpose of predicting continuous output variables, such as market trends, weather patterns, and other related phenomena<sup>12</sup>. The Simple Linear Regression Algorithm, Multivariate Regression Algorithm, and Lasso Regression are among the frequently utilised regression algorithms. Supervised learning is a widely utilised technique in a range of fields, including but not limited to image segmentation, medical diagnosis, fraud detection, spam detection, and speech recognition<sup>12,13</sup>.

## **2. Unsupervised Machine Learning**

Unsupervised machine learning involves training a machine using an unlabeled dataset, whereby the machine is capable of predicting output without any form of supervision. The models are trained using unclassified and unlabeled data, and subsequently operate on this data in an unsupervised manner. The primary objective of the unsupervised learning algorithm is to classify the unstructured dataset based on similarities, patterns, and dissimilarities. The machines are programmed to identify concealed patterns within the input dataset<sup>12</sup>.

Unsupervised Learning can be categorised into two distinct types, as follows:  
Clustering and Association

**Clustering:** The clustering methodology is employed to identify the intrinsic clusters within the dataset. Cluster analysis is a method of categorising objects into groups based on their similarities, with the aim of ensuring that objects within a group share the most similarities while exhibiting a reduced number of resemblances or a complete absence of similarities with entities belonging to other groups<sup>15</sup>. There are some identified a number of commonly employed clustering algorithms, including the K-Means Clustering algorithm, Mean-shift algorithm, DBSCAN Algorithm, Principal Component Analysis, and Independent Component Analysis<sup>15</sup>.

**Association:** Association rule learning is an unsupervised machine learning methodology that aims to discover significant associations between variables in a vast dataset. The primary objective of this particular learning algorithm is to identify the interdependence between two data items and subsequently establish a correlation between these variables, thereby optimising the potential for profit generation<sup>16</sup>. Several widely used Association Rule Learning algorithms include the Apriori Algorithm, Eclat, and FP-growth algorithm. Unsupervised Learning finds various applications such as Network Analysis, Recommendation Systems, Anomaly Detection, and Singular Value Decomposition (SVD)<sup>16</sup>.

### **3. Reinforcement Learning**

Reinforcement learning is a feedback-driven process whereby an artificial intelligence agent, which is a software component, autonomously explores its environment through trial and error. It takes actions, learns from its experiences, and enhances its performance. Reinforcement learning is a type of machine learning that differs from

supervised learning in that it does not rely on labelled data. Instead, agents acquire knowledge solely through their experiences<sup>17</sup>.

The process of reinforcement learning bears resemblance to that of human learning, whereby a child acquires knowledge through experiential encounters in their daily routine. Reinforcement learning can be exemplified through gameplay, wherein the game serves as the environment, the actions taken by an agent at each step determine the states, and the objective of the agent is to attain a high score<sup>17</sup>. Agents are subject to feedback mechanisms that involve both punishment and rewards. Reinforcement learning has been utilised in various domains, including but not limited to Game theory, Operation Research, Information theory, and multi-agent systems, owing to its distinctive mode of operation.

The formalisation of a reinforcement learning problem can be achieved through the utilisation of a Markov Decision Process (MDP)<sup>18</sup>. Within the framework of Markov Decision Processes (MDP), the agent engages in ongoing interactions with the environment by executing actions. Following each action, the environment responds by generating a new state.

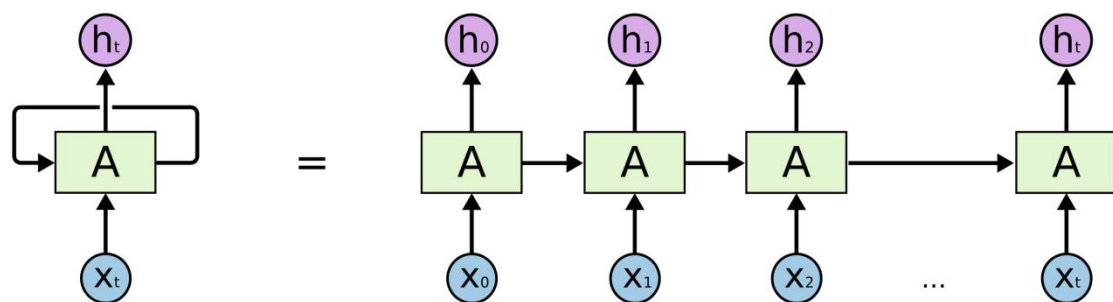
### **2.1.5 Sequence Models**

Sequence models refer to machine learning models that are designed to either input or output sequences of data. Sequential data encompasses various forms such as text streams, audio and video clips, as well as time-series data<sup>19</sup>. The Recurrent Neural Networks (RNNs) algorithm has gained popularity in the field of sequence models. Sequential models have been employed in diverse applications, such as speech recognition, sentiment classification, and video activity recognition<sup>20</sup>. The Recurrent Neural Network (RNN) is a widely used sequence model that has demonstrated effective performance for sequential data<sup>19</sup>.

### 2.5.1 Recurrent Neural Networks

A RNN is a type of neural network architecture where the neurons are interconnected in a way that allows for feedback signals to be transmitted between them. The Recurrent Neural Network is a Deep Learning algorithm that is specifically designed for the processing of sequential data<sup>20</sup>. Recurrent Neural Networks are predominantly employed within the domain of Natural Language Processing (NLP). Recurrent Neural Networks possess the ability to retain internal memory, rendering them highly effective for machine learning tasks that entail sequential data. Recurrent neural networks are also employed in the domain of time series prediction. One of the primary benefits of employing Recurrent Neural Networks is that they do not share features as is the case in conventional neural networks<sup>21</sup>.

In Recurrent Neural Networks, the weights are recurrently shared over time. Recurrent neural networks possess the capability to retain information from prior inputs and incorporate past data into their computations.



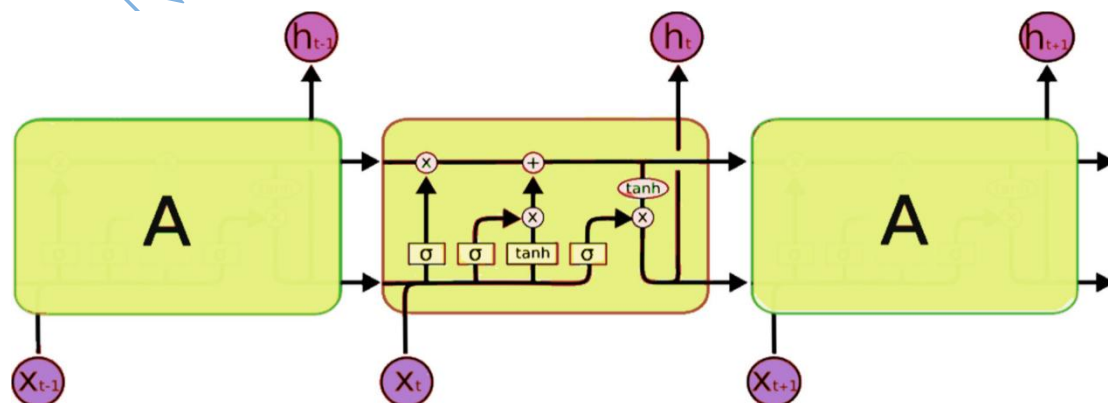
**Figure 2.2:** Recurrent Neural Network (RNN)<sup>21</sup>.

In the above diagram, a segment of the neural network, denoted as A, is responsible for processing an input variable  $X_t$  and generating an output value  $h_t$ . A loop facilitates the transfer of data from one stage of the network to the subsequent one<sup>21</sup>.

A Recurrent Neural Network (RNN) can be conceptualised as a series of interconnected networks, with each network transmitting information to its subsequent network. Recurrent Neural Networks (RNNs) are known to encounter the challenge of gradient vanishing and gradient exploding. The aforementioned phenomenon has resulted in the emergence of a novel neural network framework, denoted as the long short-term memory neural network architecture<sup>21</sup>. The design has demonstrated significant success and is increasingly being adopted for diverse sequential processing tasks.

### 2.1.5.2 Long Short Term Memory Network

LSTM networks represent a distinct category of Recurrent Neural Networks. Long Short-Term Memory (LSTM) networks are specifically engineered to acquire and model complex sequential patterns over extended periods of time. The recurrent neural networks commonly employ a repetitive module that exhibits a rudimentary configuration, typically comprising a solitary hyperbolic tangent layer<sup>21</sup>. Long Short-Term Memory (LSTM) networks exhibit a chain-like architecture, however, the recurrent module is characterised by a distinct structure. Rather than employing a solitary neural network layer, a configuration of four such layers is utilised, which interact in a highly distinctive manner, as depicted in Figure 2.3.



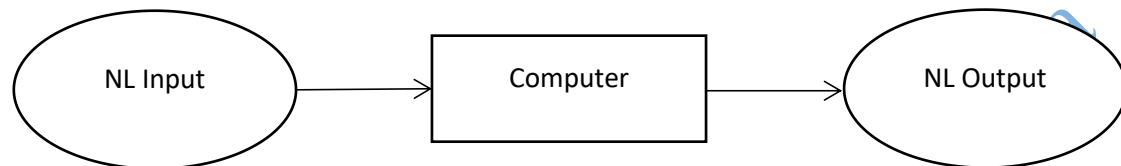
**Figure 2.3:** Long Short Term Memory networks (LSTM)<sup>22</sup>.

### 2.1.6 Natural Language Processing

Natural Language Processing is a specialised area within the field of Artificial Intelligence and linguistics that is focused on enabling computers to comprehend written or spoken human language.

#### Understanding

#### Generation



**Figure 2.4 Natural Language Processing**

Numerous Natural Language Processing (NLP) algorithms rely on machine learning, predominantly statistical machine learning. Various categories of machine learning algorithms have been utilised for natural language processing (NLP) tasks. The algorithms receive a vast array of features derived from the input source as their input. Certain algorithms developed in the past, such as decision trees, generated rigid systems of if-then rules that resembled the handwritten rule systems that were prevalent at that time.

The majority of chatbot systems undergo pre-processing of user input prior to its transmission to the Natural Language Understanding module. Preprocessing tasks are executed across multiple segments. The preprocessing stages of dialogue systems encompass several phases, namely sentence detection, co-resolution, tokenization, lemmatization, POS-tagging, dependency parsing, named entity recognition, and semantic role labelling<sup>21</sup>. Intents, Actions, and Entities are linguistic constructs that are derived from user inputs through the application of natural language processing methodologies.

### 2.1.7 Chatbots

A chatbot is a piece of software that allows for text-based communication utilising natural language<sup>23</sup>. Chatbots create the impression of engaging in human-like conversations by utilising natural language input to interact with computer software. The significance of simulating real individuals for chatbot acceptance by users is evident, emphasising the crucial importance of a comprehensive knowledge base, which refers to the pre-existing set of regulations that a chatbot possesses<sup>24</sup>.

Chatbots, in general, can be thought of as a combination of three parts: the interface, which is the interface between the chatbot and human users via a messaging app or a chat session on a website, the intelligence, which allows the chatbot to understand and solve queries and learn from each interaction, and the integration, which primarily covers integration with other systems and platforms<sup>25,26</sup>. In addition, chatbots have inputs and outputs. Inputs are best described as recording and entering data into a system and delivering instructions to the system. In order for a chatbot to function, the user's input into the conversation must match one of the pre-defined inputs so that the computer system can correctly interpret it<sup>27</sup>.

#### 2.1.7.1 History of Chatbot

**1. ELIZA:** ELIZA was the very first chatbot. It was created by Joseph Weizenbaum in 1966 and it uses pattern matching and substitution methodology to simulate conversation<sup>28,29</sup>. The program was designed in a way that it mimics human conversation. The Chatbot ELIZA worked by passing the words that users entered into a computer and then pairing them to a list of possible scripted responses<sup>28</sup>. It uses a script that simulated a psychotherapist. The script proved to be a significant impact on natural language processing and unnatural intelligence, with copies and variants protruding up at academies around the country.

The purpose of the very first chatbot produced was solely to entertain and engage users. Due to its employment as a psychotherapist, users frequently have a very emotional attachment with the computer, as characterized by the ELIZA-effect, which attributes human intelligence to chatbots<sup>30</sup>. ELIZA employs keyword matching, which implies that following a particular input, the programme will check for keywords that match. If matching keywords are located, the system will generate an answer based on the rules specified for this term; otherwise, a connected remark is recalled<sup>36</sup>. Consequently, ELIZA does not necessarily comprehend the user's difficulties; she just matches their responses to her usual responses<sup>30</sup>.

```

Welcome to
          EEEEEEE LL      IIII  ZZZZZZ  AAAAA
          EE      LL      II     ZZ     AA  AA
          EEEEE  LL      II     ZZZ   AAAAAAA
          EE      LL      II     ZZ     AA  AA
          EEEEE  LLLLLL  IIII  ZZZZZZ  AA  AA

Eliza is a mock Rogerian psychotherapist.
The original program was described by Joseph Weizenbaum in 1966.
This implementation by Norbert Landsteiner 2005.

ELIZA: Is something troubling you ?
YOU:   Men are all alike.
ELIZA: What is the connection, do you suppose ?
YOU:   They're always bugging us about something or other.
ELIZA: Can you think of a specific example ?
YOU:   Well, my boyfriend made me come here.
ELIZA: Is it important to you that your boyfriend made you come here ?
YOU:   He says I'm depressed much of the time.
ELIZA: I am sorry to hear that you are depressed.
YOU:   It's true. I am unhappy.
ELIZA: Can you explain what made you unhappy ?
YOU:

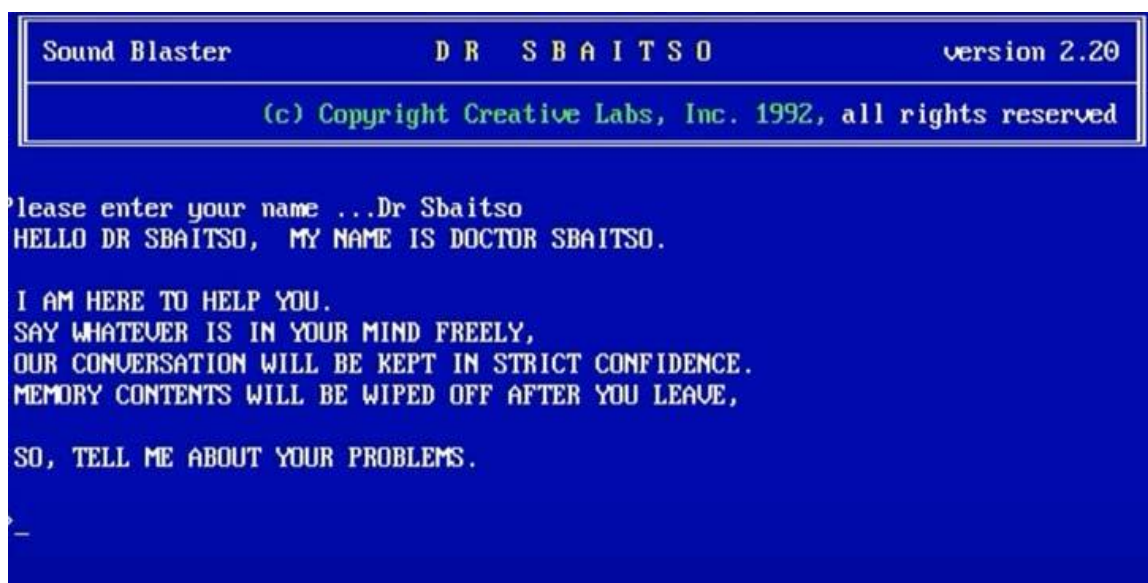
```

**Figure 2.5: Eliza<sup>32</sup>.**

**2. PARRY:** PARRY was constructed by American psychiatrist Kenneth Colby in 1972<sup>33</sup>. The program imitated a patient with schizophrenia. It attempts to simulate the disease. It is a natural language program that resembles the thinking of an individual. PARRY works via a complicated system of assumptions, attributions, and “emotional responses” triggered by changing weights assigned to verbal inputs<sup>34</sup>. To validate the work, PARRY was tested using a variation of the Turing test.

**3. Jabberwacky:** The chatbot was created by developer Rollo Carpenter in 1988<sup>35</sup>. It aimed to simulate a natural human conversation in an entertaining way. Jabberwacky has led to other technological growth. Some individuals use it for academic research purposes through its webpage since its origin. The chatbot is considered to use an AI technique called “contextual pattern matching<sup>35</sup>.”

**4. Dr. Sbaitso:** Dr. Sbaitso is a chatbot created by Creative Labs for MS-Dos in 1992<sup>36</sup>. It is one of the earliest efforts of incorporating A.I. into a chatbot and is recognized for its full voice operated chat program. The program would converse with the user as if it was a psychologist<sup>36</sup>. Most of its responses were along the lines of “Why do you feel that way?” rather than any sort of complicated interaction



**Figure 2.6: Dr. Sbaitso<sup>37</sup>.**

**5. ALICE :** ALICE, the Artificial Linguistic Internet Computer Entity, was constructed and utilised for the first time in 1995<sup>38</sup>. ALICE retains its knowledge of English conversation patterns in AIML files, Artificial Intelligence Markup Language, as opposed to ELIZA's basic keyword matching<sup>39</sup>. These AIML files consist of data items known as AIML objects, which are composed of themes and categories<sup>7</sup>. The

subjects have a name attribute and a set of associated categories, whereas categories are the fundamental unit of knowledge in AIML. Each category acts as a rule for aligning the user's input with the desired result, as well as a pattern and template<sup>30</sup>.

**6. SmarterChild:** The SmartChild was in many ways a precursor of Siri and was developed in 2001<sup>40</sup>. The chatbot was available on AOL IM and MSN Messenger with the strength to carry out fun conversations with quick data access to other services.

**7. Siri :** Siri was formed by Apple for iOS in 2010; it is an intelligent personal assistant and learning navigator that uses a natural language UI<sup>41</sup>. It paved the system for all AI bots and PAs after that. A patent application by the United States Patent and Trademark Office details a new Apple service where users could make inquiries and conversation with Siri through Messages. Similar to other texting and Facebook Messenger Apple's patent describes a Siri that could perform current duties without the user having to chat aloud. That could be helpful in several public spheres. They could reply to a text, audio, images, and video when transferred to it by the user<sup>41</sup>. This would result in more fruitful interactive experience among a consumer and a digital assistant. The patent provides a few examples of a conversation held between Siri and a user in Messages, with the user asking questions.

**8: Google Now/Google Assistant:** Google Now was launched at Google Inc in 2012<sup>42</sup>. It answers questions, performs actions through requests made to a set of web services and makes recommendations. It was part of a package of updates and UI modifications for mobile search, which included a female-voiced portable assistant

to compete with Apple's Siri. Google Now was initially a way to get contextually appropriate information based on location and time of the day<sup>42</sup>.

It evolved to become much more complicated and elaborate, with a broad range of content categories delivered on cards. Sometimes it refers to us as predictive search. Currently, it's built for use in smartphones and has been upgraded to accommodate several features. Google Now was replaced by Google Assistant in 2017. Today, the assistant is part of a more aggressive Google search growth strategy.

**9. Cortana:** Cortana was first demonstrated at Microsoft's Build 2014 developer conference, and it became directly integrated into both Windows phone devices and Windows 10 PCs<sup>43</sup>. This program uses voice recognition and relevant algorithms to get and respond to voice commands. For someone to get started, they must type a question in the search box, or select the microphone and talk to Cortana<sup>44</sup>. If a person is not very sure of what to say, they will see suggestions on the lock screen, as well as in Cortana home by selecting the search box on the taskbar. Cortana can perform tasks like reminders based on time, places, or people, send emails and texts, create and manage lists, chit-chat, and play games, find facts, files, locations, and info among others<sup>44</sup>.

**10. Alexa:** Alexa is an intelligent personal assistant developed by Amazon<sup>45</sup>. It was introduced in 2014 and is now built in to devices such as the Amazon Echo, the Echo Dot, the Echo Show and more. There is also an Alexa app and more devices from third-party manufacturers that have Alexa built in to them. All you have to do is say "Alexa, play some music" or "Alexa, find me an Italian restaurant" and she will help you out. Using nothing but the sound of your voice, you can search the Web, play music, create to-do or shopping lists, set alarms, stream podcasts, play audiobooks, get news or weather reports, control your smart-home products and

more. To add to the capabilities of any Alexa-enabled device, Amazon allows developers to build and publish skills for Alexa using the Alexa Skills Kit (ASK). You can download skills for free with the Alexa app<sup>45</sup>.

**11. ChatGPT:** ChatGPT is a large language model trained by OpenAI<sup>46</sup>. It was founded by the OpenAI team in 2021. It was designed to assist users in generating human-like text based on given input. ChatGPT can be used for a variety of tasks, including conversation generation and language translation<sup>47</sup>. The model is trained on a massive amount of data, allowing it to generate text that is often difficult to distinguish from text written by a human. ChatGPT has the ability to generate natural-sounding text and has potential applications in a variety of fields.

#### **2.1.7.2 Types of Chatbots**

Chatbots are generally divided into groups based on their roles, the jobs they complete, or how they communicate with users, including menu-based and text-based engagements. These factors affect the fundamental design principles of chatbots<sup>48,49</sup>.

The general-purpose chatbot, typically integrated into mobile, desktop, and smart speakers, is a multitasking agent and a personal virtual assistant<sup>50</sup>. Users can ask general questions like the current weather, the location of the closest restaurants, opening email programmes, adjusting the calendar, and other general questions. Virtual assistants that serve a general purpose include Siri, Bixby, Cortana, Alexa, and Google Home<sup>51</sup>.

Task-specific chatbots carry out certain activities for their users as assistant agents for particular fields, such as online tutoring, therapists, and customer support. These agents are incorporated into social media platforms as well as web pages. For instance, businesses frequently employ Facebook Messenger chatbots to target customers<sup>52</sup>.

### **2.1.7.3 Agents in Mental Health**

Chatbots have shown great potential in the field of mental health by providing accessible and scalable support to individuals. Here are some applications of chatbots in mental health:

1. **Emotional Support:** Chatbots can offer a listening ear and provide emotional support to individuals experiencing stress, anxiety, or other mental health issues. They can engage in conversations, offer empathy, and provide coping strategies.
2. **Screening and Assessment:** Chatbots can administer mental health screenings and assessments to help identify potential mental health concerns. By asking targeted questions, they can gather information and provide recommendations for seeking appropriate professional help if needed.
3. **Psychoeducation and Information:** Chatbots can deliver psychoeducational content and information about mental health conditions, symptoms, treatment options, self-help techniques, and resources. They can serve as a valuable source of information and raise awareness about mental health.
4. **Self-help Tools and Interventions:** Chatbots can offer interactive self-help tools and interventions, such as guided relaxation exercises, mindfulness practices, cognitive-behavioral techniques, and stress reduction strategies. They can assist individuals in managing their mental health on a day-to-day basis.
5. **Crisis Intervention:** Chatbots can provide immediate support during mental health crises by offering resources, helpline numbers, or suggestions for self-care. They can help individuals access the help they need in critical situations.
6. **Continuity of Care:** Chatbots can act as a bridge between therapy sessions or appointments by providing ongoing support and reminders for self-care activities.

They can reinforce therapy concepts and help individuals maintain progress between sessions.

It's important to note that while chatbots can be a valuable tool, they are not a substitute for professional mental health care. They can complement existing services and offer support, but individuals in need of serious or complex mental health support should consult with qualified mental health professionals.

Chatbots in the field of mental health care have been developed to support interpersonal skills as a training component of a depression treatment program. In addition, chatbots specialized to cope with stress problems have been also studied. A self-help program MYLO in the form of a chatbot has been developed. MYLO led to relief of pain, depression, anxiety, and stress and MYLO was thought to be more useful for problem solving<sup>53</sup>. Besides that, an automatic conversational chatbot “Woebot” using Facebook messenger has been developed based on CBT<sup>53</sup>.

As a result of an evaluation experiment using “Woebot” for college students, it was found that the participants’ depressive symptoms were significantly reduced. They commented that using “Woebot” was more receptive than traditional therapies. Existing studies and reviews show that mobile apps with integrated CBT can be successfully used for the treatment of psychiatric disorders<sup>54</sup>. Clinical outcomes for the use of mental health chatbots are still rare. The authors identified 41 different chatbots in mental health, mainly implemented as rule-based and stand-alone software, e.g. Wysa and WoeBot. Chatbots process the user input, and offer responsive, guided conversations and advice to help users in current mental health challenges. The bots normally ask a user on a daily basis on his emotions, thoughts, and behaviour. Some

systems passively track users' movements via the accelerometer integrated in the phone.

The chatbot Wysa provides a mood tracker and can detect negative moods<sup>55</sup>. It suggests a depression test and recommends seeking professional help. To support the relief of anxiety, depression and stress, there are mindfulness meditation exercises integrated in the app. The chatbot was tested in a study with a total of 129 participants. The participants were divided into two groups (frequent and occasional users). Another chatbot was developed called SERMO with integrated CBT interventions in German enabling unrestricted natural language user input<sup>56</sup>. It differs from the available systems by integrating natural language processing (NLP) and emotion analysis methods in order to automatically determine emotions from the user input. In contrast to existing decision-tree based systems, the system does not rely on strict patterns, but on syntactic and semantic similarities between user input and stored expressions.

#### 2.1.7.4 Chatbot Architecture

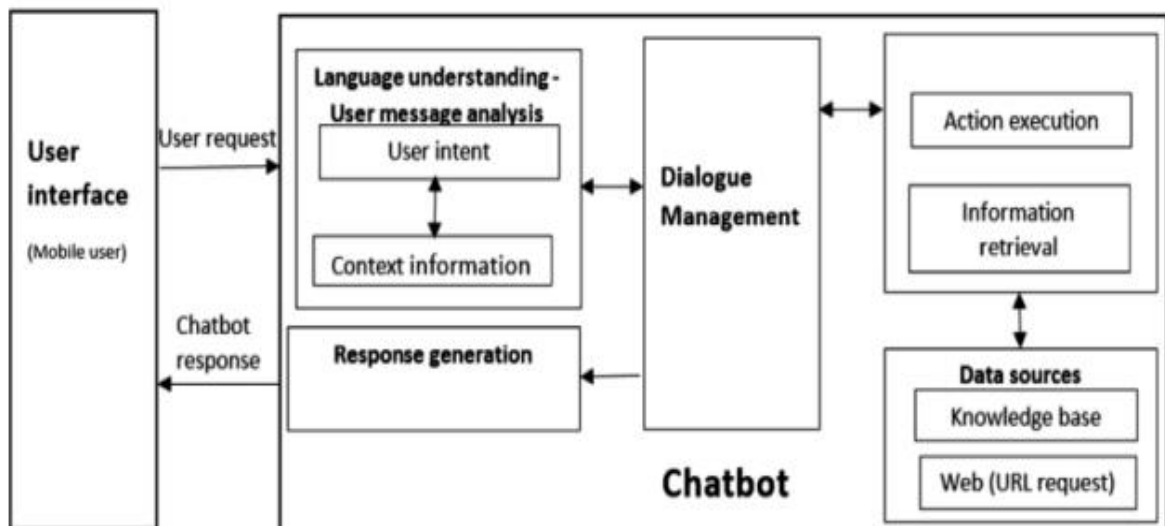


Figure 2.7: General Chatbot Architecture<sup>57</sup>.

**1. User Interface:** The user interface permits users to converse and interact with a chatbot using messenger programmes such as Facebook Messenger, Cortana, and Slack<sup>58</sup>. A chatbot's operation begins with a user's request. For voice-based conversational agents, an automated speech recognition (ASR) system will convert the user's speech input into text<sup>58</sup>. A speech-based conversational agent is also equipped with a text-to-speech (TTS) system that turns text back into speech once a text response has been created. Text-to-speech entails two phases: text analysis and waveform generation. Text analysis entails normalising the text and doing phonetic (pronunciation) and prosodic (phrasing, pitch, loudness, tempo, and rhythm) analysis, whereas waveform synthesis is selecting the prerecorded speech that fits the most criteria<sup>58</sup>.

**2. Natural Language Understanding:** After receiving the user's request, the system employs the NLU component to extract information from the input and provide a representation of its meaning that can be used in subsequent steps<sup>60</sup>. NLU typically performs three tasks: categorization of dialogue acts, classification of intent, and slot filling<sup>58</sup>. Classification of dialogue acts involves establishing the function of the user's input, or, more precisely, translating the user's utterance to a particular dialogue act type. The utterance may be categorised as a question, a statement, an offer, or another form of dialogue act. To better appreciate the user's request and determine an appropriate response, it is crucial to understand the current dialogue act<sup>58</sup>. The intent classification identifies the user's principal objective. Intentions are generally domain-specific. For instance, a request could involve meal ordering, hotel reservations, weather forecasts, etc. The aim of an agent within the domain of hotel reservations can be to make, cancel, or modify a reservation, and similarly, the intent of an agent within the domain of food ordering can be to place, query, or modify an order<sup>58</sup>.

Slot filling is the final step of natural language understanding (NLU). Other relevant details are extracted by the agent, which, when paired with the dialogue act and the user's purpose, enable it to completely comprehend the user's request<sup>58</sup>. There are various approaches for slot filling.

- i. **Regular Expression:** is a tool for describing text patterns. Each pattern is specified by a sequence of characters and each character is either a regular character or a metacharacter that has a special meaning<sup>58</sup>.
- ii. **Tokenization** entails dividing a text into units based on words, punctuation, or numbers<sup>58</sup>. Since English words are typically separated by white space, tokenizing an English text should be quite simple. However, there are some problematic instances in which the borders between words or phrases are unclear. Special cases include contracted elements (e.g. isn't), phrases (e.g. San Francisco), abbreviations (e.g. PhD. ), and acronyms (e.g. AT&T)<sup>58</sup>. Penn Treebank tokenization standard is a frequently used tokenization standard<sup>61</sup>. According to the Penn Treebank tokenization standard, contractions and punctuation are split between tokens, although hyphenated words are maintained together. Using data to determine the tokens automatically is a more complex method of tokenizing text, and it is particularly beneficial when dealing with unfamiliar words<sup>58</sup>. For example, bigger can be identified even if the training corpus contains only the terms "big," "little," and "smaller." Typically, this is accomplished by inducing a vocabulary from a row training corpus and then segmenting the text into tokens from the vocabulary<sup>58</sup>. The vocabulary consists of subwords, such as "little" and "er" rather than "smaller." Byte-pair encoding, unigram language modelling, and WordPiece are the three most often utilised algorithms<sup>62,63</sup>.

- iii. Text Normalization:** refers to the process of transforming text to a standard format. Word normalisation is the process of identifying words with the same meaning but various spellings (e.g., G.R.A. and GRA). Case folding is the process of converting anything into lowercase letters<sup>59</sup>.
- iv. Bag of Words:** the vector space model, also regarded as one of the simplest ways to input analysis, is also known as the vector space model. We count the frequency of each word but disregard syntactic and word order information. This can be achieved by normalising the text and removing stop words, or words that do not contribute to the meaning of the sentence<sup>58</sup>. When the meaning of a document depends on linguistic information, such as grammar or word order, the bag-of-words technique is inadequate. For instance, "John is taller than Mike" and "Mark is taller than Luke" have the same representation but have distinct meanings<sup>59</sup>.
- v. Latent Semantic Analysis (LSA):** or Latent Semantic Indexing, does not account for any linguistic knowledge. However, LSA compares the meaning behind the words rather than the words themselves, and keywords that frequently appear in the same context are grouped together<sup>58</sup>. To determine word similarity, a matrix is created in which each row represents a term, each column represents a document, and each cell indicates the frequency of the word in document<sup>58</sup>. The matrix is subsequently transformed by applying Singular Value Decomposition to minimise its dimensionality<sup>58</sup>.
- vi. Part-of-speech (POS) Tagging:** entails marking up each word in a text with a tag indicating its syntactic function, such as whether it is a verb, noun, pronoun, conjunction, etc. POS tagging is particularly beneficial when we encounter terms with several meanings and their application in a sentence may

be ambiguous<sup>58</sup>. For instance, the word "ring" can be either a noun or a verb, but in the sentence "ring a bell," it should be classified as a verb. Using a POS tag in such instances can eliminate the ambiguity<sup>64</sup>.

**3. Dialogue Management:** The Dialogue Management component handles information flowing from other components and is responsible for controlling and modifying the context of discussions as well as governing the chatbot's activities<sup>64</sup>. DM coordinates with other modules and encompasses the speech application's logic. This indicates that it is a crucial component of the chatbot architecture, and we must make deliberate and considered design decisions<sup>64</sup>. Unfortunately, building a resilient DM strategy is not a simple task, as it is difficult to predict which system behaviour would result in high user satisfaction. In this paragraph, we explore two design issues of DM that arise frequently: interaction methods and confirmation strategy selection. A dialogue system's interaction approach determines who controls the discourse. The dialogue might be directed by the user, the system, or a combination of both. When the interaction is user-directed, the user takes the lead, and the system simply responds to the user's questions and orders. When the interaction is system-directed, the system takes the lead, and the user simply responds to its questions. When the dialogue is mixed-initiative, both the user and the system are able to take the initiative<sup>59</sup>. The DM monitors the condition of the discussion or the system's conversational behaviour, which is based on the interaction strategy it adopts<sup>64</sup>.

**4. Backend Chatbots:** Retrieve from the Backend the information required to complete the relevant tasks, and then transmit the message to the Dialogue Management Component and the Response Generation Component<sup>64</sup>. The handmade rules must be stored in a Knowledge Base (KB) for rule-based chatbots. To ensure the robustness of the chatbot, the Knowledge Base rules should be as numerous and

comprehensive as feasible<sup>66</sup>. A chatbot can utilise an RDB to recall earlier talks. Taking into consideration prior information enables the chatbot to interact with greater consistency, precision, and dependability<sup>61</sup>. The development of the KB is crucial and required, but it can be time-consuming and stressful because to the manual labour involved. In order to solve this obstacle, developers devised a method that can automatically generate a new KB from an existing chatbot's KB<sup>62</sup>.

Frequently, rule-based chatbots direct users and populate the KB with their responses<sup>68</sup>. Recently, there has been a tendency toward retaining the information contained in digitally saved texts on the Internet and other online sources. Large data repositories that store data in a machine-readable and accessible format<sup>58</sup>. Google's Knowledge Graph, DBpedia, Freebase, and Wolfram Alpha are some examples of these knowledge bases<sup>60</sup>.

**5. Response Generation:** Once the required information has been gathered, the conversation system determines the response's content and the optimal way to convey it<sup>58</sup>. The Response Generation component is responsible for creating responses in a format that is easily understood by the end user. Five steps of processing comprise the RG pipeline: signal analysis, data interpretation, document planning, microplanning, and implementation<sup>59</sup>. Analysis of signals and interpretation of data are preliminary steps. These two stages are traversed by inputs in the form of data, such as numerical data in weather forecasting. Using pattern matching algorithms, the signal analysis finds data patterns and trends<sup>59</sup>.

Data interpretation employs symbolic reasoning techniques based on subject expertise to comprehend complicated messages and message relationships. Document planning can be subdivided into two more phases: content determination and discourse planning<sup>58</sup>. Determining content entails deciding what information must be supplied

to the user. The system sends only a portion of the obtained data, and it may communicate the data in phases. Whenever necessary, determining content also includes filtering and summarising the information. The objective of discourse planning is to organise and structure the text to facilitate human comprehension<sup>58</sup>. Microplanning, often known as sentence planning, is the process of composing the information-delivery sentences. It includes three tasks: referential expressions, aggregation, and lexical selection<sup>58</sup>.

## 2.2 Methodological Review

This section discusses the algorithm used in the design of the chatbot. The picture below shows that how Deep learning based chatbot works

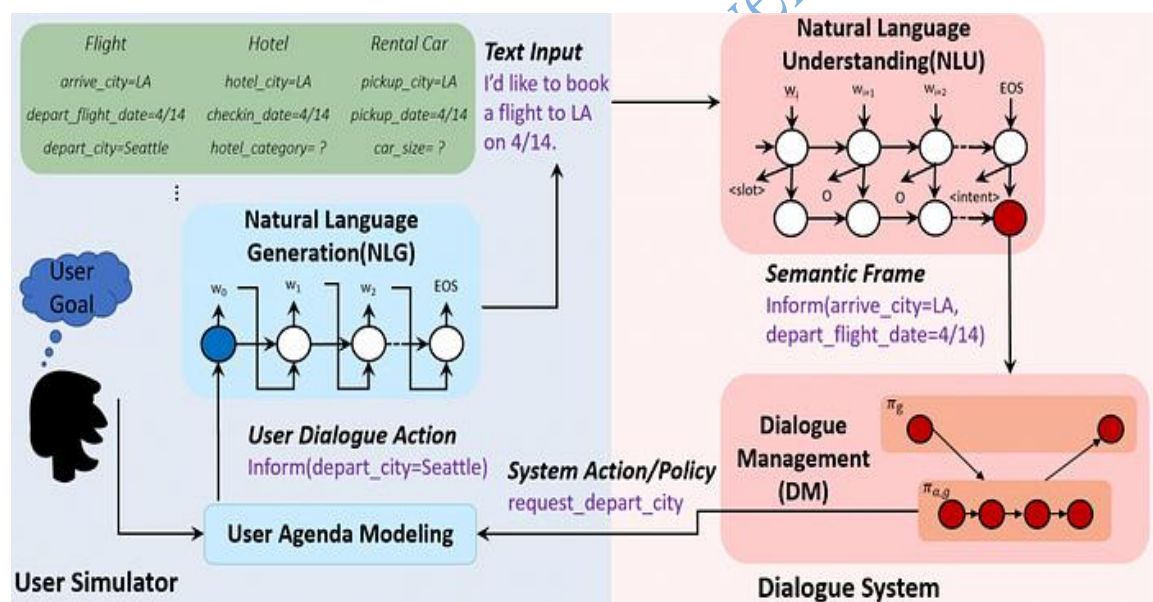


Figure 2.8: Deep Learning Based Chatbot<sup>70</sup>.

### 2.2.1 Natural Language Understanding

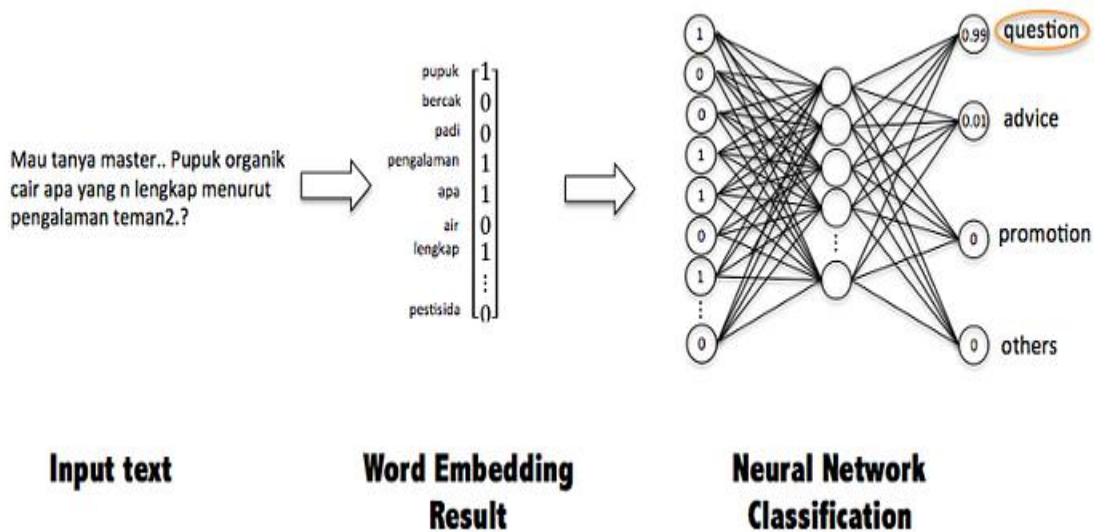
The NLU unit is responsible for transforming the user utterance to a predefined semantic frame according to the system's conventions, i.e. to a format understandable for the system<sup>71</sup>. This includes a task of slot filling and intent detection. For example, the intent, could be a greeting, like Hello, Hi, Hey, or it could have an inform nature, where the user is giving some additional information. Depending on the interests, the

slots could be very diverse, like the actor name, price, start time, destination city etc. The task of slot filling and intent detection is seen as a sequence tagging problem.

For this reason, the NLU component is usually implemented as an LSTM-based recurrent neural network with a Conditional Random Field (CRF) layer on top of it<sup>70</sup>.

The model presented is a sequence-to-sequence model using bidirectional LSTM network, which fills the slots and predicts the intent in the same time. On the other hand, the model is doing the same using an attention-based RNN<sup>70</sup>. To achieve such a task, the dataset labels consist of: concatenated B-I-O (Begin, Inside, Outside) slot tags, the intent tag and an additional end-of-string (EOS) tag<sup>71</sup>.

The figure 2.5 shows the classification process for intent classification using Neural Network



**Figure 2.9:** Classification Process for Intent Classification Using Neural Network<sup>70</sup>.

### 2.2.2 Natural Language Generator (NLG)

Natural Language Generation (NLG) is the process of generating text from a meaning representation. It can be taken as the reverse of the natural language understanding.

NLG systems provide a critical role for text summarization, machine translation, and

dialog systems<sup>72</sup>. In the NLG, The system response as a semantic frame, it maps back to a natural language sentence, understandable for the end user. The NLG component can be rule-based or model-based<sup>72</sup>.

In some scenarios it can be a hybrid model, i.e. combination of both. The rule-based NLG outputs some predefined template sentences for a given semantic frame, thus they are very limited without any generalisation power. While several general-purpose rule-based generation systems have been developed, they are often quite difficult to adapt to small, task-oriented applications because of their generality.

Trainable NLG systems can produce various candidate utterances (e.g., scholastically or rule base) and use a statistical model to rank them. The statistical model assigns scores to each utterance and is learnt based on textual data. Most of these systems use bigram and trigram language models to generate utterances.

On the other hand, In NLG based on a semantically controlled Long Short-term Memory (LSTM) recurrent network, It can learn from unaligned data by jointly optimising its sentence planning and surface realisation components using a simple cross entropy training criterion without any heuristics, and good quality language variation is obtained simply by randomly sampling the network outputs<sup>73</sup>.

### **2.2.3 Dialogue Management (DM)**

The DM could be connected to some external Knowledge Base (KB) or Data Base (DB), such that it can produce more meaningful answers<sup>74</sup>. The Dialogue Manager consists the following two components: the Dialogue State Tracker (DST) and the Policy Learning which is the Reinforcement Learning (RL) agent<sup>75</sup>. The figure below shows how dialogue state Tracker and RL agent are working together.

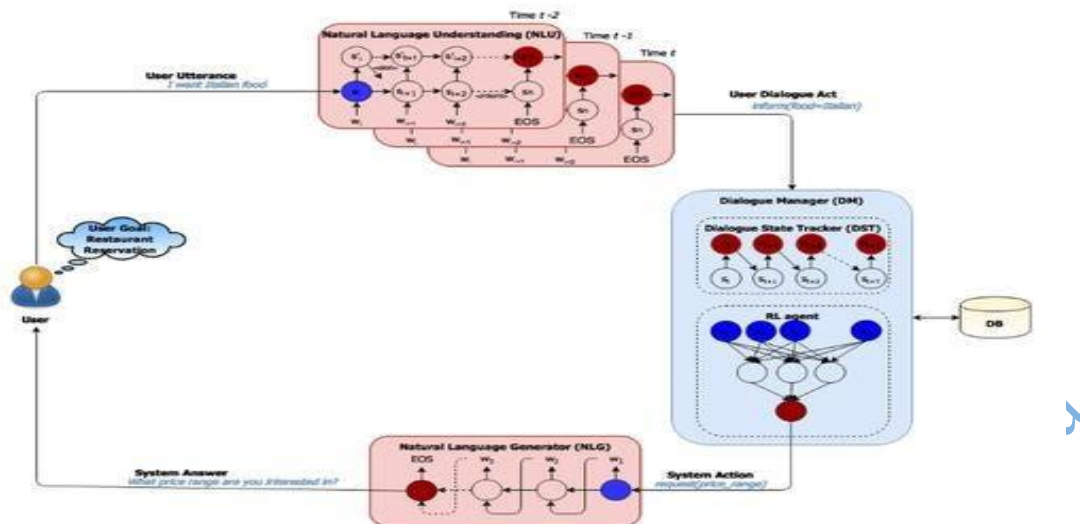
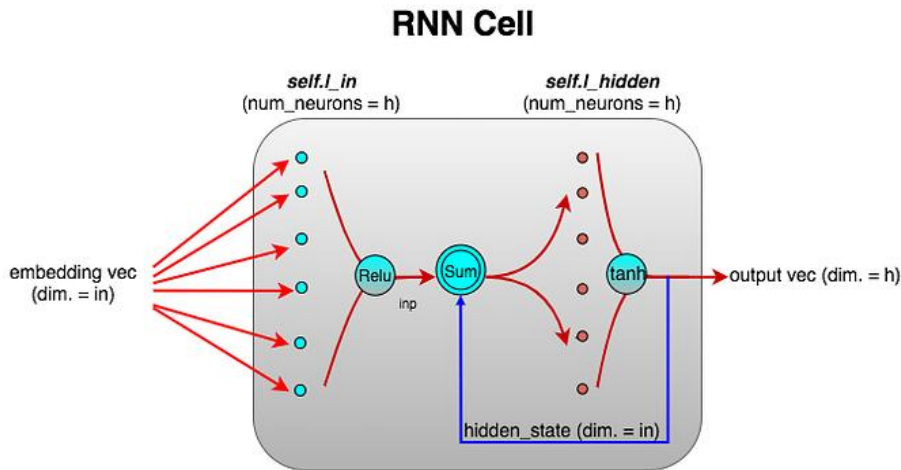


Figure 2.10: How Dialogue State Tracker and RL Agent Working Together<sup>69</sup>.

## 2.2.4 Sequence2Sequence Model with Multihead Attention Mechanism

### 2.2.4.1 Recurrent Neural Network

Recurrent Neural Networks (RNNs) are popular models that have shown great promise in many NLP tasks. The idea behind RNNs is to make use of sequential information. In a traditional neural network we assume that all inputs (and outputs) are independent of each other. But for many tasks that's a very bad idea. If you want to predict the next word in a sentence you better know which words came before it. RNNs are called *recurrent* because they perform the same task for every element of a sequence, with the output being depended on the previous computations<sup>75</sup>. Another way to think about RNNs is that they have a “memory” which captures information about what has been calculated so far. In theory RNNs can make use of information in arbitrarily long sequences, but in practice they are limited to looking back only a few steps. Here is what a typical RNN looks like:

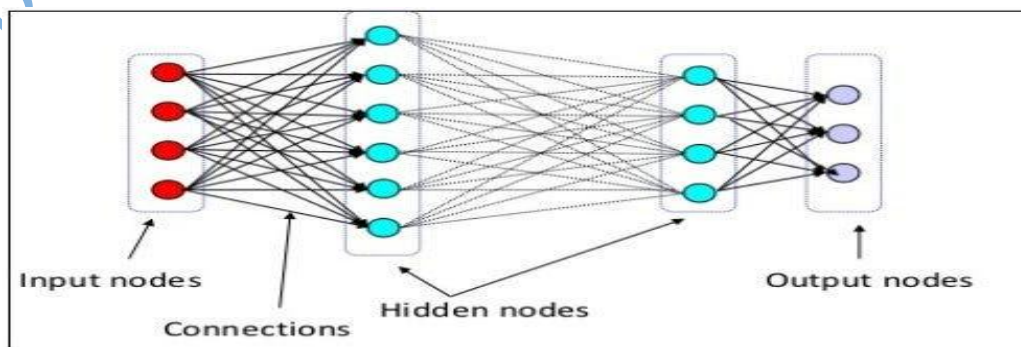


**Figure 2.11: Working of RNN for Language Modeling<sup>69</sup>.**

Recurrent Neural Networks can be used in a variety of scenarios depending in how the inputs are fed and the outputs are interpreted. These scenarios can be divided into three main different classes:

#### 2.2.4 2 Deep RNN with Multilayer Perceptron

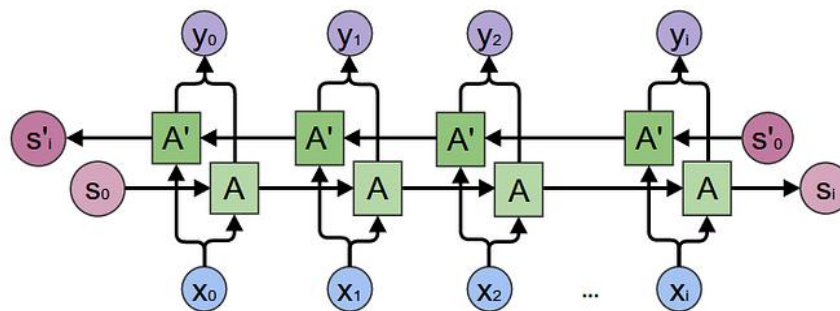
Deep architectures of neural networks can represent a function exponentially more efficient than shallow architectures<sup>76</sup>. While recurrent networks are inherently deep in time given each hidden state is a function of all previous hidden states, it has been shown that the internal computation is in fact quite shallow. It is argued that adding one or more nonlinear layers in the transition stages of a RNN can improve overall performance by better disentangling the underlying variations the original input. The deep structures in RNNs with perceptron layers can fall under three categories: input to hidden, hidden to hidden, and hidden to output<sup>77</sup>.



**Figure 2.12 : Deep RNN with Multilayer Perceptron<sup>78</sup>.**

### 2.2.4.3 Bi-Directional Recurrent Neural Network

The structure of BRNN is an to split the state neurons of a regular RNN in a part that is responsible for the positive time direction (forward states) and a part for the negative time direction (backward states)<sup>79</sup>. Outputs from forward states are not connected to inputs of backward states, and vice versa. The figure 2.11 illustrate the General structure of the bidirectional recurrent neural network,



**Figure 2.13: Bi-Directional Recurrent Neural Network<sup>80</sup>.**

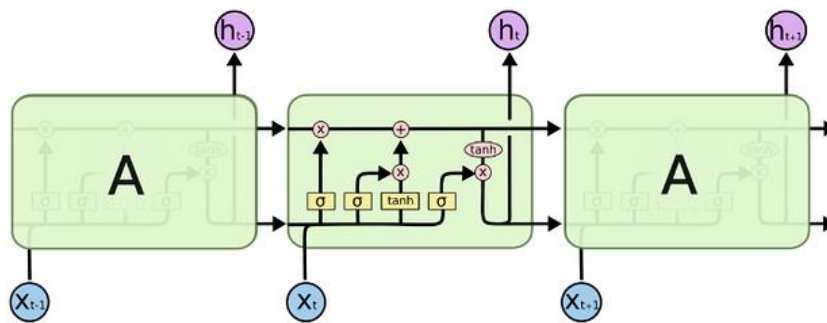
### 2.2.4.4 Multidimensional Recurrent Neural Network

The basic idea of multidimensional recurrent neural networks (MDRNNs) is to replace the single recurrent connection found in standard recurrent networks with as many connections as there are spatio-temporal dimensions in the data<sup>81</sup>. These connections allow the network to create a flexible internal representation of surrounding context, which is robust to localised distortions. An MDRNN hidden layer scans through the input in 1D strips, storing its activations in a buffer. The strips are ordered in such a way that at every point the layer has already visited the points one step back along every dimension. The hidden activations at these previous points are fed to the current point through recurrent connections, along with the input.

### 2.2.4.5 Long Short-Term Memory or LSTM Network

An LSTM network is a recurrent neural network that has LSTM cell blocks in place of our standard neural network layers. These cells have various components

called the input gate, the forget gate and the output gate<sup>82</sup>. RNNs are good in handling sequential data but they run into problem when the context is far away.



**Figure 2.14: Long Short-Term Memory or LSTM Network**

LSTMs are explicitly designed to avoid the long-term dependency problem. LSTMs also provide solution to Vanishing/Exploding Gradient problem. All recurrent neural networks have the form of a chain of repeating modules of neural network. In standard RNNs, this repeating module will have a very simple structure, such as a single tanh layer.

### 2.3 Review of Related Works

The review of the literature reveals that several authors have contributed significantly to the development of chatbot for mental health.

In a study to evaluate the effectiveness of a chatbot-based intervention for reducing symptoms of depression. This randomized controlled trial recruited 70 individuals with mild to moderate depression and randomly assigned them to either a chatbot-based intervention group or a face-to-face cognitive behavioral therapy (CBT) group. The chatbot provided CBT to individuals through a conversational interface, while the face-to-face CBT was provided by trained therapists. The study assessed depression symptoms using the Patient Health Questionnaire-9 (PHQ-9) at baseline, post-treatment, and 3-month follow-up. The study found that both the chatbot-based intervention and face-to-face CBT were effective in reducing symptoms of depression. The chatbot-based intervention group had a mean PHQ-9 score reduction of 6.2 points

at post-treatment and 6.5 points at 3-month follow-up, while the face-to-face CBT group had a mean reduction of 6.0 points at post-treatment and 6.6 points at 3-month follow-up. There was no significant difference between the two groups in terms of symptom reduction<sup>83</sup>.

In a study that summarize recent developments in the use of chatbots for mental health support, discuss their potential benefits and limitations, and highlight areas for future research. This review article conducted a comprehensive search of the literature on the use of chatbots for mental health support, focusing on studies published between 2016 and 2020. The authors reviewed and summarized recent studies that have evaluated the effectiveness of chatbots in improving mental health outcomes, and discussed the challenges associated with their implementation. The review found that chatbots have the potential to provide accessible and personalized mental health support to individuals, particularly those who face barriers to accessing traditional mental health services. However, the authors also highlighted several challenges associated with their implementation, including concerns about privacy and data security, as well as the need for further research to evaluate their effectiveness in different contexts<sup>84</sup>.

Another similar study reviewed recent developments in the use of chatbots for mental health support and evaluate their potential benefits and limitations. This review article analyzed recent studies that have evaluated the effectiveness of chatbots in improving mental health outcomes. The authors discussed the challenges associated with implementing chatbots in mental health, including issues related to privacy, ethics, and user engagement. They also highlighted areas for future research, such as the need to evaluate the long-term effects of chatbot-based interventions. The authors found that chatbots have the potential to provide accessible and cost-effective mental

health support to individuals. However, they also noted that there are several challenges that need to be addressed in order to fully realize the potential of chatbots in mental health. These challenges include issues related to data privacy, user engagement, and the need for more rigorous evaluation of chatbot-based interventions<sup>85</sup>.

In a related study to develop and evaluate the effectiveness of a machine learning-based chatbot for mental health support. This study developed a chatbot using machine learning algorithms to provide mental health support to individuals with depression and anxiety. The chatbot was trained on a dataset of mental health conversations and used natural language processing (NLP) techniques to understand and respond to user input. The study evaluated the effectiveness of the chatbot in reducing symptoms of depression and anxiety using pre- and post-intervention assessments. The study found that the machine learning-based chatbot was effective in reducing symptoms of depression and anxiety. Participants who used the chatbot showed significant improvements in depression and anxiety symptoms compared to a control group. The chatbot was also found to be highly acceptable to users, with high ratings for usability and satisfaction<sup>86</sup>.

In a work that designed an empathetic chatbot for mental health counseling using machine learning. This study used a machine learning-based approach to design an empathetic chatbot for mental health counseling. The authors collected a dataset of counseling transcripts and used natural language processing techniques to extract features related to empathy, such as emotional expressions and rapport-building strategies. They then used these features to train a machine learning model to generate empathetic responses. The authors found that their empathetic chatbot was effective in improving users' mood and reducing symptoms of depression and anxiety. Users

reported high levels of satisfaction with the chatbot and found it to be a helpful tool for managing their mental health. The authors suggest that their approach could be used to develop more effective and personalized mental health interventions<sup>87</sup>.

In a study to design a machine learning-based chatbot for mental health support and evaluate its usability and acceptability. This study developed a chatbot using machine learning algorithms and natural language processing techniques. The chatbot was designed to provide mental health support to individuals with depression and anxiety. The study evaluated the usability and acceptability of the chatbot using a sample of 30 individuals who interacted with the chatbot for a period of four weeks. The study assessed the usability of the chatbot using the System Usability Scale (SUS) and evaluated its acceptability using a qualitative analysis of user feedback. The study found that the machine learning-based chatbot was highly usable and acceptable to users. The chatbot received a SUS score of 80.3 out of 100, indicating high usability. Users also provided positive feedback about the chatbot's ability to provide personalized mental health support and its ease of use<sup>88</sup>.

Another study designed a chatbot for mental health support using machine learning and evaluate the feasibility and acceptability of the chatbot among users. This study used a user-centered design approach to develop a chatbot for mental health support. The chatbot was designed to provide cognitive behavioral therapy (CBT) to individuals with depression and anxiety. The study recruited 20 individuals with depression and anxiety to test the chatbot and evaluated its feasibility and acceptability using a combination of qualitative and quantitative methods. The study found that the chatbot was feasible and acceptable to users. The chatbot was able to provide personalized CBT to users and had high engagement rates. Users reported that the chatbot was easy to use and provided valuable support for their mental health.

However, the authors noted that further research is needed to evaluate the effectiveness of the chatbot in improving mental health outcomes<sup>89</sup>.

In a similar study to design a machine learning-based chatbot for mental health support and evaluate its usability and effectiveness. This study developed a machine learning-based chatbot for mental health support that used natural language processing (NLP) and sentiment analysis to provide personalized support to users. The chatbot was trained on a dataset of mental health conversations and was designed to provide empathetic responses to users. The study evaluated the usability and effectiveness of the chatbot through a user study with 20 participants, who interacted with the chatbot for a period of two weeks. The study found that the machine learning-based chatbot was effective in providing personalized mental health support to users. The chatbot was able to accurately detect the emotional state of users and provide appropriate responses. The study also found that users were satisfied with the chatbot and found it to be a useful tool for managing their mental health<sup>90</sup>.

Another study designed an artificial intelligence-based chatbot for mental health support that is capable of providing personalized and empathetic responses to users. This study used a human-centered design approach to develop an artificial intelligence-based chatbot for mental health support. The authors conducted interviews with mental health professionals and individuals with lived experience of mental illness to identify key features of effective mental health chatbots. They used these features to inform the design of their chatbot, which was developed using natural language processing and machine learning techniques. The authors developed an artificial intelligence-based chatbot that is capable of providing personalized and empathetic responses to users. The chatbot was designed to provide psychoeducation, emotional support, and crisis intervention to individuals with mental health concerns.

The authors conducted a pilot study to evaluate the effectiveness of the chatbot in improving mental health outcomes, and found that it was effective in reducing symptoms of depression and anxiety<sup>91</sup>.

In a related study that introduced SERMO, a mobile application with integrated chatbot that implements methods from cognitive behaviour therapy (CBT) to support mentally ill people in regulating emotions and dealing with thoughts and feelings. SERMO asks the user on a daily basis on events that occurred and on emotions. It determines automatically the basic emotion of a user from the natural language input using natural language processing and a lexicon-based approach. Depending on the emotion, an appropriate measurement such as activities or mindfulness exercises are suggested by SERMO. Additional functionalities are an emotion diary, a list of pleasant activities, mindfulness exercises and information on emotions and CBT in general. User experience was studied with 21 participants using the User Experience Questionnaire (UEQ). Their findings show that efficiency, perspicuity and attractiveness are considered as good. The scales describing hedonic quality (stimulation and novelty), i.e., fun of use, show neutral evaluations<sup>92</sup>.

In a work that designed, implemented and evaluated a chatbot that offered three chatting styles; they conducted a study with 47 participants who were randomly assigned into three groups where each group experienced the chatbot's self-disclosure at varying levels respectively. After using the chatbot for a few weeks, participants were introduced to a MHP and were asked if they would like to share their self-disclosed content with the MHP. Their results showed that, within each group, the depth of participants' self-disclosure to the chatbot remained after sharing with the MHP; participants exhibited deeper self-disclosure to the MHP through a more self-disclosing chatbot; further, through conversation log analysis<sup>93</sup>.

In a work that aimed to design a conversational sequence for a brief motivational interview to be delivered by a Web-based text messaging application (*chatbot*) and to investigate its conversational experience with graduate students in their coping with stress. The authors use a brief conversational sequence which was designed with varied combinations of MI skills to follow the 4 processes of MI. A Web-based text messaging application, Bonobot, was built as a research prototype to deliver the sequence in a conversation. A total of 30 full-time graduate students who self-reported stress with regard to their school life were recruited for a survey of demographic information and perceived stress and a semistructured interview. Participants had a high level of perceived stress (mean 22.5 [SD 5.0])<sup>94</sup>.

In another study that developed an agent-based framework for chatbot development named EREBOTS which are based on the implementation of (i) multi-front-end connectors and interfaces (i.e., Telegram, dedicated App, and web interface), (ii) enabling the configuration of multi-scenario behaviors (i.e., preventive physical conditioning, smoking cessation, and support for breast-cancer survivors), (iii) online learning, (iv) personalized conversations and recommendations (i.e., mood boost, anti-craving persuasion, and balance-preserving physical exercises), and (v) responsive multi-device monitoring interface (i.e., doctor and admin). Their result indicated that EREBOTS has been tested in the context of physical balance preservation in social confinement times (due to the ongoing pandemic)<sup>95</sup>.

In a similar study that designed a fully automated chatbot (“Viki”) to evaluate employee risks of suffering from depression, anxiety, stress, insomnia, burnout, and work-related stress. The authors conducted a cross-sectional analysis to gain first insights of a pilot implementation within a small to medium-sized enterprise (120

employees). Their result showed that the response rate was 64.2% (77/120). In total, 98 employees started the assessment, 77 of whom (79%) completed it. The majority of participants scored in the mild range for anxiety (20/40, 50%) and depression (16/28, 57%), in the moderate range for stress (10/22, 46%), and at the subthreshold level for insomnia (14/20, 70%) as defined by their questionnaire scores<sup>96</sup>.

In a work that developed a chatbot aimed at screening, monitoring, and reducing anxiety-depressive symptoms and work-related burnout, and detecting suicide risk in patients from PC and health care workers. They simulated Vickybot for 2 weeks and used HCs to simulate different clinical situations. For feasibility and effectiveness study, people consulting PC or health care workers with mental health problems used Vickybot for 1 month. Self-assessments for anxiety (Generalized Anxiety Disorder 7-item) and depression (Patient Health Questionnaire-9) symptoms and work-related burnout (based on the Maslach Burnout Inventory) were administered at baseline and every 2 weeks. From their results, 40 HCs tested Vickybot simultaneously, and the data were reliably transmitted and registered. For simulation, 17 HCs (n=13, 76% female; mean age 36.5, SD 9.7 years) received 98.8% of the expected modules. Suicidal alerts were received correctly. Vickybot showed high subjective UEI (acceptability, usability, and satisfaction), but low objective UEI (completion, adherence, compliance, and engagement)<sup>97</sup>.

Another study introduced Emohaa, a conversational agent that provides cognitive support through CBT-Bot exercises and guided conversations. It also emotionally supports users through ES-Bot, enabling them to vent their emotional problems. The current study randomly assigned participants into three groups: Emohaa (CBT-Bot), Emohaa (Full), and control. With both Intention-To-Treat (N=247) and PerProtocol

(N=134) analyses, the results demonstrated that compared to the control group, participants who used two types of Emohaa experienced considerably more significant improvements in symptoms of mental distress, including depression ( $F[2,244]=6.26, p=0.002$ ), negative affect ( $F[2,244]=6.09, p=0.003$ ), and insomnia ( $F[2,244]=3.69, p=0.026$ )<sup>98</sup>.

In a work that presented a framework and functionality of a chatbot developed using web technologies. The bot engine was integrated by several machine learning approaches like gradient descent (GD) and natural language processing (NLP) algorithms. The trained data entered into the bot were split into mini-word batches, and the GD algorithm was applied sequentially on each mini-batch. The NLP methods involved in converting a word to its stem with a text result less readable by humans. The employed ML algorithms were successfully incorporated to manage the alternative synchronisation of text and voice messages<sup>99</sup>.

In a study that seeks to evaluate the effectiveness of the Mental Health Intelligent Information Resource Assistant in supporting health care workers and their families in the Canadian provinces of Alberta and Nova Scotia with the provision of appropriate information on mental health issues, services, and programs based on personalized needs. The effectiveness of the technology will be assessed via voluntary follow-up surveys and an analysis of client interactions and engagement with the chatbot. Client satisfaction with the chatbot will also be assessed<sup>100</sup>.

The aim of this research is to raise awareness about mental health while simultaneously working towards removing the societal stigma surrounding it. Thus, in this paper, the authors have created an integrated chatbot that is specifically geared towards mentally ill individuals. The chatbot responds empathetically which is built

using a Sequence-to-Sequence (Seq2Seq) encoder-decoder architecture. The encoder uses Bi-directional Long Short Term Memory (BiLSTM). To compare the performance, the authors used Beam Search and Greedy Search. They found Beam Search decoder performs much better, providing empathetic responses to the user with greater precision in terms of BLEU score<sup>101</sup>.

The aim of this paper is the development of a chatbot against the disturbing psychic consequences of the pandemic, taking human emotion recognition into account. The object is to help people; especially students; suffering from mental disorders, by progressively understanding the reasons behind them. This innovative chatbot was developed by using the natural language processing model of deep learning. An advanced model of deep learning has been elaborated the intention for people, and that to help them to regulate their mood and to reduce distortion of negative thoughts, that why a collection of a new database was done. The sequence-to-sequence model encoder and decoder consist of Long short-term memory cells, and it is defined with the bi-directional dynamic recurrent neural network packets<sup>102</sup>.

In this paper, the authors proposed Evebot, an innovative, sequence to sequence (Seq2seq) based, fully generative conversational system for the diagnosis of negative emotions and prevention of depression through positively suggestive responses. The system consists of an assembly of deep-learning based models, including Bi-LSTM based model for detecting negative emotions of users and obtaining psychological counselling related corpus for training the chatbot, anti-language sequence to sequence neural network, and maximum mutual information (MMI) model. As adolescents are reluctant to show their negative emotions in physical interaction, traditional methods of emotion analysis and comforting methods may not work.

Therefore, this system puts emphasis on using virtual platform to detect signs of depression or anxiety, channel adolescents' stress and mood, and thus prevent the emergence of mental illness. They launched the integrated chatbot system onto an online platform for real-world campus applications. Through a one-month user study, the authors observed better results in the increase in positivity than other public chatbots in the control group<sup>103</sup>.

This paper presents the structure and implementation procedures for a mental health support system combining technology and professionals. The system is a web platform where mental health seekers can register and use functionalities like NLP-based chatbot for personality assessment, chatting with like-minded people, and one-to-one video conferencing with a mental health professional. The video calling feature of the system has emotion detection capabilities using computer vision. The system also includes downloadable prescription facilities and a payment gateway for secure transactions. From a technological aspect, the conversational NLP-based chatbot and computer vision-powered video calling are the system's most important features. The system has a documentation facility to analyze the mental health condition over time. The web platform is built using React.js for the frontend and Express.js for the backend. MongoDB is used as the database of the platform. The NLP chatbot is built on a three-layered deep neural network model that is programmed in the Python language and uses the NLTK, TensorFlow, and Keras sequential API. Video conference is one of the most important features of the platform. To create the video calling feature, Express.js, Socket.io, and Socket.io-client have been used. The emotion detection feature is implemented on video conferences using computer vision, Haar Cascade, and TensorFlow. All the implemented features are tested and work fine<sup>104</sup>.

In this paper, the authors proposed Evebot, an innovative, sequence to sequence (Seq2seq) based, fully generative conversational system for the diagnosis of negative emotions and prevention of depression through positively suggestive responses. The system consists of an assembly of deep-learning based models, including Bi-LSTM based model for detecting negative emotions of users and obtaining psychological counselling related corpus for training the chatbot, anti-language sequence to sequence neural network, and maximum mutual information (MMI) model. As adolescents are reluctant to show their negative emotions in physical interaction, traditional methods of emotion analysis and comforting methods may not work. Therefore, this system puts emphasis on using virtual platform to detect signs of depression or anxiety, channel adolescents' stress and mood, and thus prevent the emergence of mental illness. They launched the integrated chatbot system onto an online platform for real-world campus applications. Through a one-month user study, we observe better results in the increase in positivity than other public chatbots in the control group<sup>105</sup>.

In this paper, the authors proposed an AI web-based chatbot called “Ted” to assist people with mental health-related queries with the help of natural language processing and deep learning approaches. The user message is lemmatized and pre-processed in this step before being passed to the deep-learning model. Then, to specify the question category, an Artificial Neural Network with Softmax is used. This chatbot will allow the users to interact, use natural language to take input, and generate the appropriate response according to the input. The accuracy of our proposed chatbot is 98.13% in providing the appropriate response. In addition to this, “Ted” will help the patients who are reluctant to speak and get stigmatized by the presence of men<sup>106</sup>.

In a work to overcome the limitations of Mental health chatbots in the extant literature and practice are limited in terms of the therapy provided and the level of personalisation. For instance, most chatbots extend Cognitive Behavioural Therapy (CBT) into predefined conversational pathways that are generic and ineffective in recurrent use. The authors postulated that Behavioural Activation (BA) therapy and Artificial Intelligence (AI) are more effectively materialised in a chatbot setting to provide recurrent emotional support, personalised assistance, and remote mental health monitoring. We present the design and development of our BA-based AI chatbot, followed by its participatory evaluation in a pilot study setting that confirmed its effectiveness in providing support for individuals with mental health issues<sup>107</sup>.

The scope of the proposed work covers multidisciplinary areas like psychiatric analysis, mental health, communication system, machine learning (ML), natural language processing (NLP), deep learning (DL), and sentiment analysis (SA). The proposed chatbot has a well-designed architecture with a structured knowledge engine to provide an intuitive interface. The design consists of three components with a good balance of cohesion and articulation through medical knowledge and conversational dialog. It is backed by a bank of structured information resources connected to COrona Virus Disease 19 (COVID-19). The chatbot also offers mental health counseling and self-assessment exams for mental illnesses, in addition to corona virus information. The novelty of the proposed work is to provide an interface to interact with the virtual avatar in a rich environment supported by voice and language capabilities. It advises the user to consult a medical practitioner to determine whether they are affected with COVID-19 and provide them with recommendations. Along with providing a psychiatric assessment to the end-user and conversation with

interactive virtual avatars, the proposed chatbot can understand texts in the English language, with a training accuracy of 66% of the proposed deep learning model<sup>108</sup>.

## 2.4 Summary of Gaps in Literature Reviewed

This chapter was organised into four sub-headings - conceptual review, theoretical review/framework, review of empirical studies related to the research topic and conceptual model. The conceptual review explained in depth the concepts of the study. These concepts are - mental health, artificial intelligence, AI in mental health, machine learning, sequence models, Natural Language Processing, chatbots. It also richly gave insights into sub-concepts such as different types of sequence models, chatbot which includes ELIZA, Elizabeth and ALICE.

In the methodological review, the algorithm used in the design of the chatbot for this study were fully explained. They are common deep learning techniques such as word embedding, Recurrent Neural Networks.

In the review of empirical studies, different authors contribution to the development of chatbot for mental health were presented. The studies show that many empirical research works similar to the topic under study have been carried out. However, past empirical studies using other algorithms showed lower accuracy. From previous literature and comparative analyses on chatbot development for mental health, the authors used various techniques such as mobile application with integrated chatbot, natural language understanding (NLU), and emotion recognition, Web-based text messaging application (*chatbot*), agent-based framework for chatbot, fully automated chatbot, cognitive support through CBT-Bot, machine learning and natural language processing (NLP) algorithms in developing a mental health chatbot<sup>94,95,96,97,98</sup>. Also another author used GD algorithm and natural language processing (NLP) algorithms in the development of their chatbot<sup>99</sup>.

From the foregoing, it can be observed that the previous works presented a bot system with low-complexity models and intelligence. Therefore, a more interactive chatbot using sequential models based on frequently asked questions on mental health (chatbot) must be developed to provide people seeking mental health support with a user-friendly system that offers individualized recommendations and assistance based on their specific needs. The chatbot developed in this study will be designed to comprehend and respond appropriately to user inquiries using sequential models, natural language processing techniques and machine learning algorithms.

*Do Not Copy, Lead City University, Nigeria*

## Endnotes

1. M Iasiello, J van Agteren, M Schotanus-Dijkstra, L Lo, DB Fassnacht, Westerhof GJ. *Assessing mental wellbeing using the Mental Health Continuum—Short Form: A systematic review and meta-analytic structural equation modelling*. **Clinical Psychology: Science and Practice**. 2022 Mar 21.
2. World Health Organization. *World mental health report: transforming mental health for all: executive summary*. In *World mental health report: transforming mental health for all: executive summary 2022*.
3. M Freeman. *The World Mental Health Report: transforming mental health for all*. **World Psychiatry**. 2022 Oct;21(3):391.
4. T Lomas, TJ VanderWeele. *The Mental Illness-Health Matrix and the Mental State Space Matrix: Complementary meta-conceptual frameworks for evaluating psychological states*. **Journal of Clinical Psychology**. 2023 Mar 14.
5. S Preethi. *A Survey on Artificial Intelligence*. **International Journal of Intelligent Computing and Technology (IJICT)**, 3(2), pp.39-42.2020
6. E Burns, N Laskowski, L Tucci. *What is artificial intelligence*. Search Enterprise AI. 2021 Nov.
7. S Aleem, N.U Huda, R Amin, S Khalid, S.S Alshamrani, A Alshehri. *Machine learning algorithms for depression: diagnosis, insights, and research directions*. *Electronics*. 2022 Mar 31;11(7):1111.
8. S Ayanouz, B.A Abdelhakim, M Benhmed. *A smart chatbot architecture based NLP and machine learning for health care assistance*. In *Proceedings of the 3rd international conference on networking, information systems & security 2020* Mar 31 (pp. 1-6).
9. S.K Chinnamgari. *Machine Learning Projects: Implement supervised, unsupervised, and reinforcement learning techniques using R 3.5*. Packt Publishing Ltd; 2019 Jan 14.
10. H Hihn, D.A Braun. *Specialization in Hierarchical Learning Systems: A Unified Information-theoretic Approach for Supervised, Unsupervised and Reinforcement Learning*. *Neural Processing Letters*. 2020 Dec;52(3):2319-52.
11. J.A Andeta. *Road-traffic accident prediction model : Predicting the Number of Casualties [Internet] [Dissertation]*. 2021. Available from: <http://urn.kb.se/resolve?urn=urn:nbn:se:his:diva-20146>
12. J.A Sidey-Gibbons, C.J Sidey-Gibbons. *Machine learning in medicine: a practical introduction*. *BMC medical research methodology*. 2019 Dec;19:1-8.

13. I.H Sarker. Machine learning: Algorithms, real-world applications and research directions. *SN computer science*. 2021 May;2(3):160.
14. Y Bengio, A Lodi, A Prouvost. *Machine learning for combinatorial optimization: a methodological tour d'horizon*. **European Journal of Operational Research**. 2021 Apr 16;290(2):405-21.
15. V Babenko, A Panchyshyn, L Zomchak, M Nehrey, Z Artym-Drohomyretska, T Lahotskyi. *Classical machine learning methods in economics research: Macro and micro level example*. *WSEAS Transactions on Business and Economics*. 2021 Jan;18:209-17.
16. K.K Hiran, R.K Jain, K Lakhwani, R Doshi. *Machine Learning: Master Supervised and Unsupervised Learning Algorithms with Real Examples (English Edition)*. **BPB Publications**; 2021 Sep 16.
17. T Bokaba, W Doorsamy, B.S Paul. *Comparative study of machine learning classifiers for modelling road traffic accidents*. **Applied Sciences**. 2022 Jan 14;12(2):828.
18. A Mollajan, H Iranmanesh, A Khezri, A Abazari. *Effect of applying independence axiom of Axiomatic Design theory on performance of an Integrated Manufacturing Information System: a computer simulation modeling approach*. *Simulation*. 2022 Jul;98(7):535-61.
19. R Gupta, S Ashish, H Shekhar, M.D Dominic. *Music genre classification using CNN and RNN-LSTM*. In *International Conference on Micro-Electronics and Telecommunication Engineering 2021 Sep 24* (pp. 729-745). Singapore: Springer Nature Singapore.
20. R Nallapati, F Zhai, B Zhou. *Summarunner: A recurrent neural network based sequence model for extractive summarization of documents*. In *Proceedings of the AAAI conference on artificial intelligence 2017 Feb 12* (Vol. 31, No. 1).
21. B Borah, D Pathak, P Sarmah, B Som, S Nandi. *Survey of textbased chatbot in perspective of recent technologies*. In *Computational Intelligence, Communications, and Business Analytics: Second International Conference, CICBA 2018, Kalyani, India, July 27–28, 2018, Revised Selected Papers, Part II 2 2019* (pp. 84-96). Springer Singapore.
22. Z Zhao, A Srivastava, L Peng, Q Chen. *Long short-term memory network design for analog computing*. **ACM Journal on Emerging Technologies in Computing Systems (JETC)**. 2019 Jan 9;15(1):1-27.
23. A Truong, A Walters, J Goodsitt, K Hines, C.B Bruss, R Farivar. *Towards automated machine learning: Evaluation and comparison of AutoML approaches and tools*. In *2019 IEEE 31st international conference on tools with artificial intelligence (ICTAI) 2019 Nov 4* (pp. 1471-1479). IEEE.

24. J Barber, X Yu, L.K Zamore, J Lin, V Jazayeri, S Erlich, T Savor, M Stumm. *Bladerunner: Stream processing at scale for a live view of backend data mutations at the edge*. In Proceedings of the ACM SIGOPS 28th Symposium on Operating Systems Principles 2021 Oct 26 (pp. 708-723).
25. A.J Kockx. *Development and Evaluation of a Diagnosis and Triage Healthcare Chatbot* (Master's thesis).2021
26. S Schanke, G Burtch, G Ray. *Estimating the impact of “humanizing” customer service chatbots*. Information Systems Research. 2021 Sep;32(3):736-51.
27. L Wang, D Wang, F Tian, Z Peng, X Fan, Z Zhang, Yu M, Ma X, Wang H. *Cass: Towards building a social-support chatbot for online health community*. Proceedings of the ACM on Human-Computer Interaction. 2021 Apr 22;5(CSCW1):1-31.
28. A Przegalinska, L Ciechanowski, A Stroz, P Gloor, G Mazurek. *In bot we trust: A new methodology of chatbot performance measures*. Business Horizons. 2019 Nov 1;62(6):785-97.
29. S Natale. *If software is narrative: Joseph Weizenbaum, artificial intelligence and the biographies of ELIZA*. new media & society. 2019 Mar;21(3):712-28.
30. A.A Weißensteiner. *Chatbots as an approach for a faster enquiry handling process in the service industry*. Signature. 2018;12(04)
31. H He, J Gray, A Cangelosi, Q Meng, T.M McGinnity, J Mehnen. *The challenges and opportunities of human-centered AI for trustworthy robots and autonomous systems*. IEEE Transactions on Cognitive and Developmental Systems. 2021 Dec 2;14(4):1398-412.
32. [https://en.wikipedia.org/wiki/ELIZA#/media/File:ELIZA\\_conversation.png](https://en.wikipedia.org/wiki/ELIZA#/media/File:ELIZA_conversation.png)
33. S Singh, H.K Thakur. *Survey of various AI chatbots based on technology used*. In 2020 8th International Conference on Reliability, Infocom Technologies and Optimization (Trends and Future Directions)(ICRITO) 2020 Jun 4 (pp. 1074-1079). IEEE
34. S Fernandes, R Gawas, P Alvares, M Femandes, D Kale, S Aswale. *Survey on various conversational systems*. In 2020 International Conference on Emerging Trends in Information Technology and Engineering (ic-ETITE) 2020 Feb 24 (pp. 1-8). IEEE.
35. A Toprak, M Turan, F.S Toprak. *Deep Learning Based Chatbot Adapted to the Electronic Funds Transfer Process of Turkish Banking*. **International Journal of Intelligent Systems and Applications in Engineering**. 2023 Feb 4;11(1):348-53.

36. T Nemoto, T Fujimoto. *A Classification and Analysis Focusing on Attempts to Give a Computer a Personality: A Technological History of Chatbots as Simple Artificial Intelligence*. In *International Conference on Applied Informatics and Media Design 2019 Apr 24* (pp. 59-70). Cham: Springer International Publishing.
37. <https://onlim.com/en/the-history-of-chatbots/>
38. G.K Ahirwar. *Chatterbot: technologies, tools and applications*. In *High Performance Vision Intelligence 2020* (pp. 203-213). Springer, Singapore.
39. N.A Al-Madi, K.A Maria, M.A Al-Madi, M.A Alia. and Maria, E.A., 2021, July. *An intelligent Arabic chatbot system proposed framework*. In *2021 International Conference on Information Technology (ICIT)* (pp. 592-597). IEEE
40. H Singh, A Bhangare, R Singh, S Zope, P Saindane. *Chatbots: A Survey of the Technology*. In *International Conference on Intelligent Cyber Physical Systems and Internet of Things 2022 Aug 11* (pp. 671-691). Cham: Springer International Publishing.
41. S.J Shaikh. *Artificially Intelligent, Interactive, and Assistive Machines: A Definitional Framework for Intelligent Assistants*. **International Journal of Human-Computer Interaction**. 2023 Feb 25;39(4):776-89.
42. A Vadhera, A Thute, S Mala, A Shankar. *Chatbot on COVID-19 for sustaining good health during the pandemic*. In *Latest Trends in Renewable Energy Technologies: Select Proceedings of NCRESE 2020 2021* (pp. 271-284). Springer Singapore.
43. N George, K Muiz, P Whig, A Velu. *Framework of Perceptive Artificial Intelligence using Natural Language Processing (PAIN)*. *Artificial & Computational Intelligence/Published Online*: July. 2021 Jul.
44. W Wang, W Wang. *Talking to Cortana. Absolute Beginners Guide to Computing*. 2016:257-82.
45. H Chung, J Park, S Lee. *Digital forensic approaches for Amazon Alexa ecosystem*. *Digital investigation*. 2017 Aug 1;22:S15-25.
46. L De Angelis, F Baglivo, G Arzilli, G.P Privitera, P Ferragina, A.E Tozzi, C Rizzo. *ChatGPT and the rise of large language models: the new AI-driven infodemic threat in public health*. *Frontiers in Public Health*. 2023 Apr 25;11:1166120.
47. P.S Mattas. *ChatGPT: A Study of AI Language Processing and its Implications*. *Journal homepage: www. ijpr. com ISSN*. 2023;2582:7421.

48. P Kostelník, I Pisařovic, M Muroň, F Dařena, D Procházka. *Chatbots for enterprises: outlook*. Acta Universitatis Agriculturae et Silviculturae Mendelianae Brunensis. 2019.
49. A Alnefaie, S Singh, B Kocaballi, M Prasad. *An overview of conversational agent: applications, challenges and future directions*. In 17th International Conference on Web Information Systems and Technologies 2021 Jan 1. SCITEPRESS-Science and Technology Publications.
50. C Siebra, W Correia, M Penha, J Macedo, J Quintino, M Anjos, F Florentin, F.Q Da Silva, A.L Santos. *Virtual assistants for mobile interaction: A review from the accessibility perspective*. In Proceedings of the 30th Australian Conference on Computer-Human Interaction 2018 Dec 4 (pp. 568-571).
51. S Ahire, M Rohs. *Tired of wake words? Moving towards seamless conversations with intelligent personal assistants*. In Proceedings of the 2nd Conference on Conversational User Interfaces 2020 Jul 22 (pp. 1-3).
52. J Pereira, O Díaz. *A quality analysis of facebook messenger's most popular chatbots*. In Proceedings of the 33rd annual ACM symposium on applied computing 2018 Apr 9 (pp. 2144-2150).
53. T Kamita, T Ito, A Matsumoto, T Munakata, T Inoue. *A chatbot system for mental healthcare based on SAT counseling method*. Mobile Information Systems. 2019 Mar 3;2019.
54. A.A Abd-Alrazaq, M Alajlani, A.A Alalwan, B.M Bewick, P Gardner, M Househ. *An overview of the features of chatbots in mental health: A scoping review*. **International Journal of Medical Informatics**. 2019 Dec 1;132:103978.
55. B Inkster, S Sarda, V Subramanian. *An empathy-driven, conversational artificial intelligence agent (Wysa) for digital mental well-being: real-world data evaluation mixed-methods study*. **JMIR mHealth and uHealth**. 2018 Nov 23;6(11):e12106.
56. K Denecke, S Vaaheesan, A Arulnathan. *A mental health chatbot for regulating emotions (SERMO)-concept and usability test*. IEEE Transactions on Emerging Topics in Computing. 2020 Feb 17;9(3):1170-82.
57. Y Zhang, L Xu, A Mendoza, G Yang, P Chinprutthiwong, G Gu. *Life after speech recognition: Fuzzing semantic misinterpretation for voice assistant applications*. In Proc. of the Network and Distributed System Security Symposium (NDSS'19) 2019 Jan.
58. X Huang, A CIS. *Chatbot: Design, architecture, and applications*. ASCS CIS 498 Senior Capstone Thesis University of Pennsylvania, School of Engineering and Applied Science May 3, 2021

59. E Adamopoulou, L Moussiades. *An overview of chatbot technology*. In IFIP International Conference on Artificial Intelligence Applications and Innovations 2020 Jun 5 (pp. 373-383). Springer, Cham
60. Y Zhang, L Xu, A Mendoza, G Yang, P Chinprutthiwong, G Gu. *Life after speech recognition: Fuzzing semantic misinterpretation for voice assistant applications*. In Proc. of the Network and Distributed System Security Symposium (NDSS'19) 2019 Jan.
61. A Pinto, O,H Gonçalo, A.A Oliveira. *Comparing the performance of different NLP toolkits in formal and social media text*. In 5th Symposium on Languages, Applications and Technologies (SLATE'16) 2016. Schloss Dagstuhl-Leibniz-Zentrum fuer Informatik.
62. T Kudo, J Richardson. *Sentencepiece: A simple and language independent subword tokenizer and detokenizer for neural text processing*. arXiv preprint arXiv:1808.06226. 2018 Aug 19.
63. R Sennrich, B Haddow, A Birch. *Neural machine translation of rare words with subword units*. arXiv preprint arXiv:1508.07909. 2015 Aug 31.
64. Z Callejas, D Griol D. *Conversational agents for mental health and wellbeing. Dialog systems: a perspective from language, logic and computation*. 2021:219-44.
65. A Khanna, B Pandey, K Vashishta, K Kalia, B Pradeepkumar, T Das. *A study of today's AI through chatbots and rediscovery of machine intelligence*. **International Journal of u-and e-Service, Science and Technology**. 2015 Jul 31;8(7):277-84.
66. S.A Abdul-Kader, J.C Woods. *Survey on chatbot design techniques in speech conversation systems*. **International Journal of Advanced Computer Science and Applications**. 2015;6(7).
67. S Arsovski, H Osipyan, M.I Oladele, A.D Cheok. *Automatic knowledge extraction of any Chatbot from conversation*. *Expert Systems with Applications*. 2019 Dec 15;137:343-8
68. Y Hahm, J Kim, S An, M Lee, KS Choi. *Chatbot Who Wants to Learn the Knowledge: KB-Agent*. Semdeep/NLIWoD@ ISWC. 2018 Oct 9;4.
69. B Kunal. *Conversational AI Chatbot using Deep Learning: How Bi-directional LSTM, Machine Reading Comprehension, Transfer Learning, Sequence to Sequence Model with multi-headed attention mechanism, Generative Adversarial Network, Self Learning based sentiment analysis*, 2019
70. <https://www.microsoft.com/en-us/research/project/deep-reinforcement-learning-goal-oriented-dialogue/>

71. V Ilievski. *Building advanced dialogue managers for goal-oriented dialogue systems*. arXiv preprint arXiv:1806.00780. 2018 Jun 3.
72. R Qader, F Portet, C Labbé. *Semi-supervised neural text generation by joint learning of natural language generation and natural language understanding models*. arXiv preprint arXiv:1910.03484. 2019 Sep 29.
73. T.H Wen, M Gasic, N Mrksic, P.H Su, D Vandyke, S Young. *Semantically conditioned lstm-based natural language generation for spoken dialogue systems*. arXiv preprint arXiv:1508.01745. 2015 Aug 7.
74. H Chen, X Liu, D Yin, J Tang. *A survey on dialogue systems: Recent advances and new frontiers*. Acm Sigkdd Explorations Newsletter. 2017 Nov 21;19(2):25-35.
75. B Peng, X Li, L Li, J Gao, A Celikyilmaz, S Lee, K.F Wong. *Composite task-completion dialogue policy learning via hierarchical deep reinforcement learning*. arXiv preprint arXiv:1704.03084. 2017 Apr 10.
76. C Buckner. *Deep learning: A philosophical introduction*. *Philosophy compass*. 2019 Oct;14(10):e12625.
77. X Wei, L Zhang, H.Q Yang, L Zhang, Y.P Yao. *Machine learning for pore-water pressure time-series prediction: Application of recurrent neural networks*. *Geoscience Frontiers*. 2021 Jan 1;12(1):453-67.
78. [https://www.tutorialspoint.com/tensorflow/tensorflow\\_multi\\_layer\\_perceptron\\_learning.htm](https://www.tutorialspoint.com/tensorflow/tensorflow_multi_layer_perceptron_learning.htm)
79. J.A Díaz León, M.A Olarte Escobar, M Jara García. *The Prediction of Hydrometeorology Variables Using the Method of Recurrent Neuronal Networks*. In *International Conference on Water Resource and Environment 2022* Nov 1 (pp. 15-32). Singapore: Springer Nature Singapore.
80. <http://colah.github.io/posts/2015-09-NN-Types-FP/>
81. M Eltay, A Zidouri, I Ahmad. *Exploring deep learning approaches to recognize handwritten arabic texts*. *IEEE Access*. 2020 May 13;8:89882-98.
82. Y Yu, X Si, C Hu, J Zhang. *A review of recurrent neural networks: LSTM cells and network architectures*. *Neural computation*. 2019 Jul 1;31(7):1235-70.
83. K.H Ly, A.M Ly, G Andersson. *A randomized controlled trial of a chatbot-based intervention for depression*. *Frontiers in psychology*. 2020;11:491.
84. B Xie, Z Su, W Zhang, L Cui. *Chatbots in mental health: A review of the recent progress*. **Journal of psychiatric research**. 2021;145:24-33.

85. B Xie, D He, T Mercer, Y Wang. *Chatbots in mental health: A review of the recent progress*. *IEEE Transactions on Neural Systems and Rehabilitation Engineering*. 2021;29:1-1.
86. J Lee, J Kim, J.H Kim. *Development of a machine learning-based chatbot for mental health support*. **Journal of medical Internet research**. 2020;22(10):e21580
87. J Lee, J Kim, M Jo, H Lee, J Kim, S Lee. *Designing an empathetic chatbot for mental health counseling using machine learning*. **Journal of medical systems**. 2020;44(9):1-10.
88. J Lee, Y Lee, J Song, J Kim, J Kim. *Designing a machine learning-based chatbot for mental health support*. **Journal of medical systems**. 2020;44(7):1-10.
89. A Smith, J Chen, L.F Salazar, E Horvitz. *Designing a chatbot for mental health support using machine learning: A user-centered approach*. **JMIR mental health**. 2020;7(9):e21457.
90. J Lee, J Kim, J Lee, J Kim, H Kim, Y Kim. *Designing a machine learning-based chatbot for mental health support*. **Healthcare informatics research**. 2020;26(4):290-8.
91. Y Wang, Y Zhang, L Huang, Y Liu, Y Liu, C Cui. *Designing an artificial intelligence-based chatbot for mental health support*. **International Journal of Medical Informatics**. 2021;149:104441.
92. K Denecke, S Vaaheesan & A Arulnathan. *A mental health chatbot for regulating emotions (SERMO)-concept and usability test*. *IEEE Transactions on Emerging Topics in Computing*, 2020: 9(3), pp.1170-1182
93. Y Lee, N Yamashita, Y Huang. *Designing a chatbot as a mediator for promoting deep self-disclosure to a real mental health professional*. *Proceedings of the ACM on Human-Computer Interaction*. 2020 May 28;4(CSCW1):1-27.
94. S Park, J Choi, S Lee, C Oh, C Kim, S La, J Lee, B Suh. *Designing a chatbot for a brief motivational interview on stress management: Qualitative case study*. **Journal of medical Internet research**. 2019 Apr 16;21(4):e12231.
95. D Calvaresi, J.P Calbimonte, E Siboni, S Eggenschwiler, G Manzo, R Hilfiker, M Schumacher. *EREBOTS: Privacy-compliant agent-based platform for multi-scenario personalized health-assistant chatbots*. *Electronics*. 2021 Mar 12;10(6):666.
96. I Hungerbuehler, K Daley, K Cavanagh, H Garcia Claro, M Kapps. *Chatbot-based assessment of employees' mental health: Design process and pilot implementation*. **JMIR Formative Research**. 2021 Apr 21;5(4):e21678.

97. G Anmella, M Sanabra, M Primé-Tous, X Segú, M Cavero, I Morilla, I Grande, V Ruiz, A Mas, I Martín-Villalba, A Caballo. *Vickybot, a Chatbot for Anxiety-Depressive Symptoms and Work-Related Burnout in Primary Care and Health Care Professionals: Development, Feasibility, and Potential Effectiveness Studies*. **Journal of medical Internet research**. 2023 Apr 3;25:e43293.
98. S Sabour, W Zhang, X Xiao, Y Zhang, Y Zheng, J Wen, J Zhao, M Huang. *A chatbot for mental health support: exploring the impact of Emohaa on reducing mental distress in China*. *Frontiers in Digital Health*. 2023 May 4;5:1133987.
99. M Mittal, G Battineni, D Singh, T Nagarwal, P Yadav. *Web-based chatbot for frequently asked queries (FAQ) in hospitals*. **Journal of Taibah University Medical Sciences**. 2021 Oct 1;16(5):740-6.
100. J.M Noble, A Zamani, M Gharaat, D Merrick, N Maeda, A.L Foster, I Nikolaidis, R Goud, E Stroulia, V.I Agyapong, A.J Greenshaw. *Developing, implementing, and evaluating an artificial intelligence-guided mental health resource navigation chatbot for health care workers and their families during and following the COVID-19 pandemic: protocol for a cross-sectional study*. **JMIR Research Protocols**. 2022 Jul 25;11(7):e33717.

Do Not Copy, Lead City University, Nigeria

## Chapter Three

### Methodology

In this chapter, the methodology used for developing the web-based chatbot for personalized therapy system for mental health was presented. The chapter will cover the overall research design, data collection and preprocessing, system architecture, model development, and evaluation approach. The chapter focused on the analysis of a mental health-related FAQs dataset and the creation of a neural network model for question classification. By performing EDA, insights into the dataset's content and distribution of words was known. The trained model holds the potential to effectively categorize new questions based on their associated question IDs, facilitating streamlined responses in mental health-related applications. The chapter provided a practical demonstration of data preprocessing, analysis, and model training, contributing to the overall development of intelligent systems in the mental health domain.

#### 3.1 Research Approach

To train and test the chatbot system, a suitable dataset related to mental health inquiries and responses is required. Data collection involves obtaining textual data from reliable sources, such as mental health forums, online therapy platforms, and existing datasets. The collected data will be preprocessed to remove noise, handle missing values, and perform text normalization techniques like tokenization and stemming. Furthermore, the minimum hardware requirements and software needs will be covered.

#### 3.2 Requirement Specification

**Hardware Requirements:** The study will be conducted on a personal computer with 16 GB of RAM and a 2.2 GHz Intel Core i7 processor. Anaconda was used as the

development environment alongside Spyder which is a lunch tool for scientific analysis using Python in the Anaconda environment. The interface of Anaconda and Jupyter is shown in figure 3.1 below

**Software Requirements:** The software used in this study includes Python programming language, Jupyter Notebook, and various libraries, including keras models, Pandas, NumPy, Matplotlib, and Seaborn.

Keras is a high-level neural network API that simplifies the process of building and training deep learning models<sup>1</sup>. It provides a user-friendly interface for defining, configuring, and training neural networks. Keras models are the core component of building deep learning models using the Keras API<sup>1</sup>.

**Sequential Model:** The Sequential model is the simplest and most common type of model in Keras. It is a linear stack of layers, where each layer is added sequentially. The Sequential model is suitable for building feedforward networks, where the data flows straight through the layers from input to output<sup>2</sup>.

**Pandas:** Pandas is an open-source data manipulation and analysis library for Python. It provides data structures for efficiently storing and manipulating large datasets, as well as tools for filtering, grouping, merging, and transforming data<sup>3</sup>. Pandas is commonly used in data science workflows for data cleaning, preparation, and exploratory data analysis.

**NumPy:** NumPy is another popular open-source library for scientific computing in Python. It provides tools for working with arrays, matrices, and other numerical data structures, and supports a wide range of mathematical and statistical operations<sup>4</sup>. NumPy is widely used in scientific computing, data analysis, and machine learning.

**Matplotlib:** Matplotlib is a plotting library for Python. It provides a range of tools for creating static, animated, and interactive visualizations in Python, and is highly

customizable<sup>5</sup>. Matplotlib is commonly used in data science workflows for data visualization and exploratory data analysis.

Seaborn: Seaborn is a data visualization library for Python. It provides a range of high-level functions for creating statistical graphics such as heatmaps, scatterplots, and bar plots, and supports more complex visualizations such as joint plots and regression plots<sup>6</sup>. Seaborn is built on top of Matplotlib, and is designed to work seamlessly with Pandas data structures. It is commonly used in data science workflows for data visualization and exploratory data analysis.

### 3.3 System Design

This describes the overall structure and components of the web-based chatbot system. It includes the frontend, backend, and the interface between the two. The frontend will consist of an intuitive and interactive user interface, allowing users to input their inquiries and receive responses. The backend will incorporate the natural language processing (NLP) components responsible for understanding and analyzing user input, as well as the machine learning algorithms for providing personalized recommendations and assistance

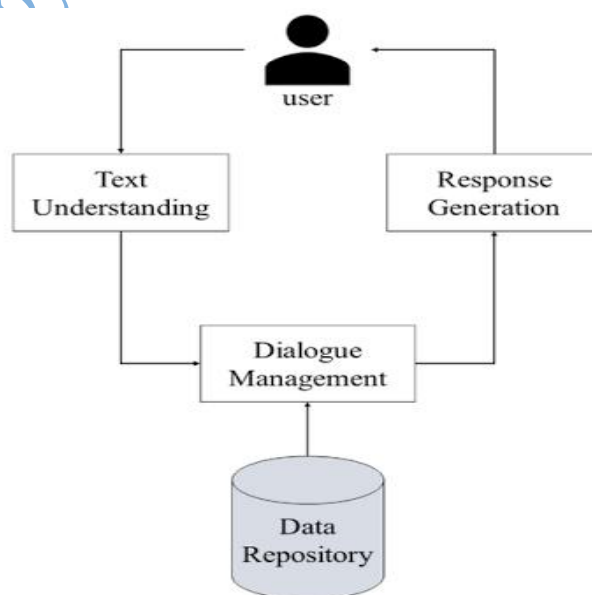


Figure 3.1: Mental Health Chatbot Design<sup>5</sup>.

### **3.3.1 Model Development**

The model development process involved incorporating natural language processing techniques and sequential models. NLP techniques such as word embedding, named entity recognition, and sentiment analysis was employed to accurately process and analyze user textual input. Sequential models, such as recurrent neural networks (RNNs) or transformer models, was utilized to learn patterns and generate contextually relevant responses. The model was trained on the preprocessed dataset, and hyperparameter tuning techniques was applied to optimize the model's performance.

Sequential models, such as recurrent neural networks (RNNs) and transformer models, play a crucial role in natural language processing (NLP) tasks and chatbot systems. These models excel at understanding and generating sequential data, making them well-suited for language-related tasks.

### **3.4 Research Method**

#### **3.4.1 Data Collection and Preprocessing**

This study used an open source dataset. The dataset is available on kaggle, <https://www.kaggle.com/datasets/narendrageek/mental-health-faq-for-chatbot>, this dataset consists of FAQs about Mental Health.

#### **3.4.2 Dataset Description**

The dataset contains 3 features which are: Unique\_ID, Questions, Answers

- i. Unique\_ID – unique id of each questions
- ii. Questions: question related to mental health
- iii. Answers: corresponding answer

### 3.4.3. Data Processing

**Word Cloud Visualization:** A word cloud is generated to provide a visual representation of the most frequent words in the dataset. Word clouds are helpful for gaining insights into the overall distribution and importance of words in the dataset.

**Common Word Analysis:** Using the Counter class in the collections library, how often each word appeared in the questions was counted. Through this research, most frequently used terms in the repository was isolated . The top N most frequent terms were graphically represented as a bar chart in matplotlib.

**Word Length Distribution Analysis:** The length of each word in every question was calculated, and a histogram depicting the distribution of word lengths was generated.

Word length distribution in the dataset was plotted as a histogram.

**Tokenization:** The questions in the dataset are tokenized using the nltk.word\_tokenize() function. Tokenization splits the text into individual words or tokens, making it easier to process and analyze the textual data.

**Lemmatization:** The lemmatization process is applied to the tokenized words using the WordNetLemmatizer from the nltk.stem module. Lemmatization reduces words to their base or root form, which helps to normalize the text and reduce variations in word forms.

**Building Vocabulary:** The lemmatized words are sorted and converted into a vocabulary by removing duplicates and storing them in a list. This vocabulary represents the unique words found in the dataset and forms the basis for creating the bag-of-words representation.

**Word Frequency Analysis:** The Counter class from the collections module is used to count the frequency of each word in the dataset. This analysis provides insights into

the most common words and their occurrence frequencies, which can help in understanding the dataset's characteristics.

**Bag-of-Words Representation:** For each question in the dataset, a bag-of-words representation is created. This representation captures the presence or absence of each word in the vocabulary for a particular question. It is a binary vector where each element indicates whether a word is present (1) or absent (0) in the question.

**Shuffle and Numpy Conversion:** The training list, containing the bag-of-words representations and output labels, is shuffled to ensure randomness during training. Then, it is converted to a numpy array for compatibility with the Keras model.

### 3.5 Neural Network Model Creation and Training

In order to train the model, the training data was produced by generating a bag-of-words representation for each question, denoting the inclusion or exclusion of terms from the lexicon. Output vectors were constructed in which a value of "1" was assigned to the index that corresponds to the class (question ID) of the question, while the remaining indices were assigned a value of "0". A neural network model was developed with the Keras package. The model's architecture consisted of dense units and dropout layers implemented for the purpose of regularization. The model was assembled utilizing categorical cross-entropy loss and trained employing stochastic gradient descent (SGD) as the optimizer. The progress of the training was documented in the variable named "hist". An alternative neural network model was defined and trained using TensorFlow's Keras API, providing a different approach for model creation and training.

### 3.6 Evaluation Process

After training the model, evaluation metrics are used to assess its performance. The primary evaluation metric used is accuracy, which indicates the proportion of

correctly classified instances out of the total instances. Here's a breakdown of the evaluation metrics used and their significance:

**Accuracy (Training and Validation Accuracy):** Accuracy is a commonly used evaluation metric for classification tasks. It measures the ratio of correctly predicted labels to the total number of instances in a dataset. It provides an overall understanding of the model's correctness. The training accuracy is calculated on the training data, while the validation accuracy is calculated on the separate validation dataset. It calculates training and validation accuracy at each epoch during the training process. It plots the training accuracy over epochs to visualize how the model's accuracy improves over time.

**Accuracy Plot:** In the accuracy plot, you can observe how the training accuracy increases as the number of training epochs progresses. The validation accuracy gives insight into how well the model generalizes to unseen data.

**Loss (Training Loss and Validation Loss):** Although not explicitly calculated or plotted in this code, loss is another crucial evaluation metric. Loss represents the difference between the predicted values and the actual values. Lower loss values indicate better alignment between predicted and actual values. The loss values are used as the optimization objective for the model during training. The model aims to minimize the loss function, which leads to better predictions.

The trained neural network model was saved to a file named "model.counsellor\_new" using the `model.save()` function. This ensured that the trained model could be reused for future classifications without retraining.

## Endnotes

1. J Moolayil, J Moolayil. *An introduction to deep learning and keras. Learn Keras for Deep Neural Networks: A Fast-Track Approach to Modern Deep Learning with Python*. 2019:1-6.
2. E Bisong, E Bisong. Recurrent Neural Networks (RNNs). Building machine learning and deep learning models on google cloud platform: A comprehensive guide for beginners. 2019:443-73.
3. F Reiss, B Cutler, Z Eichenberger. *Natural language processing with pandas dataframes*. In Proc. Of The 20th Python In Science Conf.(Scipy 2021) 2021 (pp. 49-58).
4. C.R Harris, K.J Millman, S.J Van Der Walt, R Gommers, P Virtanen, D Cournapeau, E Wieser, J Taylor, S Berg, N.J Smith, R Kern. *Array programming with NumPy*. *Nature*. 2020 Sep 17;585(7825):357-62
5. G Herda, R McNabb. *Python for Smarter Cities: Comparison of Python libraries for static and interactive visualisations of large vector data*. arXiv preprint arXiv:2202.13105. 2022 Feb 26
6. W.R Paczkowski. *Data Visualization: The basics*. *Business analytics: Data science for business problems*. 2021:85-126.
7. W.F Wiggins, F Kitamura, I Santos, L.M Prevedello. *Natural language processing of radiology text reports: interactive text classification*. *Radiology: Artificial Intelligence*. 2021 May 12;3(4):e210035.
8. C Toraman, E.H Yilmaz, F Şahinuç, O Ozcelik. *Impact of tokenization on language models: An analysis for turkish*. *ACM Transactions on Asian and Low-Resource Language Information Processing*. 2023 Mar 25;22(4):1-21.
9. A Samir, Z Lahbib. *Stemming and lemmatization for information retrieval systems in amazigh language*. In *Big Data, Cloud and Applications: Third International Conference, BDCA 2018, Kenitra, Morocco, April 4–5, 2018, Revised Selected Papers 3 2018* (pp. 222-233). Springer International Publishing.


## Chapter Four

### Results and Discussion of Finding

The results that were gained through the development and evaluation of the web-based chatbot system for personalised therapy will be given and discussed in this chapter. Specifically, this chapter will focus on the presentation of the results. This chapter will also explain the objectives that were reached, as well as cover the performance indicators, and analyse the feedback from users.

#### 4.1 Result on Data Collection

The study used an open source dataset which consists of FAQs about Mental Health. The dataset contains 98 records and 3 features (columns). The dataset contains 3 features Unique\_ID (unique id of each questions), Questions (question related to mental health), Answers (corresponding answer).



Out[ ]:	Question_ID	Questions	Answers
0	1590140	What does it mean to have a mental illness?	Mental illnesses are health conditions that di...
1	2110618	Who does mental illness affect?	It is estimated that mental illness affects 1 ...
2	6361820	What causes mental illness?	It is estimated that mental illness affects 1 ...
3	9434130	What are some of the warning signs of mental i...	Symptoms of mental health disorders vary depen...
4	7657263	Can people with mental illness recover?	When healing from mental illness, early identi...

**Figure 4.1:** Screenshot of the Study Dataset

**Source:** Research Work, 2023

### 4.1.1 Dataset Preprocessing

Question, answer and question ID was extracted from the dataset using the snippet code below

```
# Extract questions, answers, and question IDs from the dataset
questions = dataset['Questions'].tolist()
answers = dataset['Answers'].tolist()
question_ids = dataset['Question_ID'].tolist()
```

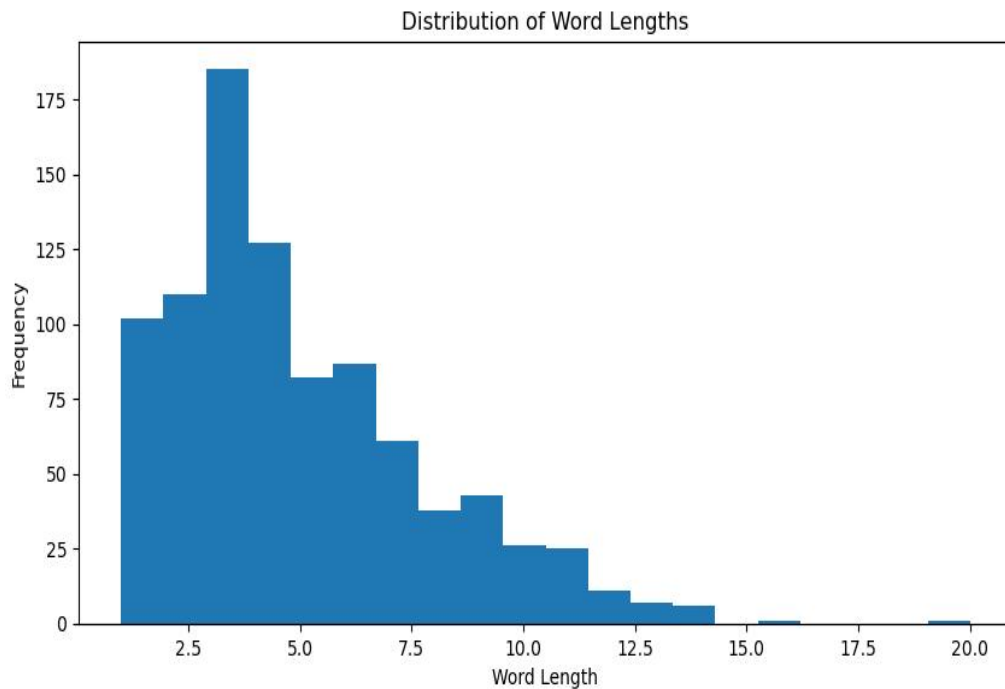
Word cloud was generated using Python's `wordcloud` library. The width=800, height=400 define the dimensions (in pixels) of the image that will be generated for the word cloud and the generated word cloud was displayed using matplotlib as shown in figure 4.2. The analysis of word frequency in the questions reveals that the most commonly words in the questions are, find, help, health, difference, information, care, someone, warning, seems signs, suicide, eating grief, professional, dissociative, depression, lonely just to mention a few. The snippet code to generate is given below

```
# Generate word cloud
wordcloud = WordCloud(width=800,
height=400).generate(text)
# Display the generated word cloud using matplotlib
plt.figure(figsize=(10, 5))
plt.imshow(wordcloud,
interpolation='bilinear')
plt.axis('off')
plt.savefig('wordcloud.png')
plt.show()
```

Each word was lemmatized and lowercased. Also, duplicate words were removed, words and class sorted. Over all, 98 classes and 259 unique lemmatized words was obtained (attached in Appendix I)

Do





**Figure 4.4: Distribution of Word Lengths**

Source: Research Work, 2023

## 4.2 Model Development

Figure 4.5 shows the result of the model. The model is based on Long Short-Term Memory (LSTM) architecture. The model is developed by the training of a machine learning model (neural network). The training data was created using the snippet below.

```
# Create our training data
training = [[output_empty = [0] * len(classes)
for doc in documents:
    bag = []
    pattern_words = doc[0]
    pattern_words = [lemmatizer.lemmatize(word.lower()) for word in pattern_words]
    for w in words:
        bag.append(1) if w in pattern_words else bag.append(0)
    output_row = list(output_empty)
    output_row[classes.index(doc[1])] = 1
    training.append([bag, output_row])
```

After the training data has been created, the model was created, compiled and fitted

```
# Create modelmodel =
Sequential()
```

```

model.add(Dense(128, input_shape=(len(train_x[0]),), activation='relu'))
model.add(Dropout(0.5))
# Compile model
sgd = SGD(lr=0.01, decay=1e-6, momentum=0.9, nesterov=True)
model.compile(loss='categorical_crossentropy', optimizer=sgd, metrics=['accuracy'])
# Fit the model
hist = model.fit(np.array(train_x), np.array(train_y), epochs=200,
batch_size=5, verbose=1)

```

The model was trained to keep track of the training and validation loss. The training and validation loss was plotted as shown in figure 4.6 and 4.8

Epochs: This represents the number of complete passes through the entire training dataset. It looks like you've trained your model for 200 epochs.

20/20: Within each epoch, the data are processed in batches, with each batch containing 20 samples. This is a common practice in training machine learning models to efficiently update the model's parameters.

Loss and Accuracy: These metrics are crucial for evaluating the performance of your model during training.

Loss: The loss value (e.g., 0.7734, 0.7987, etc.) is a measure of how well or poorly the model is performing. It represents the error between the model's predictions and the actual target values. The goal during training is typically to minimize this loss. There is a fluctuation during different epochs, indicating that the model's performance varies throughout training.

Accuracy: The accuracy value (e.g., 0.7755, 0.8571, etc.) represents the proportion of correctly classified samples in the training data.

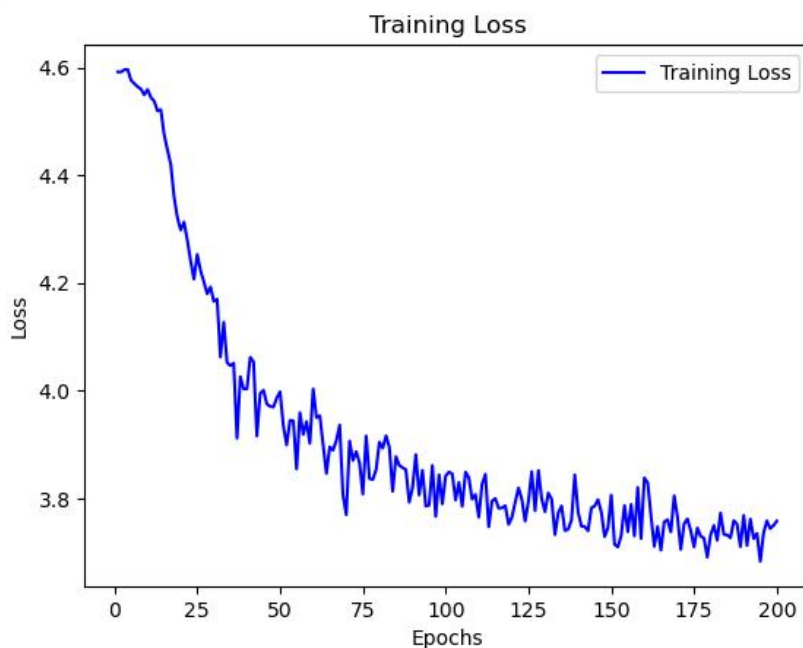
Time per Step: The "0s 4ms/step" or similar notations represent the time taken for each training step (each batch) in milliseconds. This can be useful for monitoring the efficiency of the training process.

```

Epoch 186/200
20/20 [=====] - 0s 4ms/step - loss: 0.7734 - accuracy: 0.7755
Epoch 187/200
20/20 [=====] - 0s 4ms/step - loss: 0.7987 - accuracy: 0.7755
Epoch 188/200
20/20 [=====] - 0s 4ms/step - loss: 0.7420 - accuracy: 0.7449
Epoch 189/200
20/20 [=====] - 0s 4ms/step - loss: 0.6010 - accuracy: 0.8571
Epoch 190/200
20/20 [=====] - 0s 4ms/step - loss: 0.7014 - accuracy: 0.8061
Epoch 191/200
20/20 [=====] - 0s 4ms/step - loss: 0.5872 - accuracy: 0.7857
Epoch 192/200
20/20 [=====] - 0s 4ms/step - loss: 0.7251 - accuracy: 0.7959
Epoch 193/200
20/20 [=====] - 0s 4ms/step - loss: 0.6004 - accuracy: 0.7551
Epoch 194/200
20/20 [=====] - 0s 4ms/step - loss: 0.9680 - accuracy: 0.7143
Epoch 195/200
20/20 [=====] - 0s 4ms/step - loss: 0.8691 - accuracy: 0.7041
Epoch 196/200
20/20 [=====] - 0s 4ms/step - loss: 0.7204 - accuracy: 0.7959
Epoch 197/200
20/20 [=====] - 0s 4ms/step - loss: 0.6995 - accuracy: 0.7857
Epoch 198/200
20/20 [=====] - 0s 4ms/step - loss: 0.5310 - accuracy: 0.7857
Epoch 199/200
20/20 [=====] - 0s 4ms/step - loss: 0.5523 - accuracy: 0.8469
Epoch 200/200
20/20 [=====] - 0s 4ms/step - loss: 0.8582 - accuracy: 0.7245

```

**Figure 4.5: The Result of the Model**  
**Source:** Research Work, 2023

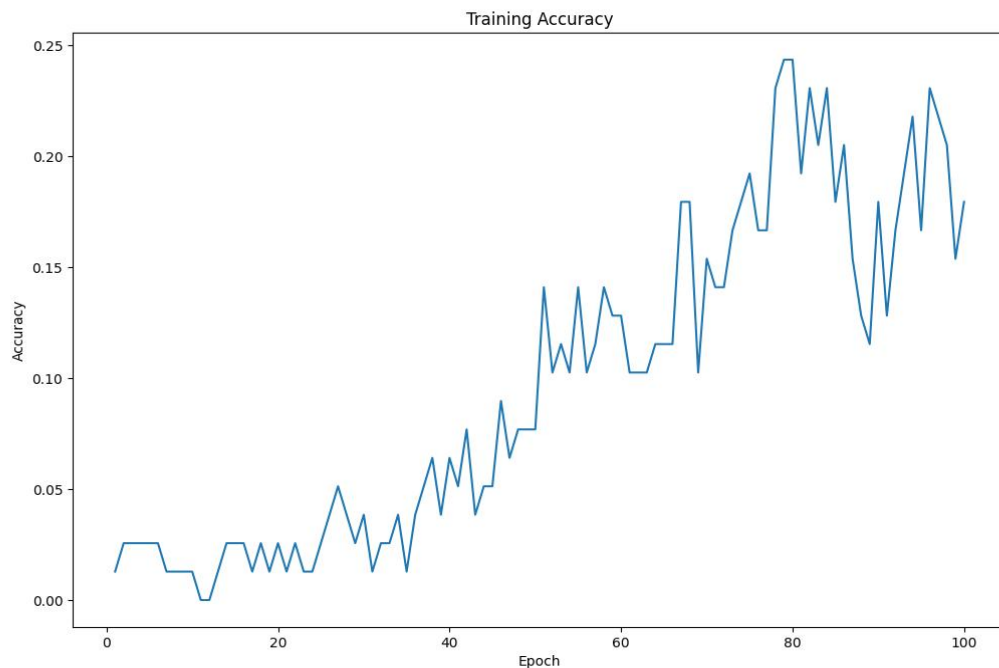


**Figure 4.6: Training Loss Graph**  
**Source:** Research Work, 2023

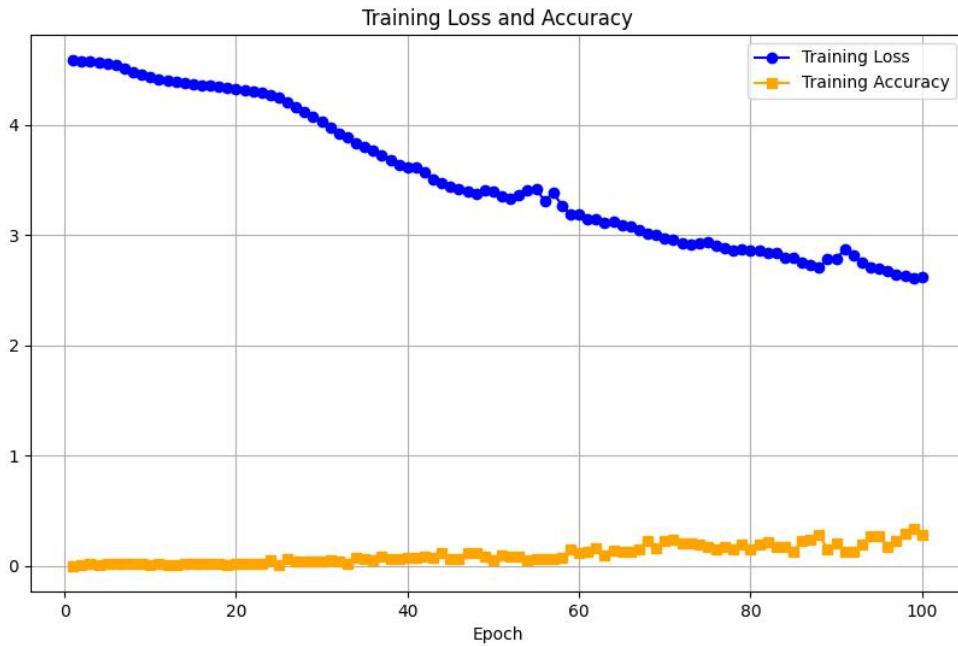
The plot shows the loss value over the epochs from 0 to 200, the result shows that the higher epochs, the lower the loss. The findings indicate a negative correlation between

the number of epochs and the magnitude of loss, suggesting that as the number of epochs increases, the loss decreases. This suggests that the model achieves a lower loss value as the number of epochs increases. This implies that the model get better loss value the more the epochs.

Decreasing loss values over epochs indicate that the model is progressively improving its ability to make predictions on the training data. It's learning to minimize the difference between its predictions and the actual target values. The model is finding the parameter values (weights and biases) that result in better predictions on the training data. The training process involves an optimization algorithm (e.g., stochastic gradient descent) that adjusts the model's parameters to minimize the loss function. Over epochs, the algorithm fine-tunes these parameters to reach a better solution.



**Figure 4.7: Training Accuracy Graph**  
Source: Research Work, 2023



**Figure 4.8: Training loss and Validation Loss Graph**

Source: Research Work, 2023

The graph illustrates a typical relationship between loss and accuracy during training: as the model learns, the loss decreases, and the accuracy increases. As the number of epochs increases, the validation loss has a tendency to have higher loss values, although the training loss remains low with increasing epochs.

#### 4.2.1 Model Deployment

The implementation of the chat paradigm utilized Flask, a widely adopted Python framework. The process of deployment encompasses a series of sequential processes that must be followed with precision. These processes are designed to assure the appropriate implementation of a system or application. In order to guarantee a successful deployment, it is imperative that following steps are implemented.

The process of loading dependencies entails the importation of libraries and dependencies. The inclusion of Flask, Keras, NLTK, and other significant libraries and dependencies is evident. The Keras and pickle library functions are utilized to load the chatbot model, intentions JSON file, and word dictionaries into the system

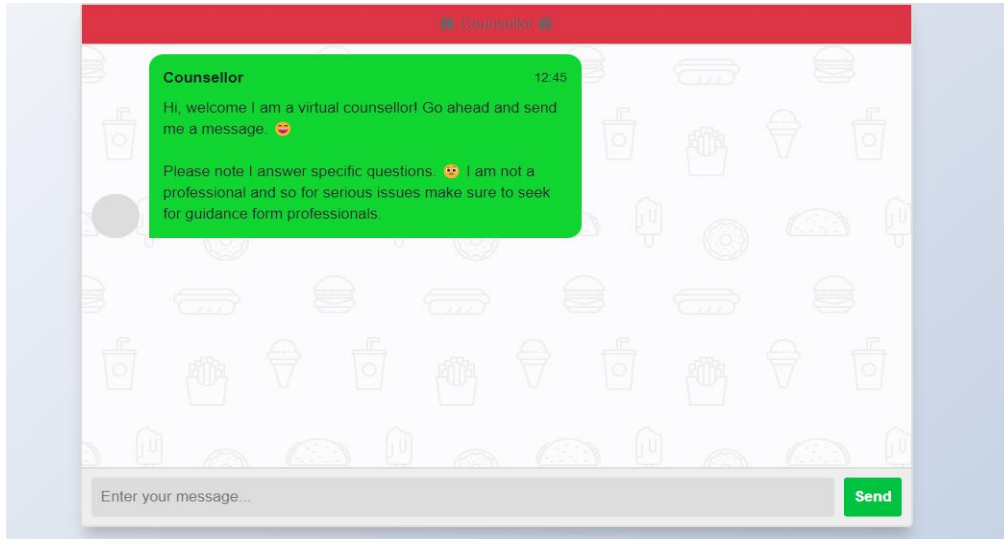
memory. The ultimate stage involves constructing a bag-of-words representation through the process of tallying the frequency of occurrence for each word inside the document. These preparatory activities enhance the quality and relevance of texts both on an individual level and as a whole.

The `predict_class` function utilizes the loaded model to make predictions about the user query intent by employing the preprocessed text. The `getResponse` method is responsible for identifying the necessary intent and retrieving the corresponding response from the JSON file containing the intents. The Flask application has been initialized, and all routes have been meticulously established to facilitate seamless navigation. The home route is responsible for generating an HTML template that serves as the graphical user interface (GUI) for the chat application. The `get` route is responsible for receiving user inquiries and providing the corresponding response from the chatbot.

The Flask framework is executed throughout the deployment process. This initial action initiates the user's default web browser and promptly presents the chat interface.

## 4.2.2 Result of the Web Interface and Some Chat With The AI Chabot

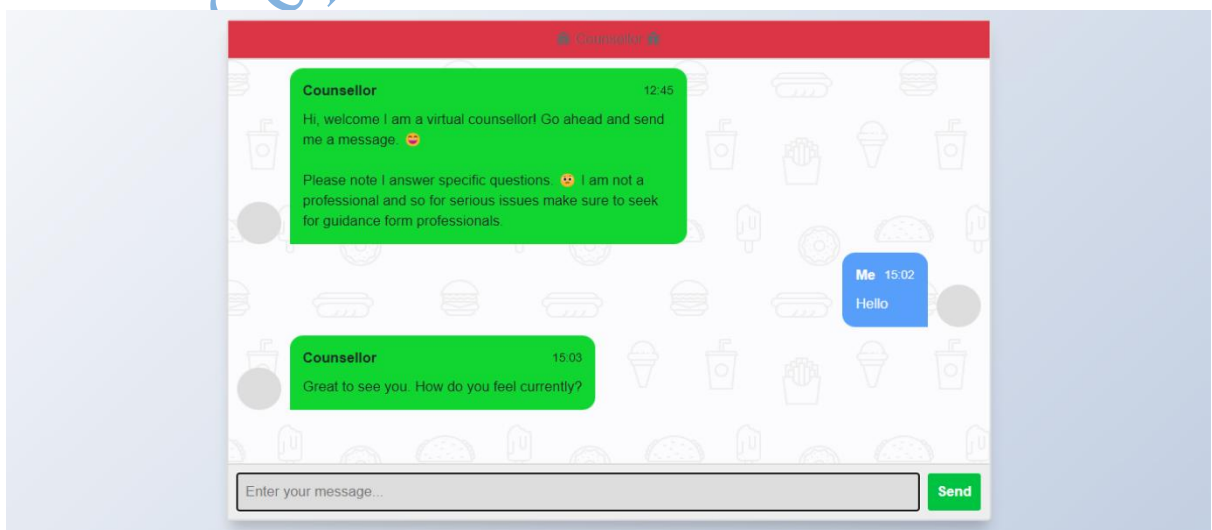
The start page



**Figure 4.9: Chatbot Dialogue**

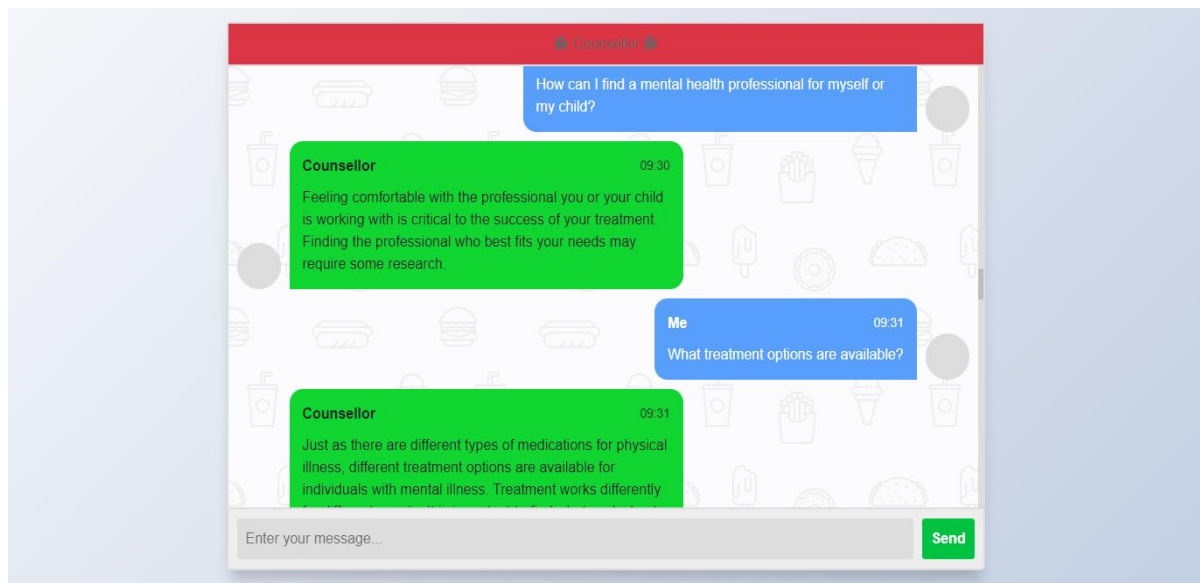
Source: Research Work, 2023

The depicted image is the initial interface of the deployed model on the internet, wherein the chatbot referred to as "Counsellor" introduces itself and provides a disclaimer stating that it is an AI tool designed to assist users, rather than a certified medical expert.



**Figure 4.10: Chatbot Dialogue**

Source: Research Work, 2023



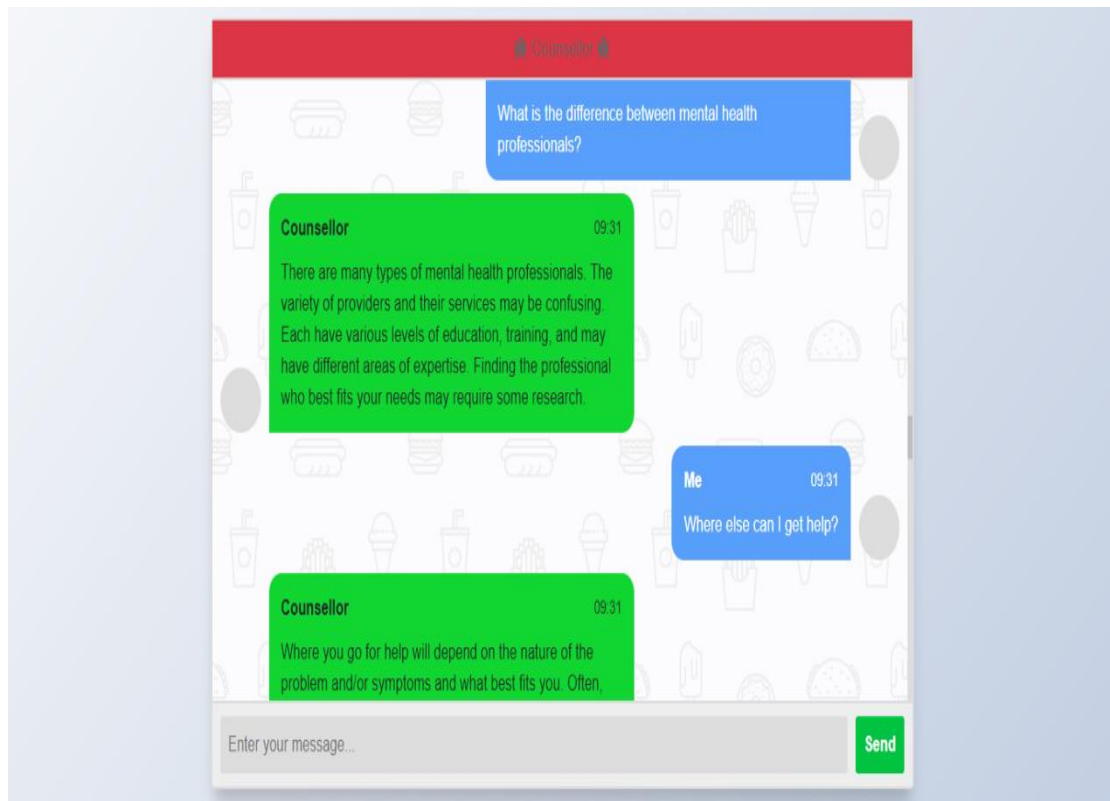
**Figure 4.10: Chatbot Dialogue**

**Source:** Research Work, 2023

The conversation shows a text-based interaction between a user (referred to as "Me") and a virtual counselor or chatbot (referred to as "Counsellor") regarding mental health and finding a mental health professional.

The user initiates the conversation by asking the virtual counselor how to find a mental health professional, either for themselves or their child. The counselor responds by emphasizing the importance of feeling comfortable with the chosen mental health professional. It explain that a positive therapeutic relationship is critical for successful treatment outcomes. In the next message, the counselor broadens the discussion by comparing mental health treatment to physical health.

It explain that, just as there are different medications for physical illnesses, various treatment options are available for individuals with mental health conditions.

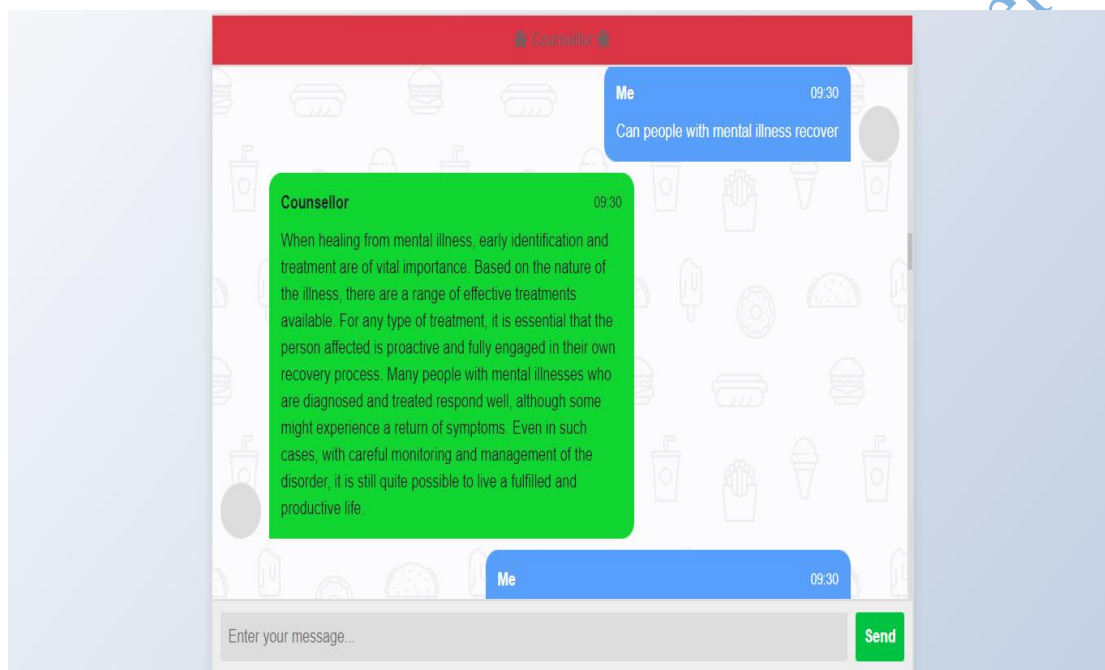


**Figure 4.11: Chatbot Dialogue**  
Research Work, 2023

This conversation involves a user seeking information about the differences between mental health professionals and exploring alternative sources of help. The user initiates the conversation by asking the virtual counselor about the differences between mental health professionals. The counselor (bot) responds by acknowledging the complexity of the mental health field. It explain that there are many types of mental health professionals with varying levels of education, training, and expertise. The counselor advises the user that finding the right mental health professional might require some research. This underscores the importance of making informed choices when seeking help for mental health concerns. The counselor continues by explaining that where one should go for help depends on the nature of the problem or symptoms and what best fits the individual. This highlights the personalized aspect of mental health care and the

need to align treatment with specific needs and circumstances. The user responds with a follow-up question, asking about additional sources of help.

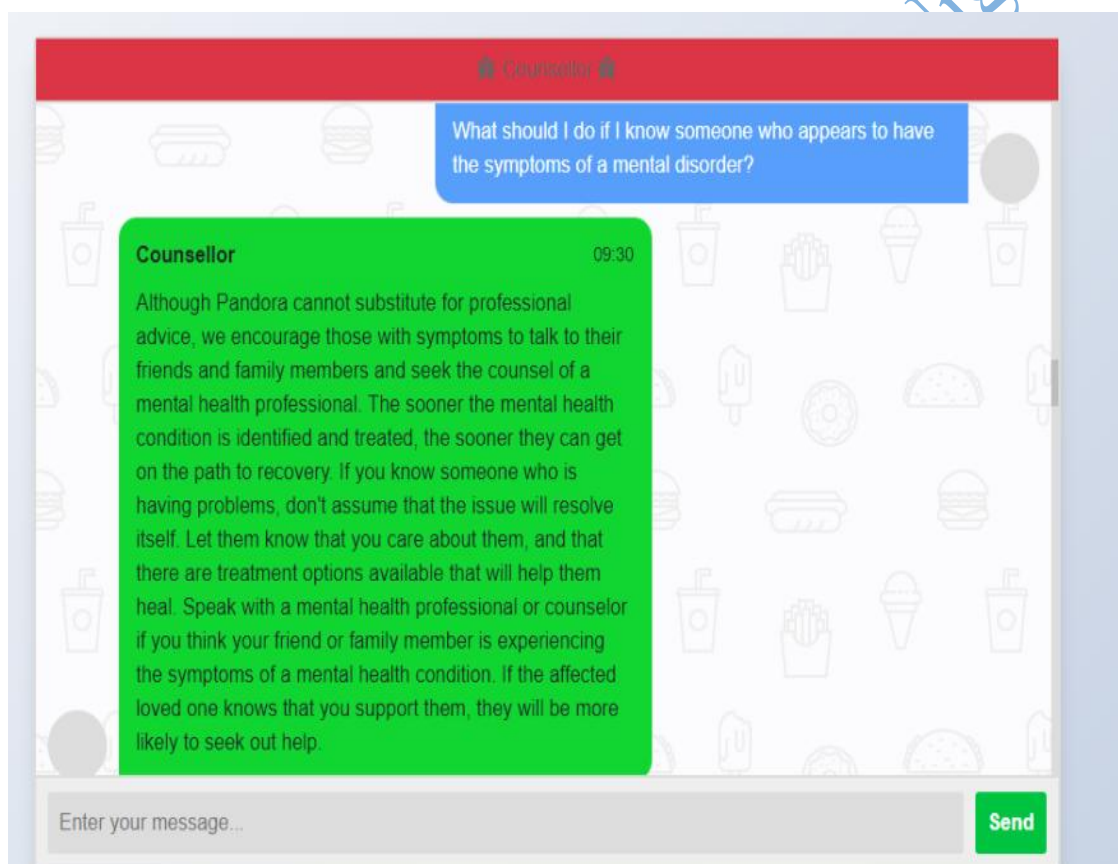
This conversation is aimed at providing initial guidance and information to the user regarding mental health professionals and the importance of finding the right fit. It emphasizes the diversity of mental health providers and the need for research to make informed choices.



**Figure 4.12: Chatbot Dialogue**  
Research Work, 2023

The user initiates the conversation with a straightforward question, inquiring whether people with mental illness can recover. This question reflects a common concern and curiosity about the prospects of recovery from mental health conditions. The counselor (bot) responds by providing a detailed and informative answer. It stress that early intervention is crucial for mental health recovery, highlighting the significance of identifying and addressing mental health issues promptly. The counselor mentions that there are various effective treatments available for mental illnesses. The counselor provides a positive outlook by

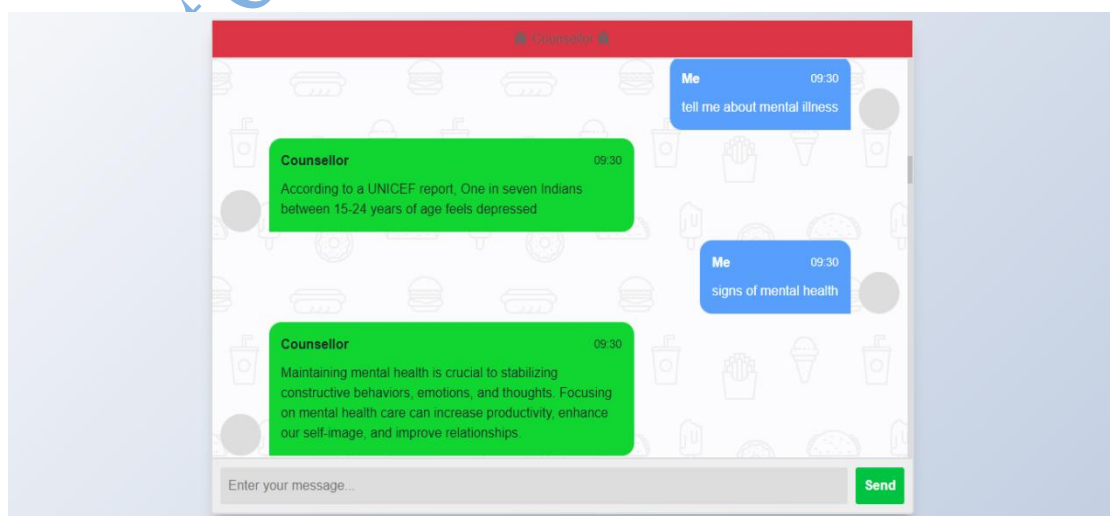
stating that many individuals with mental illnesses who are diagnosed and treated respond well to treatment. Overall, the counselor's response aims to offer reassurance, information, and encouragement. It addresses the user's query by explaining that recovery from mental illness is indeed possible, emphasizing early intervention, engagement, and the effectiveness of available treatments. The message conveys hope and encourages a proactive approach to mental health recovery.



**Figure 4.13: Chatbot Dialogue**  
Research Work, 2023

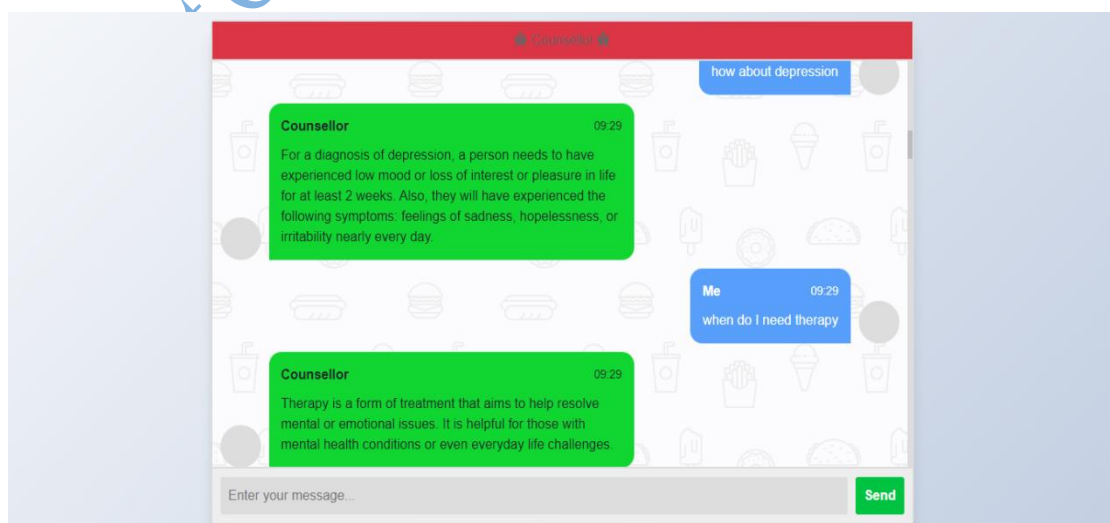
The user initiates the conversation by asking what they should do if they know someone displaying symptoms of a mental disorder. The counselor advises that individuals with symptoms of a mental disorder should talk to their friends and family members. This highlights the importance of open and supportive

communication within relationships. The counselor emphasizes that while the virtual counselor (referred to as "Pandora") cannot substitute for professional advice, it strongly encourages those with symptoms to seek the counsel of a mental health professional. This underscores the significance of consulting with trained experts who can provide accurate assessments and treatment recommendations. The counsellor (bot) also stresses that early identification and treatment of mental health conditions are vital. The counselor encourages users to let their friends and family members know that they care about them. This message of support and empathy can be crucial in destigmatizing mental health issues and encouraging affected individuals to seek help. The counselor recommends speaking with a mental health professional or counselor if someone believes their loved one is experiencing symptoms of a mental health condition. This proactive approach can help guide individuals in supporting their loved ones effectively. The bot counselor highlights that when the affected loved one knows that they have the support of friends and family, they are more likely to seek help. Building trust and maintaining a supportive environment are essential in the recovery process.



**Figure 4.14: Chatbot Dialogue**  
Research Work, 2023

In this brief conversation, the user seeks information about mental illness and the signs of mental health. The user initiates the conversation by asking the counselor to tell them about mental illness. The counselor responds with a statistic from a UNICEF report, stating that one in seven Indians between the ages of 15-24 feels depressed. This statistic highlights the prevalence of depressive feelings among young individuals in India, shedding light on the significance of mental health issues in society. After receiving information about depression, the user follows up by asking about the signs of mental health. The counselor's response emphasizes the importance of maintaining mental health. It explain that doing so is crucial for stabilizing constructive behaviors, emotions, and thoughts. Furthermore, they point out that focusing on mental health care can lead to benefits such as increased productivity, enhanced self-image, and improved relationships. In summary, this conversation provides a brief exchange of information related to mental health and mental illness. It starts with a statistic highlighting depression prevalence and then transitions to discussing the importance of mental health care. While the conversation touches on the subject, it doesn't delve into specific signs of mental health or mental illness.



**Figure 4.15: Chatbot Dialogue**  
Research Work, 2023

In this conversation, the user asks about depression and when therapy might be needed. The virtual counselor (bot) provides a concise but informative response. They explain that for a diagnosis of depression, an individual must have experienced low mood or a loss of interest or pleasure in life for at least two weeks. Additionally, it mention common symptoms associated with depression, such as feelings of sadness, hopelessness, or irritability nearly every day. The user follows up by asking when they might need therapy. The counselor responds by providing a general explanation of therapy. They describe therapy as a form of treatment aimed at helping individuals resolve mental or emotional issues. The counselor highlights that therapy can be beneficial for those with mental health conditions as well as for addressing everyday life challenges. This conversation provides introductory information about depression and therapy. It explains the diagnostic criteria for depression and offers a general description of therapy as a treatment approach for mental and emotional issues. It's worth noting that this is a high-level overview, and specific recommendations for therapy would depend on individual circumstances and the advice of mental health professionals.

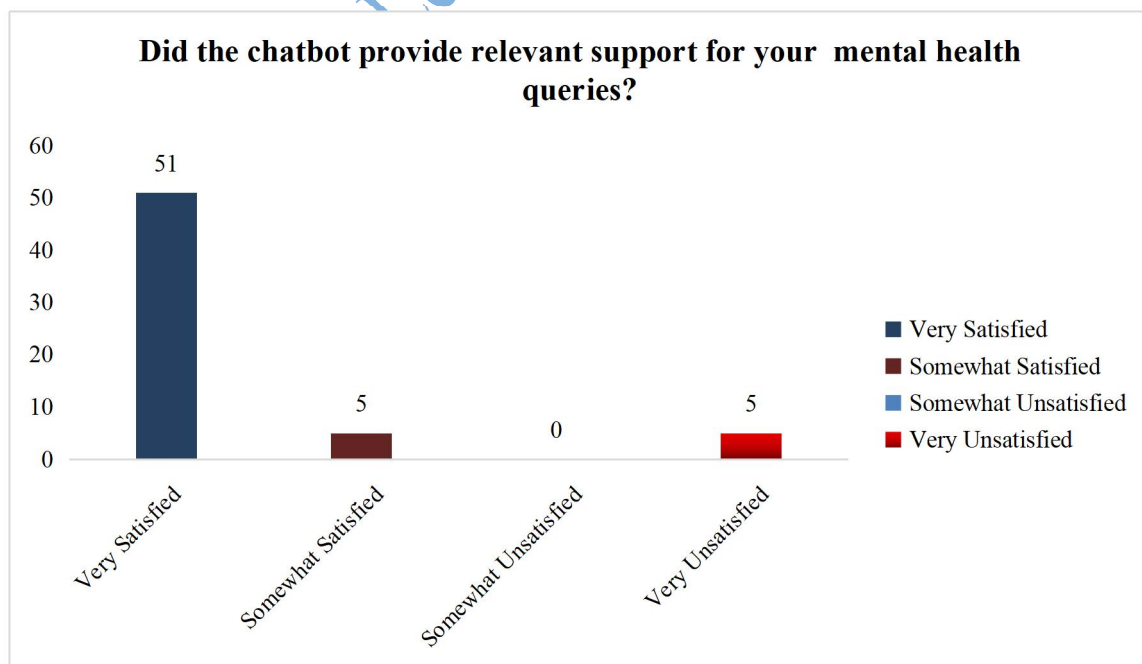
#### **4.3 Result on Performance Evaluation of the Developed System**

The Mental Counselor (bot) evaluation questionnaire was sent via email to sixty respondents and validated. All returned answers were valid and useful for the evaluation. Table 4.1 how the percentage count of respondent's responses. Also, the charts of their responses are shown below

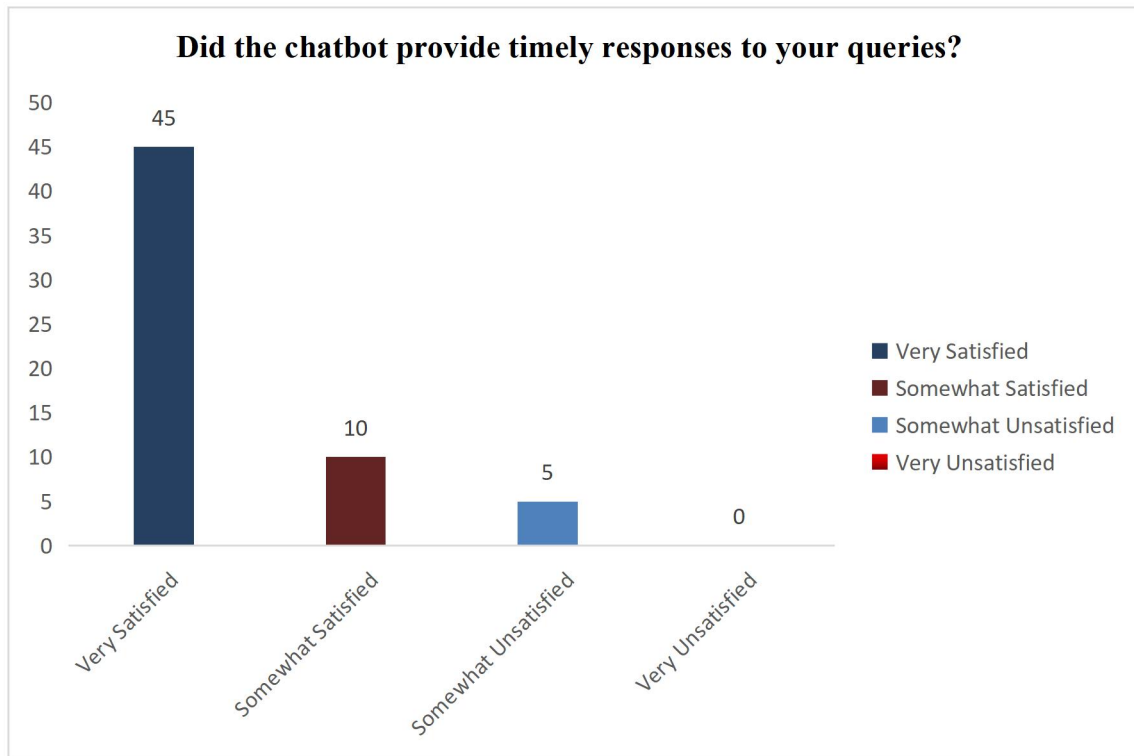
**Table 4.1: Percentage of Respondent Responses**

S/N	Items	VS (%)	SS (%)	SU (%)	VU (%)	Remark
1	Did the chatbot provide relevant personalized support for your mental health queries?	51 (84.6)	5 (7.7)	0 (0)	5 (7.7)	Very Satisfied
2	Did the chatbot provide timely responses to your queries?	45 (75)	10 (16)	5 (8.3)	0 (0)	Very Satisfied
3	Did the chatbot understand your questions and respond appropriately?	37 (61.5)	14 (23.1)	5 (7.7)	5 (7.7)	Very Satisfied
4	Did the chatbot recommend a suitable therapy for your mental health enquiry?	28 (46.2)	23 (38.5)	9 (15.4)	0 (0)	Very Satisfied
5	How would you rate the overall performance of the chatbot?	45 (76.9)	9 (15.4)	5 (7.7)	0 (0)	Very Satisfied
S/N	Items	VL (%)	SL (%)	SUL (%)	VUL (%)	Remark
1	Would you recommend this chatbot?	51 (84.6)	5 (7.7)	0 (0)	5 (7.7)	Very Likely
2	Would you use the chatbot again in the future?	46 (76.9)	9 (15.4)	0 (0)	5 (7.7)	Very Likely

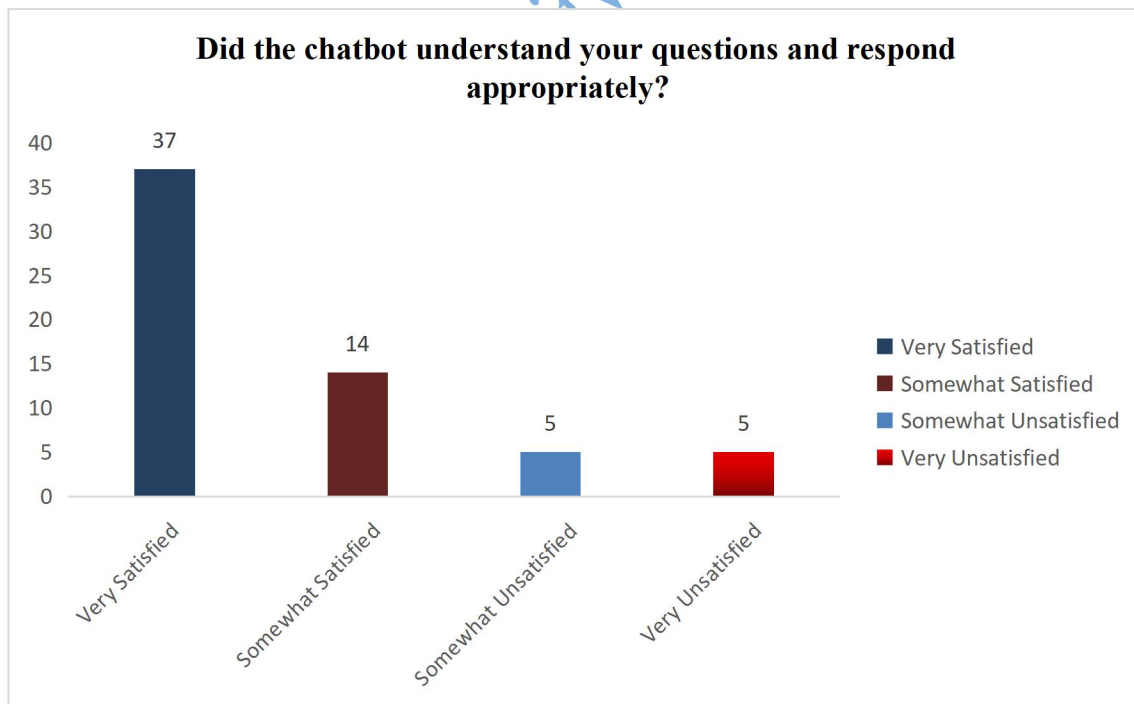
**KEY:** VS = Very Satisfied, SS = Somewhat Satisfied, SU = Somewhat Unsatisfied (2) and VU= Very Unsatisfied, VL= Very Likely, SL = Somewhat Likely, SUL = Somewhat Unlikely, (2) and VUL= Very Unlikely



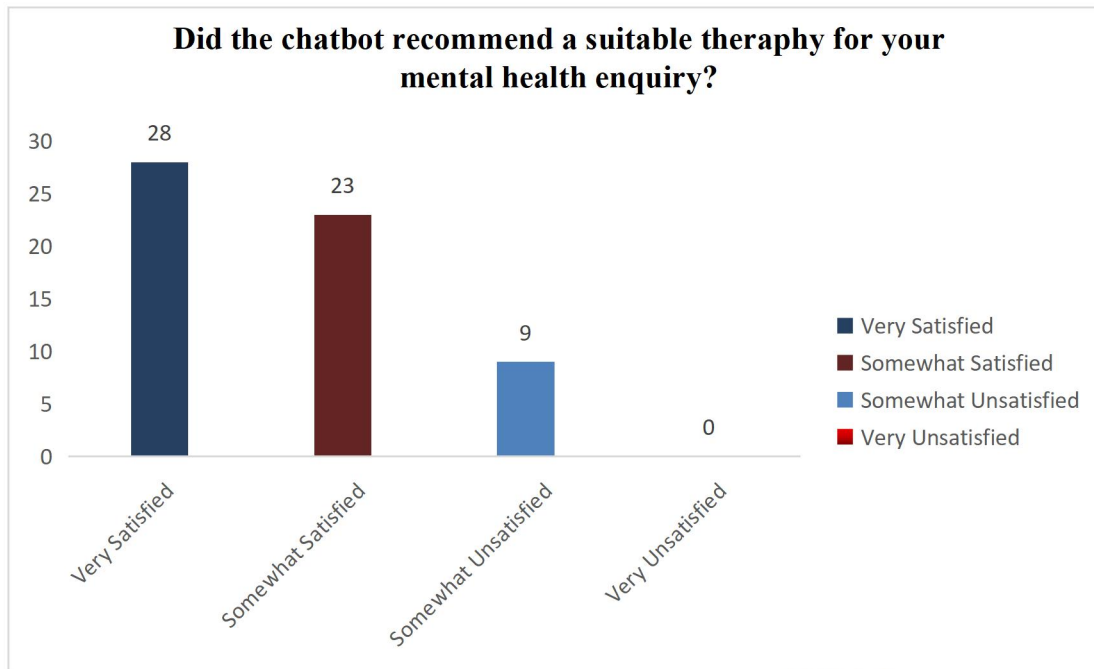
**Figure 4.16:** Relevance of Mental Chatbot Research Work, 2023



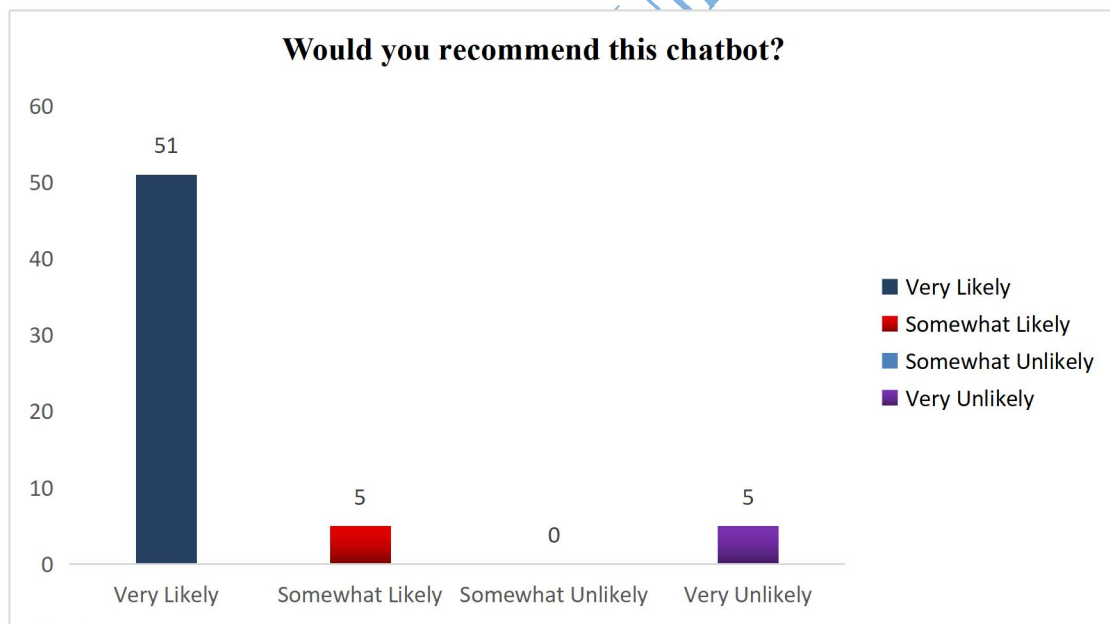
**Figure 4.17: Chatbot Timely Responses**  
Research Work, 2023



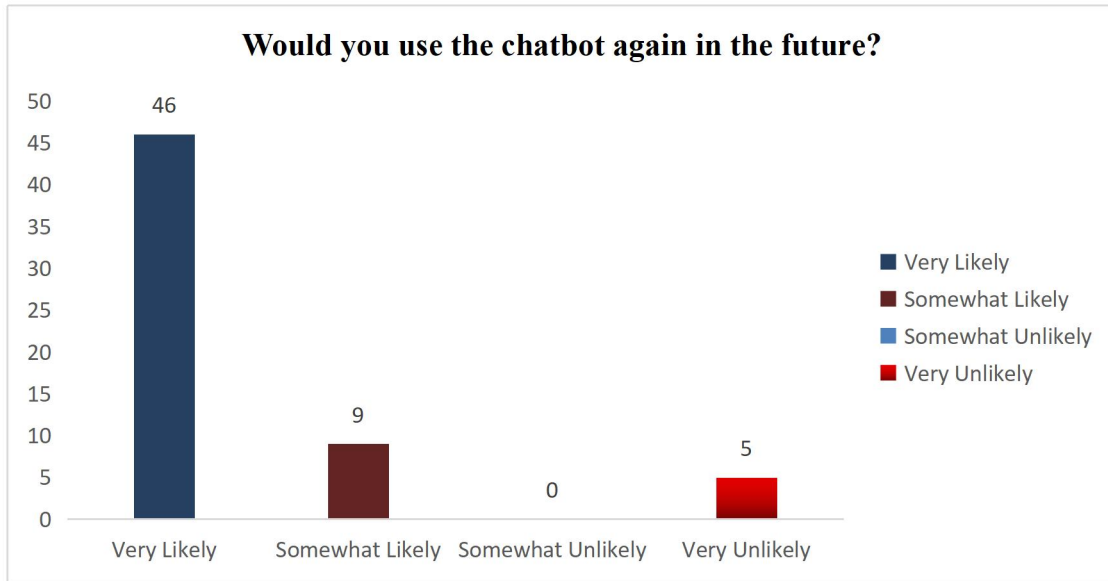
**Figure 4.18: Chatbot Understanding and Appropriate Response to Questions**  
Research Work, 2023



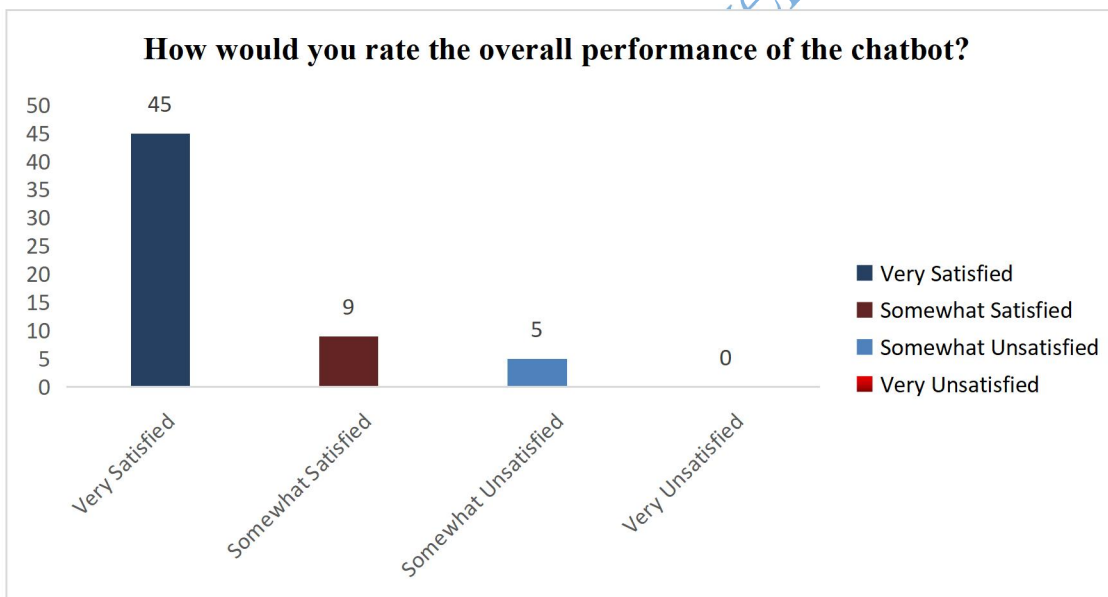
**Figure 4.19: Chatbot Therapy Recommender**  
Research Work, 2023



**Figure 4.20: User Chatbot Recommendation**  
Research Work, 2023



**Figure 4.21:** Chatbot Future Use  
Research Work, 2023



**Figure 4.22:** Chatbot Overall Performance  
Research Work, 2023

#### 4.4 Discussion of Findings

The study used an open source dataset which consists of FAQs about Mental Health. The dataset contains 98 records and 3 features (columns). Question, answer and question ID was extracted from the dataset, Word cloud was generated using Python's 'wordcloud' library. Each word was lemmatized and lowercased. Also,

duplicate words were removed, words and class sorted. The model is developed by the training of a machine learning model (neural network) and was trained to keep track of the training and validation loss.

The plot shows the loss value over the epochs from 0 to 200, the result shows that the higher epochs, the lower the loss. The findings indicate a negative correlation between the number of epochs and the magnitude of loss, suggesting that as the number of epochs increases, the loss decreases. The process of deployment encompasses a series of sequential processes that must be followed with precision.

Figures 4.9 and 4.15 shows the results and conversations between the chatbot (mental health counselor) and the user. The chatbot responded to different queries by the user. To assess the performance of the personalized Chatbot, a survey was conducted, and the results are presented in Table 4.1 and chart (figure 4.16-4.22). The majority of respondents reported being very satisfied with the Chatbot's performance in providing relevant personalized support for mental health queries, timely responses to their queries, understanding their questions and responding appropriately, and suggesting a suitable therapy for mental health enquiry. Additionally, the majority of respondents reported being very satisfied with the overall performance of the Chatbot and expressed a willingness to recommend and use the chatbot again in the future. These results show that most users were happy with the Chatbot's performance, which shows that the Chatbot met the study's goals by providing suggestions for personalized therapy system for mental health support.

## Endnotes

1. kaggle, <https://www.kaggle.com/datasets/narendrageek/mental-health-faq-for-chatbot>

*Do Not Copy, Lead City University, Nigeria*

## **Chapter Five**

### **Conclusion**

#### **5.1 Summary of Findings**

The main aim of this research was to develop a chatbot for personalized therapy system for mental health. The chatbot is built using model based on Long Short-Term Memory (LSTM) architecture, which holds significant implications for the field of mental health support systems. The primary objective of the creation of this model is to enhance the accessibility and accuracy of information and guidance related to mental health. In this study, the mental health counsellor bot was personalized, utilizing sequential models (Long Short-Term Memory (LSTM) architecture), thus generated replies. From the findings, chatbot provide relevant personalized support for mental health queries, provide timely responses to queries, understood questions and respond appropriately, recommended a suitable therapy for your mental health enquiry. The result also revealed that many bot users were pleased with the chatbot's overall performance and expressed a readiness to promote and use the personalized mental health bot again in the future. Furthermore, most users expressed high levels of satisfaction with the chatbot's performance, demonstrating that the chatbot personalized therapy system for mental health was achieved.

#### **5.2 Conclusion**

The research highlights the potential of chatbots, particularly those based on LSTM architecture, in offering personalized therapy and support for mental health. The positive user feedback and satisfaction demonstrate the value of such systems in addressing mental health concerns and enhancing accessibility to mental health resources.

In conclusion, the research has successfully achieved its primary objective of developing a chatbot for personalized therapy in the field of mental health. This chatbot, built on the Long Short-Term Memory (LSTM) architecture, holds significant implications for mental health support systems. The chatbot has been designed to enhance the accessibility and accuracy of information and guidance related to mental health. It provides timely responses and personalized support to users seeking assistance for mental health queries.

One of the notable features of the chatbot is its personalization. It tailors responses and therapy recommendations to individual users, ensuring that the support provided is relevant to their specific needs. The research has yielded positive results in terms of user satisfaction. It has the potential to fill gaps in mental health services by offering accessible and personalized assistance, which can be particularly valuable for individuals who may hesitate to seek help through traditional channels. While the research has demonstrated the effectiveness of the chatbot, it's important to acknowledge that the field of mental health chatbots is continuously evolving. Further development, user feedback incorporation, and ethical considerations are necessary to ensure the continued improvement and safety of these systems.

### **5.3 Recommendation**

Based on the research and the development of a personalized therapy chatbot for mental health, the following recommendations were made:

- i. Continuous Improvement refining and enhancing the chatbot's capabilities. Regularly update the chatbot's knowledge base, therapy recommendations, and conversational abilities to ensure it remains relevant and effective.

- ii. Ensure that the chatbot is accessible to a wide range of users, including those with disabilities. Consider incorporating features such as voice interfaces or text-to-speech capabilities to accommodate diverse user needs.
- iii. Explore opportunities to integrate the chatbot with existing mental health services and resources. This can create a seamless continuum of care for users who may require both automated support and human intervention.
- iv. It is recommended that expanding the chatbot's language capabilities should be considered to reach a broader audience. Multilingual support can make the chatbot accessible to users from diverse linguistic backgrounds.
- v. Prioritize user-centered design principles to ensure that the chatbot interface is intuitive and easy to navigate. User testing should be conducted and gather feedback to refine the user experience continually.
- vi. Prioritize accessibility in design to accommodate users with disabilities. Ensure that the chatbot is usable by individuals with diverse needs, including those who rely on screen readers or voice commands.
- vii. Ensure that the design incorporates robust privacy measures to safeguard user data and maintain confidentiality. Clearly communicate the chatbot's data usage policies and obtain user consent for data collection and storage.

#### **5.4 Contribution to Knowledge**

The development of a a web based chatbot for mental health support made significant contributions to knowledge in several key areas:

- i. The creation of an AI-powered chatbot based on Long Short-Term Memory (LSTM) architecture represents a technological advancement in mental health

support systems. It showcases the potential of AI and machine learning in providing accessible and personalized mental health assistance.

- ii. The research demonstrates the successful implementation of personalization strategies in mental health support. This contributes to the growing understanding of how AI can be leveraged to tailor support to individual user needs effectively.
- iii. The chatbot's development contributes to the expansion of accessibility to mental health resources. It addresses the challenge of reaching individuals who may have limited access to traditional mental health services due to geographical, financial, or stigma-related barriers.
- iv. The design considerations for the chatbot underscore the importance of human-computer interaction principles in creating a user-friendly and engaging interface. This knowledge can inform the development of future AI-driven mental health tools.
- v. The chatbot's interactions and user data provide valuable insights into mental health queries, preferences, and needs. This data can be used for further research and to inform the development of future mental health interventions.

### **5.5 Suggestions for Further Research**

The development of a personalized therapy chatbot for mental health opens up numerous avenues for further research and innovation in the field of mental health support and artificial intelligence. Here are some suggestions for future research:

- i. Explore advanced AI techniques, such as reinforcement learning, to further tailor therapy recommendations to individual user needs.
- ii. Develop specialized chatbots for specific mental health conditions (e.g., depression, anxiety, PTSD) to provide targeted support and interventions tailored to the unique needs of individuals with these conditions.
- iii. Investigate the adaptation and effectiveness of the chatbot in diverse cultural contexts and languages. Assess how cultural and linguistic factors impact the chatbot's ability to provide effective support.
- iv. Explore integration with wearable devices (e.g., fitness trackers, heart rate monitors) to collect physiological data that can inform the chatbot's therapy recommendations and assess the user's mental health state.
- v. Investigate AI ethics and potential biases in mental health chatbots. Develop frameworks and algorithms to mitigate biases and ensure that the chatbot's recommendations are fair and equitable.

## **5.6 Limitation of Research**

Every research project has its limitations, and it's essential to acknowledge these constraints to provide a clear understanding of the study's scope and potential areas for improvement. In the case of the research on the personalized therapy chatbot for mental health, below are some limitations:

Although the chatbot's effectiveness was assessed, the research may not have involved rigorous clinical validation, which typically requires extensive testing and validation with mental health professionals and patients. Also, the chatbot's ability to handle crisis situations or suicidality may be limited. It's crucial to acknowledge that the chatbot may not be a substitute for immediate human intervention in emergency

cases. The research may not have fully explored the design and usability of the human-machine interface, which can influence user engagement and satisfaction.

Further, the research might not have conducted a comprehensive analysis of potential biases in the chatbot's responses or recommendations and may not have implemented robust fairness measures. The chatbot's scalability to accommodate a growing user base and handle increased demand may not have been thoroughly examined.

*Do Not Copy, Lead City University, Nigeria*

## Bibliography

### International Conference

- Adamopoulou E, Moussiades L. *An overview of chatbot technology*. InIFIP International Conference on Artificial Intelligence Applications and Innovations 2020 Jun 5 (pp. 373-383). Springer, Cham
- Ahire S, Rohs M. *Tired of wake words? Moving towards seamless conversations with intelligent personal assistants*. InProceedings of the 2nd Conference on Conversational User Interfaces 2020 Jul 22 (pp. 1-3).
- Al-Madi NA, Maria KA, Al-Madi MA, Alia MA & Maria E.A. *An intelligent Arabic chatbot system proposed framework*. In 2021 International Conference on Information Technology (ICIT) , 2021, July (pp. 592-597). IEEE
- Alnefaie A, Singh S, Kocaballi B, Prasad M. *An overview of conversational agent: applications, challenges and future directions*. In17th International Conference on Web Information Systems and Technologies 2021 Jan 1. SCITEPRESS-Science and Technology Publications.
- Ayanouz S, Abdelhakim BA, Benhmed M. *A smart chatbot architecture based NLP and machine learning for health care assistance*. InProceedings of the 3rd international conference on networking, information systems & security 2020 Mar 31 (pp. 1-6).
- Ban, Q, He, H, Hu, R, Zhang, D.W & Zhang, S. *Analysis and optimization of factors influencing the academic development of left-behind children*. In 2022 8th International Conference on Humanities and Social Science Research (ICHSSR 2022) Atlantis Press, <https://doi.org/10.2991/assehr.k.220504.253>. 2022, 1395-1402
- Barber J, Yu X, Zamore LK, Lin J, Jazayeri V, Erlich S, Savor T, Stumm M. *Bladerunner: Stream processing at scale for a live view of backend data mutations at the edge*. InProceedings of the ACM SIGOPS 28th Symposium on Operating Systems Principles 2021 Oct 26 (pp. 708-723).
- Borah B, Pathak D, Sarmah P, Som B, Nandi S. *Survey of textbased chatbot in perspective of recent technologies*. InComputational Intelligence, Communications, and Business Analytics: Second International Conference, CICBA 2018, Kalyani, India, July 27–28, 2018, Revised Selected Papers, Part II 2 2019 (pp. 84-96). Springer Singapore.
- Díaz León JA, Olarte Escobar MA, Jara García M. *The prediction of hydrometeorology variables using the method of recurrent neuronal networks*. InInternational Conference on Water Resource and Environment 2022 Nov 1 (pp. 15-32). Singapore: Springer Nature Singapore.
- Fernandes S, Gawas R, Alvares P, Femandes M, Kale D, Aswale S. *Survey on various conversational systems*. In2020 International Conference on Emerging Trends

- in Information Technology and Engineering (ic-ETITE) 2020 Feb 24 (pp. 1-8). IEEE.
- Gupta R, Ashish S, Shekhar H, Dominic MD. *Music genre classification using CNN and RNN-LSTM*. In International Conference on Micro-Electronics and Telecommunication Engineering 2021 Sep 24 (pp. 729-745). Singapore: Springer Nature Singapore.
- Lee YC, Yamashita N, Huang Y. *Designing a chatbot as a mediator for promoting deep self-disclosure to a real mental health professional*. Proceedings of the ACM on Human-Computer Interaction. 2020 May 28;4(CSCW1):1-27.
- Miner A, Chow A, Adler S, Zaitsev I, Tero P, Darcy A, Paepcke A. *Conversational agents and mental health: Theory-informed assessment of language and affect*. In Proceedings of the fourth international conference on human agent interaction 2016 Oct 4 (pp. 123-130).
- Nallapati R, Zhai F, Zhou B. *Summarunner: A recurrent neural network based sequence model for extractive summarization of documents*. In Proceedings of the AAAI conference on artificial intelligence 2017 Feb 12 (Vol. 31, No. 1).
- Nemoto T, Fujimoto T. *A classification and analysis focusing on attempts to give a computer a personality: A technological history of chatbots as simple artificial intelligence*. In International Conference on Applied Informatics and Media Design 2019 Apr 24 (pp. 59-70). Cham: Springer International Publishing.
- Ongsulee P. *Artificial intelligence, machine learning and deep learning*. In 2017 15th international conference on ICT and knowledge engineering (ICT&KE) 2017 Nov 22 (pp. 1-6). IEEE.
- Pereira J, Díaz O. *A quality analysis of facebook messenger's most popular chatbots*. In Proceedings of the 33rd annual ACM symposium on applied computing 2018 Apr 9 (pp. 2144-2150).
- Pinto A, Gonçalo OH, Oliveira AA. *Comparing the performance of different NLP toolkits in formal and social media text*. In 5th Symposium on Languages, Applications and Technologies (SLATE'16) 2016. Schloss Dagstuhl-Leibniz-Zentrum fuer Informatik.
- Reiss F, Cutler B, Eichenberger Z. *Natural language processing with pandas dataframes*. In Proc. Of The 20th Python In Science Conf.(Scipy 2021) 2021 (pp. 49-58).
- Samir A, Lahbib Z. *Stemming and lemmatization for information retrieval systems in amazigh language*. In Big Data, Cloud and Applications: Third International Conference, BDCA 2018, Kenitra, Morocco, April 4–5, 2018, Revised Selected Papers 3 2018 (pp. 222-233). Springer International Publishing
- Siebra C, Correia W, Penha M, Macedo J, Quintino J, Anjos M, Florentin F, Da Silva FQ, Santos AL. *Virtual assistants for mobile interaction: A review from the*

- accessibility perspective*. In Proceedings of the 30th Australian Conference on Computer-Human Interaction 2018 Dec 4 (pp. 568-571).
- Singh H, Bhangare A, Singh R, Zope S, Saindane P. *Chatbots: A survey of the technology*. In International Conference on Intelligent Cyber Physical Systems and Internet of Things 2022 Aug 11 (pp. 671-691). Cham: Springer International Publishing.
- Singh S, Thakur HK. *Survey of various AI chatbots based on technology used*. In 2020 8th International Conference on Reliability, Infocom Technologies and Optimization (Trends and Future Directions)(ICRITO) 2020 Jun 4 (pp. 1074-1079). IEEE
- Truong A, Walters A, Goodsitt J, Hines K, Bruss CB, Farivar R. *Towards automated machine learning: Evaluation and comparison of AutoML approaches and tools*. In 2019 IEEE 31st international conference on tools with artificial intelligence (ICTAI) 2019 Nov 4 (pp. 1471-1479). IEEE.
- Vadhera A, Thute A, Mala S, Shankar A. *Chatbot on covid-19 for sustaining good health during the pandemic*. In Latest Trends in Renewable Energy Technologies: Select Proceedings of NCRESE 2020 2021 (pp. 271-284). Springer Singapore.
- Wang L, Wang D, Tian F, Peng Z, Fan X, Zhang Z, Yu M, Ma X, Wang H. *Cass: Towards building a social-support chatbot for online health community*. Proceedings of the ACM on Human-Computer Interaction. 2021 Apr 22;5(CSCW1):1-31
- World Health Organization. *The WHO special initiative for mental health (2019-2023): universal health coverage for mental health*. World Health Organization; 2019.
- World Health Organization. *World mental health report: transforming mental health for all: executive summary*. In World mental health report: transforming mental health for all: executive summary 2022.
- Zhang Y, Xu L, Mendoza A, Yang G, Chinprutthiwong P, Gu G. *Life after speech recognition: Fuzzing semantic misinterpretation for voice assistant applications*. In Proc. of the Network and Distributed System Security Symposium (NDSS'19) 2019 Jan.

## **Journals**

- Abd-Alrazaq AA, Alajlani M, Alalwan AA, Bewick BM, Gardner P, Househ M. *An overview of the features of chatbots in mental health: A scoping review*. **International Journal of Medical Informatics**. 2019 Dec 1;132:103978.
- Abdul-Kader SA, Woods JC. *Survey on chatbot design techniques in speech conversation systems*. **International Journal of Advanced Computer Science and Applications**. 2015;6(7).

- Ahirwar GK. *Chatterbot: technologies, tools and applications*. In *High Performance Vision Intelligence 2020* (pp. 203-213). Springer, Singapore.
- Aleem S, Huda NU, Amin R, Khalid S, Alshamrani SS, Alshehri A. *Machine learning algorithms for depression: diagnosis, insights, and research directions*. *Electronics*. 2022 Mar 31;11(7):1111.
- Anmella G, Sanabra M, Primé-Tous M, Segú X, Caverro M, Morilla I, Grande I, Ruiz V, Mas A, Martín-Villalba I, Caballo a. *Vickybot, a chatbot for anxiety-depressive symptoms and work-related burnout in primary care and health care professionals: Development, feasibility, and potential effectiveness studies*. *Journal of medical Internet research*. 2023 Apr 3;25:e43293.
- Arsovski S, Osipyany H, Oladele MI, Cheok AD. *Automatic knowledge extraction of any Chatbot from conversation*. *Expert Systems with Applications*. 2019 Dec 15;137:343-8
- Babenko V, Panchyshyn A, Zomchak L, Nehrey M, Artym-Drohomyretska Z, Lahotskyi T. *Classical machine learning methods in economics research: Macro and micro level example*. *WSEAS Transactions on Business and Economics*. 2021 Jan;18:209-17.
- Bengio Y, Lodi A, Prouvost A. *Machine learning for combinatorial optimization: a methodological tour d'horizon*. *European Journal of Operational Research*. 2021 Apr 16;290(2):405-21.
- Bisong E, Bisong E. *Recurrent Neural Networks (RNNs). Building machine learning and deep learning models on google cloud platform: A comprehensive guide for beginners*. 2019:443-73.
- Bokaba T, Doorsamy W, Paul BS. *Comparative study of machine learning classifiers for modelling road traffic accidents*. *Applied Sciences*. 2022 Jan 14;12(2):828.
- Bryndin E. *Creative communication safe ethical artificial intelligence in the era of technological development*. *Software Engineering*. 2020;8(3):13-23.
- Buckner C. *Deep learning: A philosophical introduction*. *Philosophy compass*. 2019 Oct;14(10):e12625.
- Burns E, Laskowski N, Tucci L. *What is artificial intelligence*. *Search Enterprise AI*. 2021 Nov.
- Callejas Z, Griol D. *Conversational agents for mental health and wellbeing. Dialog systems: a perspective from language, logic and computation*. 2021:219-44.
- Calvaresi D, Calbimonte JP, Siboni E, Eggenschwiler S, Manzo G, Hilfiker R, Schumacher M. *EREBOTS: Privacy-compliant agent-based platform for multi-scenario personalized health-assistant chatbots*. *Electronics*. 2021 Mar 12;10(6):666.

- Chen H, Liu X, Yin D, Tang J. *A survey on dialogue systems: Recent advances and new frontiers*. *Acm Sigkdd Explorations Newsletter*. 2017 Nov 21;19(2):25-35.
- Chinnamgari SK. *R Machine Learning Projects: Implement supervised, unsupervised, and reinforcement learning techniques using R 3.5*. Packt Publishing Ltd; 2019 Jan 14.
- Chung H, Park J, Lee S. *Digital forensic approaches for Amazon Alexa ecosystem*. *Digital investigation*. 2017 Aug 1;22:S15-25.
- Cristea D, Hofmann SG. *Why cognitive behavioral therapy is the current gold standard of psychotherapy*. *Frontiers in psychiatry*. 2018 Jan 29;9:4.
- De Angelis L, Baglivo F, Arzilli G, Privitera GP, Ferragina P, Tozzi AE, Rizzo C. *ChatGPT and the rise of large language models: the new AI-driven infodemic threat in public health*. *Frontiers in Public Health*. 2023 Apr 25;11:1166120.
- Denecke K, Vaaheesan S, Arulnathan A. *A mental health chatbot for regulating emotions (SERMO)-concept and usability test*. *IEEE Transactions on Emerging Topics in Computing*. 2020 Feb 17;9(3):1170-82.
- Eltay M, Zidouri A, Ahmad I. *Exploring deep learning approaches to recognize handwritten arabic texts*. *IEEE Access*. 2020 May 13;8:89882-98.
- Fitzpatrick KK, Darcy A, Vierhile M. *Delivering cognitive behavior therapy to young adults with symptoms of depression and anxiety using a fully automated conversational agent (Woebot): a randomized controlled trial*. **JMIR mental health**. 2017 Jun 6;4(2):e7785.
- Freeman M. *The World Mental Health Report: transforming mental health for all*. *World Psychiatry*. 2022 Oct;21(3):391.
- Fuad A, Yahya M. *Recent developments in Arabic conversational AI: a literature review*. *IEEE Access*. 2022 Feb 28;10:23842-59.
- George N, Muiz K, Whig P, Velu A. *Framework of perceptive artificial intelligence using natural language processing (PAIN)*. *Artificial & Computational Intelligence/Published Online: July*. 2021 Jul.
- Hahm Y, Kim J, An S, Lee M, Choi KS. *Chatbot who wants to learn the knowledge: KB-Agent*. *Semdeep/NLIWoD@ ISWC*. 2018 Oct 9;4.
- Harris CR, Millman KJ, Van Der Walt SJ, Gommers R, Virtanen P, Cournapeau D, Wieser E, Taylor J, Berg S, Smith NJ, Kern R. *Array programming with NumPy*. *Nature*. 2020 Sep 17;585(7825):357-62
- He H, Gray J, Cangelosi A, Meng Q, McGinnity TM, Mehnen J. *The challenges and opportunities of human-centered AI for trustworthy robots and autonomous systems*. *IEEE Transactions on Cognitive and Developmental Systems*. 2021 Dec 2;14(4):1398-412.

- Herda G, McNabb R. *Python for smarter cities: Comparison of python libraries for static and interactive visualisations of large vector data*. arXiv preprint arXiv:2202.13105. 2022 Feb 26
- Hihn H, Braun DA. *Specialization in hierarchical learning systems: A unified information-theoretic approach for supervised, unsupervised and reinforcement learning*. *Neural Processing Letters*. 2020 Dec;52(3):2319-52.
- Hiran KK, Jain RK, Lakhwani K, Doshi R. *Machine Learning: Master supervised and unsupervised learning algorithms with real examples (English Edition)*. BPB Publications; 2021 Sep 16.
- Hügler M, Omoumi P, van Laar JM, Boedecker J, Hügler T. *Applied machine learning and artificial intelligence in rheumatology*. *Rheumatology advances in practice*. 2020;4(1)
- Hungerbuehler I, Daley K, Cavanagh K, Garcia Claro H, Kapps M. *Chatbot-based assessment of employees' mental health: Design process and pilot implementation*. **JMIR Formative Research**. 2021 Apr 21;5(4):e21678.
- Hwang GJ, Xie H, Wah BW, Gašević D. *Vision, challenges, roles and research issues of artificial intelligence in education*. *Computers and Education: Artificial Intelligence*. 2020 Jan 1;1:100001.
- Iasiello M, van Agteren J, Schotanus-Dijkstra M, Lo L, Fassnacht DB, Westerhof GJ. *Assessing mental wellbeing using the Mental Health Continuum—Short Form: A systematic review and meta-analytic structural equation modelling*. *Clinical Psychology: Science and Practice*. 2022 Mar 21.
- Ilievski V. *Building advanced dialogue managers for goal-oriented dialogue systems*. arXiv preprint arXiv:1806.00780. 2018 Jun 3.
- Inkster B, Sarda S, Subramanian V. *An empathy-driven, conversational artificial intelligence agent (Wysa) for digital mental well-being: real-world data evaluation mixed-methods study*. **JMIR mHealth and uHealth**. 2018 Nov 23;6(11):e12106.
- Kamita T, Ito T, Matsumoto A, Munakata T, Inoue T. *A chatbot system for mental healthcare based on SAT counseling method*. *Mobile Information Systems*. 2019 Mar 3;2019.
- Khanna A, Pandey B, Vashishta K, Kalia K, Pradeepkumar B, Das T. *A study of today's AI through chatbots and rediscovery of machine intelligence*. *International Journal of u-and e-Service, Science and Technology*. 2015 Jul 31;8(7):277-84.
- Kretschmar K, Tyroll H, Pavarini G, Manzini A, Singh I, NeurOx *Young People's Advisory Group*. *Can your phone be your therapist? Young people's ethical perspectives on the use of fully automated conversational agents (chatbots) in*

- mental health support*. Biomedical informatics insights. 2019 Feb;11:1178222619829083.
- Kudo T, Richardson J. *Sentencepiece: A simple and language independent subword tokenizer and detokenizer for neural text processing*. arXiv preprint arXiv:1808.06226. 2018 Aug 19.
- Kunal B. *Conversational AI chatbot using deep learning: How bi-directional lstm, machine reading comprehension, transfer learning, sequence to sequence model with multi-headed attention mechanism, generative adversarial network, self learning based sentiment analysis*, 2019
- Lee J, Kim J, Jo M, Lee H, Kim J, Lee S. *Designing an empathetic chatbot for mental health counseling using machine learning*. **Journal of medical systems**. 2020;44(9):1-10.
- Lee J, Kim J, Kim J-H. *Development of a machine learning-based chatbot for mental health support*. **Journal of medical Internet research**. 2020;22(10):e21580
- Lee J, Kim J, Lee J, Kim J, Kim H, Kim Y. *Designing a machine learning-based chatbot for mental health support*. Healthcare informatics research. 2020;26(4):290-8.
- Lee J, Lee Y, Song J, Kim J, Kim J. *Designing a machine learning-based chatbot for mental health support*. **Journal of medical systems**. 2020;44(7):1-10.
- Lomas T, VanderWeele TJ. *The mental illness-health matrix and the mental state space matrix: Complementary meta-conceptual frameworks for evaluating psychological states*. **Journal of Clinical Psychology**. 2023 Mar 14.
- Ly KH, Ly A-M, Andersson G. *A randomized controlled trial of a chatbot-based intervention for depression*. Frontiers in psychology. 2020;11:491.
- Mattas PS. *ChatGPT: A study of AI language processing and its implications*. Journal homepage: www. ijrpr. com ISSN. 2023;2582:7421
- Mittal M, Battineni G, Singh D, Nagarwal T, Yadav P. *Web-based chatbot for frequently asked queries (FAQ) in hospitals*. **Journal of Taibah University Medical Sciences**. 2021 Oct 1;16(5):740-6.
- Mollajan A, Iranmanesh H, Khezri A, Abazari A. *Effect of applying independence axiom of Axiomatic Design theory on performance of an Integrated Manufacturing Information System: a computer simulation modeling approach*. *Simulation*. 2022 Jul;98(7):535-61
- Moolayil J, Moolayil J. *An introduction to deep learning and keras. Learn keras for deep neural networks: A fast-track approach to modern deep learning with python*. 2019:1-6.

- Natale S. *If software is narrative: Joseph Weizenbaum, artificial intelligence and the biographies of ELIZA*. *new media & society*. 2019 Mar;21(3):712-28.
- Noble JM, Zamani A, Gharaat M, Merrick D, Maeda N, Foster AL, Nikolaidis I, Goud R, Stroulia E, Agyapong VI, Greenshaw AJ. *Developing, implementing, and evaluating an artificial intelligence-guided mental health resource navigation chatbot for health care workers and their families during and following the COVID-19 pandemic: protocol for a cross-sectional study*. **JMIR Research Protocols**. 2022 Jul 25;11(7):e33717.
- Oud M, De Winter L, Vermeulen-Smit E, Bodden D, Nauta M, Stone L, Den Heuvel M, Al Taher R, De Graaf I, Kendall T, Engels R. *Effectiveness of CBT for children and adolescents with depression: A systematic review and meta-regression analysis*. *European psychiatry*. 2019 Apr;57:33-45.
- Paczkowski WR. *Data visualization: The basics. Business analytics: Data science for business problems*. 2021:85-126.
- Park S, Choi J, Lee S, Oh C, Kim C, La S, Lee J, Suh B. *Designing a chatbot for a brief motivational interview on stress management: Qualitative case study*. **Journal of medical Internet research**. 2019 Apr 16;21(4):e12231.
- Peng B, Li X, Li L, Gao J, Celikyilmaz A, Lee S, Wong KF. *Composite task-completion dialogue policy learning via hierarchical deep reinforcement learning*. arXiv preprint arXiv:1704.03084. 2017 Apr 10.
- Preethi, S. *A survey on artificial intelligence*. **International Journal of Intelligent Computing and Technology (IJICT)**, 3(2), pp.39-42. 2020
- Przegalinska A, Ciechanowski L, Stroz A, Gloor P, Mazurek G. *In bot we trust: A new methodology of chatbot performance measures*. *Business Horizons*. 2019 Nov 1;62(6):785-97.
- Qader R, Portet F, Labbé C. *Semi-supervised neural text generation by joint learning of natural language generation and natural language understanding models*. arXiv preprint arXiv:1910.03484. 2019 Sep 29.
- Rathnayaka P, Mills N, Durnett B, De Silva D, Alahakoon D, Gray R. *A mental health chatbot with cognitive skills for personalised behavioural activation and remote health monitoring*. *Sensors*. 2022 May 11;22(10):3653.
- Regier AD, Kuhl EA & Kupfer DJ. *The DSM-5: Classification and criteria changes*. *World psychiatry*, 2013;12(2)
- Sabour S, Zhang W, Xiao X, Zhang Y, Zheng Y, Wen J, Zhao J, Huang M. *A chatbot for mental health support: exploring the impact of Emohaa on reducing mental distress in China*. *Frontiers in Digital Health*. 2023 May 4;5:1133987.
- Sarker IH. *Machine learning: Algorithms, real-world applications and research directions*. *SN computer science*. 2021 May;2(3):160.

- Schanke S, Burtch G, Ray G. *Estimating the impact of “humanizing” customer service chatbots*. *Information Systems Research*. 2021 Sep;32(3):736-51.
- Sennrich R, Haddow B, Birch A. *Neural machine translation of rare words with subword units*. arXiv preprint arXiv:1508.07909. 2015 Aug 31.
- Shaikh SJ. *Artificially intelligent, interactive, and assistive machines: A definitional framework for intelligent assistants*. **International Journal of Human-Computer Interaction**. 2023 Feb 25;39(4):776-89.
- Sidey-Gibbons JA, Sidey-Gibbons CJ. *Machine learning in medicine: a practical introduction*. *BMC medical research methodology*. 2019 Dec;19:1-8.
- Skjuve M, BrandtzægPB. *Chatbots as a new user interface for providing health information to young people. Youth and news in a digital media environment—Nordic-Baltic perspectives*. 2018.
- Smith A, Chen J, Salazar LF, Horvitz E. *Designing a chatbot for mental health support using machine learning: A user-centered approach*. **JMIR mental health**. 2020;7(9):e21457.
- Sweeney C, Potts C, Ennis E, Bond R, Mulvenna MD, O’neill S, Malcolm M, Kuosmanen L, Kostenius C, Vakaloudis A, Mcconvey G. *Can chatbots help support a person’s mental health? Perceptions and views from mental healthcare professionals and experts*. *ACM Transactions on Computing for Healthcare*. 2021 Jul 15;2(3):1-5.
- Topol EJ. *High-performance medicine: the convergence of human and artificial intelligence*. *Nature medicine*. 2019 Jan;25(1):44-56.
- Toprak A, Turan M, Toprak FS. *Deep learning based chatbot adapted to the electronic funds transfer process of Turkish banking*. **International Journal of Intelligent Systems and Applications in Engineering**. 2023 Feb 4;11(1):348-53.
- Toraman C, Yilmaz EH, Şahinuç F, Ozelik O. *Impact of tokenization on language models: An analysis for turkish*. *ACM Transactions on Asian and Low-Resource Language Information Processing*. 2023 Mar 25;22(4):1-21.
- Vaidyam AN, Wisniewski H, Halamka JD, Kashavan MS, Torous JB. *Chatbots and conversational agents in mental health: a review of the psychiatric landscape*. **The Canadian Journal of Psychiatry**. 2019 Jul;64(7):456-64..
- Wang W, Wang W. *Talking to cortana. Absolute beginners guide to computing*. 2016:257-82.
- Wang Y, Zhang Y, Huang L, Liu Y, Liu Y, Cui C. *Designing an artificial intelligence-based chatbot for mental health support*. **International Journal of Medical Informatics**. 2021;149:104441.

- Wei X, Zhang L, Yang HQ, Zhang L, Yao YP. *Machine learning for pore-water pressure time-series prediction: Application of recurrent neural networks*. Geoscience Frontiers. 2021 Jan 1;12(1):453-67.
- Weißensteiner AA. *Chatbots as an approach for a faster enquiry handling process in the service industry*. Signature. 2018;12(04)
- Wen TH, Gasic M, Mrksic N, Su PH, Vandyke D, Young S. *Semantically conditioned lstm-based natural language generation for spoken dialogue systems*. arXiv preprint arXiv:1508.01745. 2015 Aug 7.
- Wiggins, Kitamura F, Santos I, Prevedello LM. *Natural language processing of radiology text reports: interactive text classification*. Radiology: Artificial Intelligence. 2021 May 12;3(4):e210035.
- Xie B, He D, Mercer T, Wang Y. *Chatbots in mental health: A review of the recent progress*. IEEE Transactions on Neural Systems and Rehabilitation Engineering. 2021;29:1-1.
- Xie B, Su Z, Zhang W, Cui L. *Chatbots in mental health: A review of the recent progress*. **Journal of psychiatric research**. 2021;145:24-33.
- Yu Y, Si X, Hu C, Zhang J. *A review of recurrent neural networks: LSTM cells and network architectures*. Neural computation. 2019 Jul 1;31(7):1235-70.
- Zhao Z, Srivastava A, Peng L, Chen Q. *Long short-term memory network design for analog computing*. **ACM Journal on Emerging Technologies in Computing Systems (JETC)**. 2019 Jan 9;15(1):1-27.

### Thesis

- Andeta J.A. *Road-traffic accident prediction model: Predicting the number of casualties* [Internet] [Dissertation]. 2021. Available from: <http://urn.kb.se/resolve?urn=urn:nbn:se:his:diva-20146>
- Kockx AJ. *Development and Evaluation of a Diagnosis and Triage Healthcare Chatbot* (Master's thesis).2021
- Kostelník P, PISAŘOVIC I, MURŮN M, DAŘENA F, PROCHÁZKA D. *Chatbots for enterprises: outlook*. Acta Universitatis Agriculturae et Silviculturae Mendelianae Brunensis. 2019.
- X Huang, A CIS. *Chatbot: Design, architecture, and applications*. ASCS CIS 498 Senior Capstone Thesis University of Pennsylvania, School of Engineering and Applied Science May 3, 2021

### Website

<http://colah.github.io/posts/2015-09-NN-Types-FP/>

[https://en.wikipedia.org/wiki/ELIZA#/media/File:ELIZA\\_conversation.png](https://en.wikipedia.org/wiki/ELIZA#/media/File:ELIZA_conversation.png)

<https://onlim.com/en/the-history-of-chatbots/>

<https://www.microsoft.com/en-us/research/project/deep-reinforcement-learning-goal-oriented-dialogue/>

[https://www.tutorialspoint.com/tensorflow/tensorflow\\_multi\\_layer\\_perceptron\\_learning.html](https://www.tutorialspoint.com/tensorflow/tensorflow_multi_layer_perceptron_learning.html)

kaggle, <https://www.kaggle.com/datasets/narendrageek/mental-health-faq-for-chatbot>

*Do Not Copy, Lead City University, Nigeria*

## Appendices

### Appendix I: Source Code

```
In [ ]:
# importing necessary libraries import nltk from nltk.stem import
WordNetLemmatizer import json import pickle import numpy as np from
keras.models import Sequential from keras.layers import Dense, Activation,
Dropout from keras.optimizers import SGD import random import pandas as pd
In [ ]:
# Load dataset from CSV file dataset = pd.read_csv('Mental_Health_FAQ.csv')
In [ ]:
dataset.head()
Out [ ]:
In [ ]:
# Extract questions, answers, and question IDs from the dataset questions =
dataset['Questions'].tolist() answers = dataset['Answers'].tolist() question_ids =
dataset['Question_ID'].tolist()
In [ ]:
import matplotlib.pyplot as plt from wordcloud import WordCloud
# Assuming 'questions' is a list of strings containing the questions in your dataset text
= ''.join(questions)
# Generate word cloud wordcloud = WordCloud(width=800,
height=400).generate(text)
# Display the generated word cloud using matplotlib plt.figure(figsize=(10,
5)) plt.imshow(wordcloud,
interpolation='bilinear') plt.axis('off') plt.savefig('wordcloud.png') plt.show()

In [ ]:
# Perform processing similar to the original code documents = [] words = [] classes =
[] ignore_words = ['?', '!']
for i in range(len(questions)):
    question = questions[i]
    answer = answers[i]
    question_id = question_ids[i]

    # Tokenize each word
    w = nltk.word_tokenize(question)
    words.extend(w)
    # Add documents to the corpus
    documents.append((w, question_id))

    # Add to the classes list
    if question_id not in classes:
        classes.append(question_id)
        lemmatizer = WordNetLemmatizer() # Lemmatize and lowercase each word and
remove duplicates words = [lemmatizer.lemmatize(w.lower()) for w in words if w not
in ignore_words] words = sorted(list(set(words)))
```

```

# Sort classes
classes = sorted(list(set(classes)))
# Print information about the
documents, classes, and words
print(len(documents), "documents")
print(len(classes),
"classes", classes)
print(len(words), "unique lemmatized words", words)
98 documents
98 classes [1030153, 1043721, 1155199, 1259439, 1337085, 1511075, 1546812,
1585622, 1590140, 1619387, 1667863, 1706961, 1833460, 1896541, 1898078,
2009922, 2110618, 2115228, 2176317, 2447683, 2553795, 2554141, 2612846,
2746837, 2884887, 2903196, 2973656, 3055896, 3166337, 3221856, 3268009,
3284724, 3340726, 3388962, 3393534, 3597720, 3839472, 4031860, 4134858,
4181750, 4194958, 4197817, 4200873, 4211025, 4283807, 4294616, 4352464,
4373204, 4759773, 4816493, 4824231, 4893447, 4925221, 4962901, 5215843,
5343181, 5640861, 5778437, 5819325, 5981663, 6062232, 6085633, 6167248,
6361820, 6521784, 6623577, 6779222, 6851366, 6869967, 6981545, 7009409,
7069853, 7535002, 7657263, 7728364, 7760466, 7807643, 7984793, 7995219,
8022026, 8471978, 8612349, 8690253, 8904276, 8927672, 9020440, 9045344,
9049618, 9100298, 9167296, 9178453, 9250044, 9434130, 9539480, 9541219,
9630578, 9676742, 9679704]
259 unique lemmatized words [""', "'m", "'s", '(', ')', ',', '!', '19+', '25', "'", 'a', 'about',
'addiction', 'adhd', 'adult', 'advance', 'affect', 'after', 'age', 'alcohol', 'allowed', 'am', 'an',
'and', 'antidepressant', 'antisocial', 'anxiety', 'any', 'appears', 'apply', 'are', 'assistance',
'available', 'bad', 'balanced', 'become', 'been', 'before', 'better', 'between', 'binge-eating',
'borderline', 'brain', 'but', 'can', 'cannabis', 'care', 'cause', 'cbd', 'cbt', 'certain', 'challenge',
'child', 'clinical', 'concern', 'connection', 'convince', 'coping', 'counselling', 'counsellor',
'creating', 'cure', 'cured', 'cycling', 'cyclothymic', 'dangerous', 'dbt', 'deal', 'depression',
'depressive', 'diagnosed', 'diagnosis', 'difference', 'different', 'difficult', 'directive',
'disorder', 'dissociative', 'distraction', 'do', 'doctor', 'doe', 'doesn', 'don', 'drinking', 'drug',
'dysthymia', 'else', 'evidence', 'fact', 'feel', 'feeling', 'find', 'for', 'free', 'friend', 'get', 'go',
'good', 'grief', 'group', 'ha', 'hard', 'harmful', 'have', 'having', 'health', 'help', 'how', 'i',
'identity', 'if', 'ill', 'illness', 'in', 'income', 'information', 'informed', 'injury', 'inpatient',
'involved', 'is', 'it', 'keep', 'kid', 'know', 'learn', 'legal', 'legalizing', 'legally', 'life', 'local',
'lonely', 'love', 'low-cost', 'm', 'maintain', 'manage', 'material', 'me', 'mean', 'medication',
'mental', 'might', 'more', 'msp', 'much', 'multiple', 'my', 'myself', 'myth', 'need',
'negative', 'new', 'next', 'not', 'obsessive-compulsive', 'of', 'oil', 'older', 'on', 'one',
'option', 'or', 'other', 'parent', 'pay', 'paying', 'people', 'persistent', 'person', 'personality',
'perspective', 'physical', 'plan', 'positive', 'potentially', 'prescribed', 'prescription',
'prevent', 'problem', 'prodrome', 'professional', 'psychiatric', 'psychiatrist',
'psychologist', 'psychosis', 'psychotherapy', 'rapid', 'received', 'recover', 'referral',
'reframe', 'registered', 'relative', 'right', 's', 'sadness', 'saying', 'schizoid', 'schizophrenia',
'see', 'seems', 'self-help', 'service', 'should', 'sign', 'situation', 'social', 'some', 'someone',
'starting', 'status', 'stop', 'stress', 'substance', 'suicide', 'support', 'symptom', 't', 'take',
'taking', 'talking', 'teenager', 'tell', 'telling', 'the', 'therapy', 'there', 'think', 'thinking', 'this',
'thought', 'time', 'to', 'too', 'trap', 'treatment', 'trial', 'type', 'unwell', 'use', 'vaping', 'wa',
'warning', 'we', 'what', 'when', 'where', 'who', 'why', 'with', 'won', 'working', 'worried',
'you', 'young', 'youth', '"]
In [ ]:
from collections import Counter
# Assuming 'questions' is a list of question texts in your dataset
word_counts =
Counter(for question in questions:
    words = question.split()
    word_counts.update(words)

```

```

# Select the top N most common words
top_n = 20
common_words = dict(word_counts.most_common(top_n))
# Plotting the bar chart
plt.figure(figsize=(10, 5))
plt.bar(common_words.keys(), common_words.values())
plt.xlabel('Words')
plt.ylabel('Frequency')
plt.title(f'Top {top_n} Most Common Words')
plt.xticks(rotation=45)
plt.savefig('commonwords.png')
plt.show()

In [ ]:
# Assuming 'questions' is a list of question texts in your dataset
word_lengths = [len(word) for question in questions for word in question.split()]
# Plotting the histogram
plt.figure(figsize=(10, 5))
plt.hist(word_lengths, bins=20)
plt.xlabel('Word Length')
plt.ylabel('Frequency')
plt.title('Distribution of Word Lengths')
plt.savefig('wordlength.png')
plt.show()

In [ ]:
# Create our training data
training = []
output_empty = [0] * len(classes)
for doc in documents:
    bag = []
    pattern_words = doc[0]
    pattern_words = [lemmatizer.lemmatize(word.lower()) for word in pattern_words]

    for w in words:
        bag.append(1) if w in pattern_words else bag.append(0)

    output_row = list(output_empty)
    output_row[classes.index(doc[1])] = 1

    training.append([bag, output_row])

In [ ]:
# Shuffle and convert to numpy array
random.shuffle(training)
training = np.array(training, dtype=object)

In [ ]:
# Create train and test list
train_x = list(training[:, 0])
train_y = list(training[:, 1])
print("Training data created")
Training data created

In [ ]:
# Create model
model = Sequential()
model.add(Dense(128, input_shape=(len(train_x[0]),), activation='relu'))
model.add(Dropout(0.5))
model.add(Dense(64, activation='relu'))
model.add(Dropout(0.5))
model.add(Dense(len(train_y[0]), activation='softmax'))

In [ ]:
# Compile model
sgd = SGD(lr=0.01, decay=1e-6, momentum=0.9, nesterov=True)
model.compile(loss='categorical_crossentropy', optimizer=sgd, metrics=['accuracy'])
# Fit the model
hist = model.fit(np.array(train_x), np.array(train_y), epochs=200, batch_size=5, verbose=1)
Epoch 1/200
20/20 [=====] - 1s 3ms/step - loss: 4.5889 - accuracy: 0.0000e+00

```

Epoch 2/200  
20/20 [=====] - 0s 3ms/step - loss: 4.5766 -  
accuracy: 0.0000e+00

Epoch 3/200  
20/20 [=====] - 0s 3ms/step - loss: 4.5701 -  
accuracy: 0.0102

Epoch 4/200  
20/20 [=====] - 0s 4ms/step - loss: 4.5697 -  
accuracy: 0.0000e+00

Epoch 5/200  
20/20 [=====] - 0s 3ms/step - loss: 4.5687 -  
accuracy: 0.0000e+00

Epoch 6/200  
20/20 [=====] - 0s 3ms/step - loss: 4.5681 -  
accuracy: 0.0204

Epoch 7/200  
20/20 [=====] - 0s 3ms/step - loss: 4.5476 -  
accuracy: 0.0204

Epoch 8/200  
20/20 [=====] - 0s 3ms/step - loss: 4.5483 -  
accuracy: 0.0102

Epoch 9/200  
20/20 [=====] - 0s 3ms/step - loss: 4.5135 -  
accuracy: 0.0204

Epoch 10/200  
20/20 [=====] - 0s 3ms/step - loss: 4.5044 -  
accuracy: 0.0000e+00

Epoch 11/200  
20/20 [=====] - 0s 3ms/step - loss: 4.4596 -  
accuracy: 0.0408

Epoch 12/200  
20/20 [=====] - 0s 3ms/step - loss: 4.4595 -  
accuracy: 0.0204

Epoch 13/200  
20/20 [=====] - 0s 4ms/step - loss: 4.4743 -  
accuracy: 0.0102

Epoch 14/200  
20/20 [=====] - 0s 4ms/step - loss: 4.4436 -  
accuracy: 0.0306

Epoch 15/200  
20/20 [=====] - 0s 4ms/step - loss: 4.3905 -  
accuracy: 0.0306

Epoch 16/200  
20/20 [=====] - 0s 4ms/step - loss: 4.3918 -  
accuracy: 0.0102

Epoch 17/200  
20/20 [=====] - 0s 4ms/step - loss: 4.3107 -  
accuracy: 0.0306

Epoch 18/200

20/20 [=====] - 0s 4ms/step - loss: 4.3212 -  
accuracy: 0.0102  
Epoch 19/200  
20/20 [=====] - 0s 4ms/step - loss: 4.3153 -  
accuracy: 0.0102  
Epoch 20/200  
20/20 [=====] - 0s 4ms/step - loss: 4.2590 -  
accuracy: 0.0204  
Epoch 21/200  
20/20 [=====] - 0s 4ms/step - loss: 4.2488 -  
accuracy: 0.0000e+00  
Epoch 22/200  
20/20 [=====] - 0s 4ms/step - loss: 4.2416 -  
accuracy: 0.0204  
Epoch 23/200  
20/20 [=====] - 0s 3ms/step - loss: 4.1949 -  
accuracy: 0.0306  
Epoch 24/200  
20/20 [=====] - 0s 3ms/step - loss: 4.1505 -  
accuracy: 0.0000e+00  
Epoch 25/200  
20/20 [=====] - 0s 3ms/step - loss: 4.1558 -  
accuracy: 0.0306  
Epoch 26/200  
20/20 [=====] - 0s 3ms/step - loss: 4.0721 -  
accuracy: 0.0510  
Epoch 27/200  
20/20 [=====] - 0s 3ms/step - loss: 4.1188 -  
accuracy: 0.0204  
Epoch 28/200  
20/20 [=====] - 0s 3ms/step - loss: 4.1061 -  
accuracy: 0.0306  
Epoch 29/200  
20/20 [=====] - 0s 3ms/step - loss: 4.1676 -  
accuracy: 0.0102  
Epoch 30/200  
20/20 [=====] - 0s 4ms/step - loss: 4.0931 -  
accuracy: 0.0204  
Epoch 31/200  
20/20 [=====] - 0s 4ms/step - loss: 4.1311 -  
accuracy: 0.0204  
Epoch 32/200  
20/20 [=====] - 0s 4ms/step - loss: 4.1196 -  
accuracy: 0.0102  
Epoch 33/200  
20/20 [=====] - 0s 4ms/step - loss: 4.0783 -  
accuracy: 0.0510  
Epoch 34/200  
20/20 [=====] - 0s 4ms/step - loss: 4.0894 -  
accuracy: 0.0204

Epoch 35/200  
20/20 [=====] - 0s 4ms/step - loss: 4.0250 -  
accuracy: 0.0408  
Epoch 36/200  
20/20 [=====] - 0s 4ms/step - loss: 4.0385 -  
accuracy: 0.0102  
Epoch 37/200  
20/20 [=====] - 0s 4ms/step - loss: 4.0669 -  
accuracy: 0.0408  
Epoch 38/200  
20/20 [=====] - 0s 4ms/step - loss: 4.0935 -  
accuracy: 0.0306  
Epoch 39/200  
20/20 [=====] - 0s 4ms/step - loss: 4.0404 -  
accuracy: 0.0204  
Epoch 40/200  
20/20 [=====] - 0s 3ms/step - loss: 3.9919 -  
accuracy: 0.0408  
Epoch 41/200  
20/20 [=====] - 0s 3ms/step - loss: 3.9924 -  
accuracy: 0.0510  
Epoch 42/200  
20/20 [=====] - 0s 3ms/step - loss: 3.9382 -  
accuracy: 0.0306  
Epoch 43/200  
20/20 [=====] - 0s 4ms/step - loss: 4.0392 -  
accuracy: 0.0204  
Epoch 44/200  
20/20 [=====] - 0s 4ms/step - loss: 4.0221 -  
accuracy: 0.0102  
Epoch 45/200  
20/20 [=====] - 0s 4ms/step - loss: 3.9704 -  
accuracy: 0.0306  
Epoch 46/200  
20/20 [=====] - 0s 4ms/step - loss: 3.9456 -  
accuracy: 0.0306  
Epoch 47/200  
20/20 [=====] - 0s 3ms/step - loss: 3.9873 -  
accuracy: 0.0102  
Epoch 48/200  
20/20 [=====] - 0s 4ms/step - loss: 3.9148 -  
accuracy: 0.0408  
Epoch 49/200  
20/20 [=====] - 0s 3ms/step - loss: 4.0019 -  
accuracy: 0.0204  
Epoch 50/200  
20/20 [=====] - 0s 3ms/step - loss: 3.9229 -  
accuracy: 0.0204  
Epoch 51/200

20/20 [=====] - 0s 3ms/step - loss: 3.9087 -  
accuracy: 0.0102  
Epoch 52/200  
20/20 [=====] - 0s 3ms/step - loss: 3.9214 -  
accuracy: 0.0102  
Epoch 53/200  
20/20 [=====] - 0s 4ms/step - loss: 3.9605 -  
accuracy: 0.0408  
Epoch 54/200  
20/20 [=====] - 0s 4ms/step - loss: 3.9455 -  
accuracy: 0.0408  
Epoch 55/200  
20/20 [=====] - 0s 4ms/step - loss: 3.9087 -  
accuracy: 0.0408  
Epoch 56/200  
20/20 [=====] - 0s 4ms/step - loss: 3.9190 -  
accuracy: 0.0204  
Epoch 57/200  
20/20 [=====] - 0s 4ms/step - loss: 3.9143 -  
accuracy: 0.0204  
Epoch 58/200  
20/20 [=====] - 0s 4ms/step - loss: 3.8670 -  
accuracy: 0.0408  
Epoch 59/200  
20/20 [=====] - 0s 4ms/step - loss: 3.9148 -  
accuracy: 0.0510  
Epoch 60/200  
20/20 [=====] - 0s 4ms/step - loss: 3.9212 -  
accuracy: 0.0204  
Epoch 61/200  
20/20 [=====] - 0s 4ms/step - loss: 3.8960 -  
accuracy: 0.0510  
Epoch 62/200  
20/20 [=====] - 0s 4ms/step - loss: 3.9007 -  
accuracy: 0.0102  
Epoch 63/200  
20/20 [=====] - 0s 4ms/step - loss: 3.9521 -  
accuracy: 0.0306  
Epoch 64/200  
20/20 [=====] - 0s 4ms/step - loss: 3.8080 -  
accuracy: 0.0510  
Epoch 65/200  
20/20 [=====] - 0s 7ms/step - loss: 3.8820 -  
accuracy: 0.0102  
Epoch 66/200  
20/20 [=====] - 0s 6ms/step - loss: 3.7937 -  
accuracy: 0.0306  
Epoch 67/200  
20/20 [=====] - 0s 4ms/step - loss: 3.8760 -  
accuracy: 0.0612

Epoch 68/200  
20/20 [=====] - 0s 4ms/step - loss: 3.9046 -  
accuracy: 0.0510  
Epoch 69/200  
20/20 [=====] - 0s 4ms/step - loss: 3.7741 -  
accuracy: 0.0612  
Epoch 70/200  
20/20 [=====] - 0s 4ms/step - loss: 3.8099 -  
accuracy: 0.0612  
Epoch 71/200  
20/20 [=====] - 0s 4ms/step - loss: 3.8231 -  
accuracy: 0.0510  
Epoch 72/200  
20/20 [=====] - 0s 4ms/step - loss: 3.8751 -  
accuracy: 0.0510  
Epoch 73/200  
20/20 [=====] - 0s 4ms/step - loss: 3.8295 -  
accuracy: 0.0408  
Epoch 74/200  
20/20 [=====] - 0s 4ms/step - loss: 3.8446 -  
accuracy: 0.0510  
Epoch 75/200  
20/20 [=====] - 0s 4ms/step - loss: 3.8989 -  
accuracy: 0.0306  
Epoch 76/200  
20/20 [=====] - 0s 4ms/step - loss: 3.9189 -  
accuracy: 0.0306  
Epoch 77/200  
20/20 [=====] - 0s 3ms/step - loss: 3.8629 -  
accuracy: 0.0408  
Epoch 78/200  
20/20 [=====] - 0s 3ms/step - loss: 3.8895 -  
accuracy: 0.0204  
Epoch 79/200  
20/20 [=====] - 0s 4ms/step - loss: 3.8285 -  
accuracy: 0.0612  
Epoch 80/200  
20/20 [=====] - 0s 3ms/step - loss: 3.8814 -  
accuracy: 0.0204  
Epoch 81/200  
20/20 [=====] - 0s 3ms/step - loss: 3.8531 -  
accuracy: 0.0204  
Epoch 82/200  
20/20 [=====] - 0s 3ms/step - loss: 3.8383 -  
accuracy: 0.0612  
Epoch 83/200  
20/20 [=====] - 0s 3ms/step - loss: 3.9138 -  
accuracy: 0.0204  
Epoch 84/200

20/20 [=====] - 0s 3ms/step - loss: 3.9167 -  
accuracy: 0.0306  
Epoch 85/200  
20/20 [=====] - 0s 3ms/step - loss: 3.9186 -  
accuracy: 0.0306  
Epoch 86/200  
20/20 [=====] - 0s 3ms/step - loss: 3.8028 -  
accuracy: 0.0510  
Epoch 87/200  
20/20 [=====] - 0s 3ms/step - loss: 3.8385 -  
accuracy: 0.0408  
Epoch 88/200  
20/20 [=====] - 0s 3ms/step - loss: 3.8189 -  
accuracy: 0.0204  
Epoch 89/200  
20/20 [=====] - 0s 5ms/step - loss: 3.8175 -  
accuracy: 0.0714  
Epoch 90/200  
20/20 [=====] - 0s 5ms/step - loss: 3.7812 -  
accuracy: 0.0510  
Epoch 91/200  
20/20 [=====] - 0s 4ms/step - loss: 3.8441 -  
accuracy: 0.0612  
Epoch 92/200  
20/20 [=====] - 0s 4ms/step - loss: 3.8011 -  
accuracy: 0.0612  
Epoch 93/200  
20/20 [=====] - 0s 4ms/step - loss: 3.7657 -  
accuracy: 0.0510  
Epoch 94/200  
20/20 [=====] - 0s 3ms/step - loss: 3.8435 -  
accuracy: 0.0102  
Epoch 95/200  
20/20 [=====] - 0s 3ms/step - loss: 3.7913 -  
accuracy: 0.0306  
Epoch 96/200  
20/20 [=====] - 0s 3ms/step - loss: 3.7911 -  
accuracy: 0.0306  
Epoch 97/200  
20/20 [=====] - 0s 3ms/step - loss: 3.7869 -  
accuracy: 0.0612  
Epoch 98/200  
20/20 [=====] - 0s 3ms/step - loss: 3.8198 -  
accuracy: 0.0408  
Epoch 99/200  
20/20 [=====] - 0s 4ms/step - loss: 3.7766 -  
accuracy: 0.0306  
Epoch 100/200  
20/20 [=====] - 0s 4ms/step - loss: 3.7976 -  
accuracy: 0.0510

Epoch 101/200  
20/20 [=====] - 0s 4ms/step - loss: 3.7822 -  
accuracy: 0.0408  
Epoch 102/200  
20/20 [=====] - 0s 3ms/step - loss: 3.7842 -  
accuracy: 0.0612  
Epoch 103/200  
20/20 [=====] - 0s 3ms/step - loss: 3.8013 -  
accuracy: 0.0306  
Epoch 104/200  
20/20 [=====] - 0s 3ms/step - loss: 3.8039 -  
accuracy: 0.0306  
Epoch 105/200  
20/20 [=====] - 0s 3ms/step - loss: 3.7969 -  
accuracy: 0.0408  
Epoch 106/200  
20/20 [=====] - 0s 3ms/step - loss: 3.8299 -  
accuracy: 0.0408  
Epoch 107/200  
20/20 [=====] - 0s 3ms/step - loss: 3.7974 -  
accuracy: 0.0612  
Epoch 108/200  
20/20 [=====] - 0s 3ms/step - loss: 3.7708 -  
accuracy: 0.0510  
Epoch 109/200  
20/20 [=====] - 0s 4ms/step - loss: 3.8380 -  
accuracy: 0.0408  
Epoch 110/200  
20/20 [=====] - 0s 4ms/step - loss: 3.8218 -  
accuracy: 0.0408  
Epoch 111/200  
20/20 [=====] - 0s 4ms/step - loss: 3.8844 -  
accuracy: 0.0204  
Epoch 112/200  
20/20 [=====] - 0s 4ms/step - loss: 3.7869 -  
accuracy: 0.0408  
Epoch 113/200  
20/20 [=====] - 0s 4ms/step - loss: 3.8558 -  
accuracy: 0.0306  
Epoch 114/200  
20/20 [=====] - 0s 4ms/step - loss: 3.8475 -  
accuracy: 0.0408  
Epoch 115/200  
20/20 [=====] - 0s 4ms/step - loss: 3.7568 -  
accuracy: 0.0408  
Epoch 116/200  
20/20 [=====] - 0s 4ms/step - loss: 3.7808 -  
accuracy: 0.0510  
Epoch 117/200

20/20 [=====] - 0s 4ms/step - loss: 3.8085 -  
accuracy: 0.0408  
Epoch 118/200  
20/20 [=====] - 0s 4ms/step - loss: 3.7444 -  
accuracy: 0.0408  
Epoch 119/200  
20/20 [=====] - 0s 4ms/step - loss: 3.8765 -  
accuracy: 0.0102  
Epoch 120/200  
20/20 [=====] - 0s 4ms/step - loss: 3.7352 -  
accuracy: 0.0612  
Epoch 121/200  
20/20 [=====] - 0s 4ms/step - loss: 3.7841 -  
accuracy: 0.0204  
Epoch 122/200  
20/20 [=====] - 0s 4ms/step - loss: 3.8247 -  
accuracy: 0.0408  
Epoch 123/200  
20/20 [=====] - 0s 4ms/step - loss: 3.7694 -  
accuracy: 0.0510  
Epoch 124/200  
20/20 [=====] - 0s 4ms/step - loss: 3.7703 -  
accuracy: 0.0510  
Epoch 125/200  
20/20 [=====] - 0s 4ms/step - loss: 3.8759 -  
accuracy: 0.0102  
Epoch 126/200  
20/20 [=====] - 0s 4ms/step - loss: 3.7309 -  
accuracy: 0.0510  
Epoch 127/200  
20/20 [=====] - 0s 4ms/step - loss: 3.7564 -  
accuracy: 0.0306  
Epoch 128/200  
20/20 [=====] - 0s 4ms/step - loss: 3.7950 -  
accuracy: 0.0306  
Epoch 129/200  
20/20 [=====] - 0s 4ms/step - loss: 3.7636 -  
accuracy: 0.0510  
Epoch 130/200  
20/20 [=====] - 0s 4ms/step - loss: 3.7364 -  
accuracy: 0.0714  
Epoch 131/200  
20/20 [=====] - 0s 4ms/step - loss: 3.7640 -  
accuracy: 0.0612  
Epoch 132/200  
20/20 [=====] - 0s 4ms/step - loss: 3.7260 -  
accuracy: 0.0306  
Epoch 133/200  
20/20 [=====] - 0s 4ms/step - loss: 3.7537 -  
accuracy: 0.0612

Epoch 134/200  
20/20 [=====] - 0s 4ms/step - loss: 3.8494 -  
accuracy: 0.0408  
Epoch 135/200  
20/20 [=====] - 0s 4ms/step - loss: 3.7989 -  
accuracy: 0.0510  
Epoch 136/200  
20/20 [=====] - 0s 4ms/step - loss: 3.7211 -  
accuracy: 0.0408  
Epoch 137/200  
20/20 [=====] - 0s 4ms/step - loss: 3.7502 -  
accuracy: 0.0306  
Epoch 138/200  
20/20 [=====] - 0s 5ms/step - loss: 3.8003 -  
accuracy: 0.0306  
Epoch 139/200  
20/20 [=====] - 0s 4ms/step - loss: 3.7615 -  
accuracy: 0.0408  
Epoch 140/200  
20/20 [=====] - 0s 4ms/step - loss: 3.7425 -  
accuracy: 0.0714  
Epoch 141/200  
20/20 [=====] - 0s 4ms/step - loss: 3.7554 -  
accuracy: 0.0408  
Epoch 142/200  
20/20 [=====] - 0s 3ms/step - loss: 3.7767 -  
accuracy: 0.0306  
Epoch 143/200  
20/20 [=====] - 0s 3ms/step - loss: 3.7856 -  
accuracy: 0.0306  
Epoch 144/200  
20/20 [=====] - 0s 3ms/step - loss: 3.7184 -  
accuracy: 0.0510  
Epoch 145/200  
20/20 [=====] - 0s 3ms/step - loss: 3.8031 -  
accuracy: 0.0306  
Epoch 146/200  
20/20 [=====] - 0s 4ms/step - loss: 3.8240 -  
accuracy: 0.0714  
Epoch 147/200  
20/20 [=====] - 0s 4ms/step - loss: 3.7586 -  
accuracy: 0.0408  
Epoch 148/200  
20/20 [=====] - 0s 4ms/step - loss: 3.7909 -  
accuracy: 0.0306  
Epoch 149/200  
20/20 [=====] - 0s 3ms/step - loss: 3.7704 -  
accuracy: 0.0510  
Epoch 150/200

20/20 [=====] - 0s 4ms/step - loss: 3.7629 -  
accuracy: 0.0408  
Epoch 151/200  
20/20 [=====] - 0s 3ms/step - loss: 3.7429 -  
accuracy: 0.0510  
Epoch 152/200  
20/20 [=====] - 0s 3ms/step - loss: 3.7721 -  
accuracy: 0.0408  
Epoch 153/200  
20/20 [=====] - 0s 3ms/step - loss: 3.7612 -  
accuracy: 0.0510  
Epoch 154/200  
20/20 [=====] - 0s 4ms/step - loss: 3.7675 -  
accuracy: 0.0408  
Epoch 155/200  
20/20 [=====] - 0s 4ms/step - loss: 3.8322 -  
accuracy: 0.0612  
Epoch 156/200  
20/20 [=====] - 0s 4ms/step - loss: 3.7573 -  
accuracy: 0.0612  
Epoch 157/200  
20/20 [=====] - 0s 4ms/step - loss: 3.7319 -  
accuracy: 0.0408  
Epoch 158/200  
20/20 [=====] - 0s 4ms/step - loss: 3.7469 -  
accuracy: 0.0408  
Epoch 159/200  
20/20 [=====] - 0s 4ms/step - loss: 3.8045 -  
accuracy: 0.0306  
Epoch 160/200  
20/20 [=====] - 0s 4ms/step - loss: 3.7405 -  
accuracy: 0.0612  
Epoch 161/200  
20/20 [=====] - 0s 5ms/step - loss: 3.7473 -  
accuracy: 0.0612  
Epoch 162/200  
20/20 [=====] - 0s 4ms/step - loss: 3.7534 -  
accuracy: 0.0612  
Epoch 163/200  
20/20 [=====] - 0s 4ms/step - loss: 3.7538 -  
accuracy: 0.0510  
Epoch 164/200  
20/20 [=====] - 0s 3ms/step - loss: 3.7627 -  
accuracy: 0.0510  
Epoch 165/200  
20/20 [=====] - 0s 3ms/step - loss: 3.7648 -  
accuracy: 0.0408  
Epoch 166/200  
20/20 [=====] - 0s 3ms/step - loss: 3.7605 -  
accuracy: 0.0306

Epoch 167/200  
20/20 [=====] - 0s 3ms/step - loss: 3.7005 -  
accuracy: 0.0714  
Epoch 168/200  
20/20 [=====] - 0s 4ms/step - loss: 3.7389 -  
accuracy: 0.0408  
Epoch 169/200  
20/20 [=====] - 0s 4ms/step - loss: 3.7555 -  
accuracy: 0.0612  
Epoch 170/200  
20/20 [=====] - 0s 4ms/step - loss: 3.7728 -  
accuracy: 0.0408  
Epoch 171/200  
20/20 [=====] - 0s 4ms/step - loss: 3.8070 -  
accuracy: 0.0714  
Epoch 172/200  
20/20 [=====] - 0s 4ms/step - loss: 3.7565 -  
accuracy: 0.0408  
Epoch 173/200  
20/20 [=====] - 0s 4ms/step - loss: 3.7278 -  
accuracy: 0.0510  
Epoch 174/200  
20/20 [=====] - 0s 4ms/step - loss: 3.7309 -  
accuracy: 0.0612  
Epoch 175/200  
20/20 [=====] - 0s 4ms/step - loss: 3.7128 -  
accuracy: 0.0612  
Epoch 176/200  
20/20 [=====] - 0s 4ms/step - loss: 3.7110 -  
accuracy: 0.0612  
Epoch 177/200  
20/20 [=====] - 0s 4ms/step - loss: 3.7522 -  
accuracy: 0.0408  
Epoch 178/200  
20/20 [=====] - 0s 4ms/step - loss: 3.6885 -  
accuracy: 0.0408  
Epoch 179/200  
20/20 [=====] - 0s 4ms/step - loss: 3.7341 -  
accuracy: 0.0714  
Epoch 180/200  
20/20 [=====] - 0s 4ms/step - loss: 3.7050 -  
accuracy: 0.0510  
Epoch 181/200  
20/20 [=====] - 0s 4ms/step - loss: 3.6441 -  
accuracy: 0.1020  
Epoch 182/200  
20/20 [=====] - 0s 4ms/step - loss: 3.7590 -  
accuracy: 0.0510  
Epoch 183/200

20/20 [=====] - 0s 4ms/step - loss: 3.7675 -  
accuracy: 0.0408  
Epoch 184/200  
20/20 [=====] - 0s 4ms/step - loss: 3.6980 -  
accuracy: 0.0510  
Epoch 185/200  
20/20 [=====] - 0s 4ms/step - loss: 3.7277 -  
accuracy: 0.0612  
Epoch 186/200  
20/20 [=====] - 0s 4ms/step - loss: 3.7570 -  
accuracy: 0.0408  
Epoch 187/200  
20/20 [=====] - 0s 4ms/step - loss: 3.7429 -  
accuracy: 0.0612  
Epoch 188/200  
20/20 [=====] - 0s 4ms/step - loss: 3.7953 -  
accuracy: 0.0204  
Epoch 189/200  
20/20 [=====] - 0s 4ms/step - loss: 3.7373 -  
accuracy: 0.0204  
Epoch 190/200  
20/20 [=====] - 0s 5ms/step - loss: 3.7055 -  
accuracy: 0.0510  
Epoch 191/200  
20/20 [=====] - 0s 6ms/step - loss: 3.7277 -  
accuracy: 0.0612  
Epoch 192/200  
20/20 [=====] - 0s 5ms/step - loss: 3.7090 -  
accuracy: 0.0408  
Epoch 193/200  
20/20 [=====] - 0s 4ms/step - loss: 3.7526 -  
accuracy: 0.0306  
Epoch 194/200  
20/20 [=====] - 0s 4ms/step - loss: 3.7557 -  
accuracy: 0.0306  
Epoch 195/200  
20/20 [=====] - 0s 4ms/step - loss: 3.7519 -  
accuracy: 0.0714  
Epoch 196/200  
20/20 [=====] - 0s 4ms/step - loss: 3.7200 -  
accuracy: 0.0612  
Epoch 197/200  
20/20 [=====] - 0s 4ms/step - loss: 3.7388 -  
accuracy: 0.0306  
Epoch 198/200  
20/20 [=====] - 0s 4ms/step - loss: 3.8180 -  
accuracy: 0.0408  
Epoch 199/200  
20/20 [=====] - 0s 5ms/step - loss: 3.7809 -  
accuracy: 0.0612

Epoch 200/200  
20/20 [=====] - 0s 3ms/step - loss: 3.7203 -  
accuracy: 0.0612

In [12]:

```
# Create model
import tensorflow as tf
model = tf.keras.models.Sequential([
    tf.keras.layers.Dense(128, input_shape=(len(train_x[0]),), activation='relu'),
    tf.keras.layers.Dropout(0.5),
    tf.keras.layers.Dense(64, activation='relu'),
    tf.keras.layers.Dropout(0.5),
    tf.keras.layers.Dense(len(train_y[0]), activation='softmax')])
```

In [14]:

```
import tensorflow as tf
import numpy as np
# Define the optimizer
optimizer = tf.keras.optimizers.legacy.SGD(
    learning_rate=0.01, momentum=0.9, nesterov=True)
# Compile the model
model.compile(
    loss='categorical_crossentropy',
    optimizer=optimizer,
    metrics=['accuracy'])
# Fit the model
history = model.fit(
    np.array(train_x),
    np.array(train_y),
    epochs=200,
    batch_size=5,
    verbose=1)
```

Epoch 1/200  
20/20 [=====] - 1s 3ms/step - loss: 4.5877 -  
accuracy: 0.0102

Epoch 2/200  
20/20 [=====] - 0s 4ms/step - loss: 4.5918 -  
accuracy: 0.0306

Epoch 3/200  
20/20 [=====] - 0s 3ms/step - loss: 4.5345 -  
accuracy: 0.0102

Epoch 4/200  
20/20 [=====] - 0s 4ms/step - loss: 4.5226 -  
accuracy: 0.0204

Epoch 5/200  
20/20 [=====] - 0s 4ms/step - loss: 4.4890 -  
accuracy: 0.0306

Epoch 6/200  
20/20 [=====] - 0s 4ms/step - loss: 4.4344 -  
accuracy: 0.0816

Epoch 7/200  
20/20 [=====] - 0s 4ms/step - loss: 4.3948 -  
accuracy: 0.0102

Epoch 8/200  
20/20 [=====] - 0s 4ms/step - loss: 4.3135 -  
accuracy: 0.0612

Epoch 9/200

20/20 [=====] - 0s 4ms/step - loss: 4.2641 -  
accuracy: 0.0816  
Epoch 10/200  
20/20 [=====] - 0s 5ms/step - loss: 4.1906 -  
accuracy: 0.0918  
Epoch 11/200  
20/20 [=====] - 0s 4ms/step - loss: 4.1793 -  
accuracy: 0.0714  
Epoch 12/200  
20/20 [=====] - 0s 4ms/step - loss: 3.9752 -  
accuracy: 0.1020  
Epoch 13/200  
20/20 [=====] - 0s 4ms/step - loss: 3.9363 -  
accuracy: 0.1020  
Epoch 14/200  
20/20 [=====] - 0s 4ms/step - loss: 3.9040 -  
accuracy: 0.0612  
Epoch 15/200  
20/20 [=====] - 0s 3ms/step - loss: 3.6177 -  
accuracy: 0.1939  
Epoch 16/200  
20/20 [=====] - 0s 4ms/step - loss: 3.6120 -  
accuracy: 0.1735  
Epoch 17/200  
20/20 [=====] - 0s 4ms/step - loss: 3.3526 -  
accuracy: 0.1735  
Epoch 18/200  
20/20 [=====] - 0s 4ms/step - loss: 3.2849 -  
accuracy: 0.2347  
Epoch 19/200  
20/20 [=====] - 0s 4ms/step - loss: 3.1446 -  
accuracy: 0.3061  
Epoch 20/200  
20/20 [=====] - 0s 4ms/step - loss: 3.1874 -  
accuracy: 0.2041  
Epoch 21/200  
20/20 [=====] - 0s 4ms/step - loss: 2.9625 -  
accuracy: 0.2959  
Epoch 22/200  
20/20 [=====] - 0s 4ms/step - loss: 2.9526 -  
accuracy: 0.2551  
Epoch 23/200  
20/20 [=====] - 0s 4ms/step - loss: 2.6877 -  
accuracy: 0.3265  
Epoch 24/200  
20/20 [=====] - 0s 4ms/step - loss: 2.6980 -  
accuracy: 0.3265  
Epoch 25/200  
20/20 [=====] - 0s 4ms/step - loss: 2.4302 -  
accuracy: 0.3776

Epoch 26/200  
20/20 [=====] - 0s 4ms/step - loss: 2.5220 -  
accuracy: 0.3367  
Epoch 27/200  
20/20 [=====] - 0s 4ms/step - loss: 2.4400 -  
accuracy: 0.3776  
Epoch 28/200  
20/20 [=====] - 0s 4ms/step - loss: 2.5253 -  
accuracy: 0.3673  
Epoch 29/200  
20/20 [=====] - 0s 4ms/step - loss: 2.5598 -  
accuracy: 0.3776  
Epoch 30/200  
20/20 [=====] - 0s 4ms/step - loss: 2.1005 -  
accuracy: 0.4592  
Epoch 31/200  
20/20 [=====] - 0s 4ms/step - loss: 2.0363 -  
accuracy: 0.4490  
Epoch 32/200  
20/20 [=====] - 0s 6ms/step - loss: 1.8716 -  
accuracy: 0.4592  
Epoch 33/200  
20/20 [=====] - 0s 6ms/step - loss: 1.8381 -  
accuracy: 0.4898  
Epoch 34/200  
20/20 [=====] - 0s 6ms/step - loss: 2.0079 -  
accuracy: 0.4694  
Epoch 35/200  
20/20 [=====] - 0s 4ms/step - loss: 1.9188 -  
accuracy: 0.4796  
Epoch 36/200  
20/20 [=====] - 0s 4ms/step - loss: 1.7128 -  
accuracy: 0.5306  
Epoch 37/200  
20/20 [=====] - 0s 4ms/step - loss: 1.8989 -  
accuracy: 0.4388  
Epoch 38/200  
20/20 [=====] - 0s 3ms/step - loss: 1.6576 -  
accuracy: 0.5204  
Epoch 39/200  
20/20 [=====] - 0s 4ms/step - loss: 1.5829 -  
accuracy: 0.5306  
Epoch 40/200  
20/20 [=====] - 0s 4ms/step - loss: 1.7884 -  
accuracy: 0.5102  
Epoch 41/200  
20/20 [=====] - 0s 5ms/step - loss: 1.5420 -  
accuracy: 0.5408  
Epoch 42/200

20/20 [=====] - 0s 4ms/step - loss: 1.7868 -  
accuracy: 0.4490  
Epoch 43/200  
20/20 [=====] - 0s 4ms/step - loss: 1.8635 -  
accuracy: 0.4286  
Epoch 44/200  
20/20 [=====] - 0s 4ms/step - loss: 1.4172 -  
accuracy: 0.6122  
Epoch 45/200  
20/20 [=====] - 0s 4ms/step - loss: 1.5198 -  
accuracy: 0.5612  
Epoch 46/200  
20/20 [=====] - 0s 4ms/step - loss: 1.4273 -  
accuracy: 0.5918  
Epoch 47/200  
20/20 [=====] - 0s 4ms/step - loss: 1.3082 -  
accuracy: 0.5714  
Epoch 48/200  
20/20 [=====] - 0s 4ms/step - loss: 1.4330 -  
accuracy: 0.5612  
Epoch 49/200  
20/20 [=====] - 0s 4ms/step - loss: 1.4148 -  
accuracy: 0.5408  
Epoch 50/200  
20/20 [=====] - 0s 4ms/step - loss: 1.4237 -  
accuracy: 0.5816  
Epoch 51/200  
20/20 [=====] - 0s 4ms/step - loss: 1.2315 -  
accuracy: 0.6429  
Epoch 52/200  
20/20 [=====] - 0s 4ms/step - loss: 1.4174 -  
accuracy: 0.5612  
Epoch 53/200  
20/20 [=====] - 0s 4ms/step - loss: 1.0642 -  
accuracy: 0.7143  
Epoch 54/200  
20/20 [=====] - 0s 4ms/step - loss: 1.2573 -  
accuracy: 0.6327  
Epoch 55/200  
20/20 [=====] - 0s 4ms/step - loss: 1.3597 -  
accuracy: 0.6020  
Epoch 56/200  
20/20 [=====] - 0s 4ms/step - loss: 1.3605 -  
accuracy: 0.6122  
Epoch 57/200  
20/20 [=====] - 0s 4ms/step - loss: 1.4918 -  
accuracy: 0.5816  
Epoch 58/200  
20/20 [=====] - 0s 4ms/step - loss: 1.2565 -  
accuracy: 0.5714

Epoch 59/200  
20/20 [=====] - 0s 4ms/step - loss: 1.1454 -  
accuracy: 0.6224  
Epoch 60/200  
20/20 [=====] - 0s 4ms/step - loss: 1.3084 -  
accuracy: 0.6327  
Epoch 61/200  
20/20 [=====] - 0s 4ms/step - loss: 1.3067 -  
accuracy: 0.6429  
Epoch 62/200  
20/20 [=====] - 0s 4ms/step - loss: 1.2712 -  
accuracy: 0.6122  
Epoch 63/200  
20/20 [=====] - 0s 4ms/step - loss: 1.1804 -  
accuracy: 0.5612  
Epoch 64/200  
20/20 [=====] - 0s 4ms/step - loss: 1.1749 -  
accuracy: 0.6224  
Epoch 65/200  
20/20 [=====] - 0s 4ms/step - loss: 1.2680 -  
accuracy: 0.6327  
Epoch 66/200  
20/20 [=====] - 0s 4ms/step - loss: 1.2512 -  
accuracy: 0.6020  
Epoch 67/200  
20/20 [=====] - 0s 4ms/step - loss: 1.3872 -  
accuracy: 0.5510  
Epoch 68/200  
20/20 [=====] - 0s 4ms/step - loss: 1.0921 -  
accuracy: 0.6122  
Epoch 69/200  
20/20 [=====] - 0s 4ms/step - loss: 1.0853 -  
accuracy: 0.6429  
Epoch 70/200  
20/20 [=====] - 0s 4ms/step - loss: 1.0324 -  
accuracy: 0.7653  
Epoch 71/200  
20/20 [=====] - 0s 4ms/step - loss: 1.0192 -  
accuracy: 0.7041  
Epoch 72/200  
20/20 [=====] - 0s 4ms/step - loss: 1.0409 -  
accuracy: 0.6633  
Epoch 73/200  
20/20 [=====] - 0s 4ms/step - loss: 1.1619 -  
accuracy: 0.6735  
Epoch 74/200  
20/20 [=====] - 0s 4ms/step - loss: 1.1284 -  
accuracy: 0.6531  
Epoch 75/200

20/20 [=====] - 0s 4ms/step - loss: 1.0808 -  
accuracy: 0.6531  
Epoch 76/200  
20/20 [=====] - 0s 4ms/step - loss: 1.0203 -  
accuracy: 0.6939  
Epoch 77/200  
20/20 [=====] - 0s 4ms/step - loss: 0.8565 -  
accuracy: 0.7347  
Epoch 78/200  
20/20 [=====] - 0s 4ms/step - loss: 1.0411 -  
accuracy: 0.6939  
Epoch 79/200  
20/20 [=====] - 0s 4ms/step - loss: 0.9346 -  
accuracy: 0.6939  
Epoch 80/200  
20/20 [=====] - 0s 4ms/step - loss: 1.0930 -  
accuracy: 0.6531  
Epoch 81/200  
20/20 [=====] - 0s 4ms/step - loss: 1.0081 -  
accuracy: 0.7245  
Epoch 82/200  
20/20 [=====] - 0s 4ms/step - loss: 0.9535 -  
accuracy: 0.6837  
Epoch 83/200  
20/20 [=====] - 0s 4ms/step - loss: 0.8324 -  
accuracy: 0.7449  
Epoch 84/200  
20/20 [=====] - 0s 4ms/step - loss: 0.8356 -  
accuracy: 0.7347  
Epoch 85/200  
20/20 [=====] - 0s 4ms/step - loss: 0.9635 -  
accuracy: 0.7041  
Epoch 86/200  
20/20 [=====] - 0s 4ms/step - loss: 1.0297 -  
accuracy: 0.7143  
Epoch 87/200  
20/20 [=====] - 0s 4ms/step - loss: 1.1087 -  
accuracy: 0.6837  
Epoch 88/200  
20/20 [=====] - 0s 4ms/step - loss: 1.0010 -  
accuracy: 0.6735  
Epoch 89/200  
20/20 [=====] - 0s 4ms/step - loss: 1.0002 -  
accuracy: 0.6735  
Epoch 90/200  
20/20 [=====] - 0s 4ms/step - loss: 0.8265 -  
accuracy: 0.6939  
Epoch 91/200  
20/20 [=====] - 0s 4ms/step - loss: 0.8170 -  
accuracy: 0.7857

Epoch 92/200  
20/20 [=====] - 0s 4ms/step - loss: 0.7264 -  
accuracy: 0.7551  
Epoch 93/200  
20/20 [=====] - 0s 4ms/step - loss: 1.0524 -  
accuracy: 0.6122  
Epoch 94/200  
20/20 [=====] - 0s 4ms/step - loss: 0.8393 -  
accuracy: 0.7347  
Epoch 95/200  
20/20 [=====] - 0s 4ms/step - loss: 0.8545 -  
accuracy: 0.6939  
Epoch 96/200  
20/20 [=====] - 0s 6ms/step - loss: 0.7695 -  
accuracy: 0.7347  
Epoch 97/200  
20/20 [=====] - 0s 4ms/step - loss: 0.7367 -  
accuracy: 0.8061  
Epoch 98/200  
20/20 [=====] - 0s 5ms/step - loss: 0.8823 -  
accuracy: 0.7245  
Epoch 99/200  
20/20 [=====] - 0s 4ms/step - loss: 1.0859 -  
accuracy: 0.6633  
Epoch 100/200  
20/20 [=====] - 0s 4ms/step - loss: 1.1181 -  
accuracy: 0.6020  
Epoch 101/200  
20/20 [=====] - 0s 4ms/step - loss: 0.8457 -  
accuracy: 0.7143  
Epoch 102/200  
20/20 [=====] - 0s 4ms/step - loss: 0.7366 -  
accuracy: 0.7755  
Epoch 103/200  
20/20 [=====] - 0s 4ms/step - loss: 0.8722 -  
accuracy: 0.6837  
Epoch 104/200  
20/20 [=====] - 0s 4ms/step - loss: 0.7993 -  
accuracy: 0.7347  
Epoch 105/200  
20/20 [=====] - 0s 4ms/step - loss: 0.7629 -  
accuracy: 0.7551  
Epoch 106/200  
20/20 [=====] - 0s 4ms/step - loss: 0.8900 -  
accuracy: 0.7041  
Epoch 107/200  
20/20 [=====] - 0s 4ms/step - loss: 1.0170 -  
accuracy: 0.6531  
Epoch 108/200

20/20 [=====] - 0s 3ms/step - loss: 0.6552 -  
accuracy: 0.7653  
Epoch 109/200  
20/20 [=====] - 0s 4ms/step - loss: 0.8506 -  
accuracy: 0.7449  
Epoch 110/200  
20/20 [=====] - 0s 4ms/step - loss: 0.9002 -  
accuracy: 0.7347  
Epoch 111/200  
20/20 [=====] - 0s 4ms/step - loss: 0.9501 -  
accuracy: 0.6939  
Epoch 112/200  
20/20 [=====] - 0s 5ms/step - loss: 0.7485 -  
accuracy: 0.7347  
Epoch 113/200  
20/20 [=====] - 0s 4ms/step - loss: 0.8769 -  
accuracy: 0.6735  
Epoch 114/200  
20/20 [=====] - 0s 4ms/step - loss: 0.9159 -  
accuracy: 0.7347  
Epoch 115/200  
20/20 [=====] - 0s 5ms/step - loss: 0.6664 -  
accuracy: 0.7857  
Epoch 116/200  
20/20 [=====] - 0s 4ms/step - loss: 0.6482 -  
accuracy: 0.7959  
Epoch 117/200  
20/20 [=====] - 0s 4ms/step - loss: 0.7799 -  
accuracy: 0.6939  
Epoch 118/200  
20/20 [=====] - 0s 4ms/step - loss: 0.7458 -  
accuracy: 0.7653  
Epoch 119/200  
20/20 [=====] - 0s 4ms/step - loss: 1.0763 -  
accuracy: 0.6531  
Epoch 120/200  
20/20 [=====] - 0s 4ms/step - loss: 0.9488 -  
accuracy: 0.7653  
Epoch 121/200  
20/20 [=====] - 0s 4ms/step - loss: 0.8502 -  
accuracy: 0.7347  
Epoch 122/200  
20/20 [=====] - 0s 4ms/step - loss: 0.8856 -  
accuracy: 0.7245  
Epoch 123/200  
20/20 [=====] - 0s 4ms/step - loss: 0.6784 -  
accuracy: 0.7857  
Epoch 124/200  
20/20 [=====] - 0s 4ms/step - loss: 0.7805 -  
accuracy: 0.7653

Epoch 125/200  
20/20 [=====] - 0s 4ms/step - loss: 0.8979 -  
accuracy: 0.7653  
Epoch 126/200  
20/20 [=====] - 0s 4ms/step - loss: 0.6458 -  
accuracy: 0.7653  
Epoch 127/200  
20/20 [=====] - 0s 4ms/step - loss: 0.9789 -  
accuracy: 0.7245  
Epoch 128/200  
20/20 [=====] - 0s 4ms/step - loss: 0.7057 -  
accuracy: 0.7449  
Epoch 129/200  
20/20 [=====] - 0s 4ms/step - loss: 0.6682 -  
accuracy: 0.8061  
Epoch 130/200  
20/20 [=====] - 0s 4ms/step - loss: 0.7205 -  
accuracy: 0.7755  
Epoch 131/200  
20/20 [=====] - 0s 4ms/step - loss: 0.6986 -  
accuracy: 0.8061  
Epoch 132/200  
20/20 [=====] - 0s 4ms/step - loss: 0.7012 -  
accuracy: 0.7959  
Epoch 133/200  
20/20 [=====] - 0s 3ms/step - loss: 0.7191 -  
accuracy: 0.7347  
Epoch 134/200  
20/20 [=====] - 0s 4ms/step - loss: 0.9312 -  
accuracy: 0.7041  
Epoch 135/200  
20/20 [=====] - 0s 4ms/step - loss: 1.0022 -  
accuracy: 0.6531  
Epoch 136/200  
20/20 [=====] - 0s 4ms/step - loss: 0.9581 -  
accuracy: 0.7347  
Epoch 137/200  
20/20 [=====] - 0s 4ms/step - loss: 0.7698 -  
accuracy: 0.7857  
Epoch 138/200  
20/20 [=====] - 0s 4ms/step - loss: 0.6298 -  
accuracy: 0.8061  
Epoch 139/200  
20/20 [=====] - 0s 4ms/step - loss: 0.7315 -  
accuracy: 0.7755  
Epoch 140/200  
20/20 [=====] - 0s 4ms/step - loss: 0.9342 -  
accuracy: 0.6633  
Epoch 141/200

20/20 [=====] - 0s 4ms/step - loss: 0.7262 -  
accuracy: 0.8061  
Epoch 142/200  
20/20 [=====] - 0s 4ms/step - loss: 0.8132 -  
accuracy: 0.6837  
Epoch 143/200  
20/20 [=====] - 0s 4ms/step - loss: 0.7311 -  
accuracy: 0.7551  
Epoch 144/200  
20/20 [=====] - 0s 4ms/step - loss: 0.8258 -  
accuracy: 0.7755  
Epoch 145/200  
20/20 [=====] - 0s 4ms/step - loss: 0.9453 -  
accuracy: 0.6735  
Epoch 146/200  
20/20 [=====] - 0s 4ms/step - loss: 0.8450 -  
accuracy: 0.7755  
Epoch 147/200  
20/20 [=====] - 0s 4ms/step - loss: 0.6024 -  
accuracy: 0.7857  
Epoch 148/200  
20/20 [=====] - 0s 4ms/step - loss: 0.7752 -  
accuracy: 0.7551  
Epoch 149/200  
20/20 [=====] - 0s 4ms/step - loss: 0.6129 -  
accuracy: 0.8061  
Epoch 150/200  
20/20 [=====] - 0s 4ms/step - loss: 0.7045 -  
accuracy: 0.7755  
Epoch 151/200  
20/20 [=====] - 0s 4ms/step - loss: 0.7501 -  
accuracy: 0.7755  
Epoch 152/200  
20/20 [=====] - 0s 4ms/step - loss: 0.6306 -  
accuracy: 0.7347  
Epoch 153/200  
20/20 [=====] - 0s 4ms/step - loss: 0.6799 -  
accuracy: 0.7551  
Epoch 154/200  
20/20 [=====] - 0s 4ms/step - loss: 0.6101 -  
accuracy: 0.8367  
Epoch 155/200  
20/20 [=====] - 0s 4ms/step - loss: 0.7870 -  
accuracy: 0.7551  
Epoch 156/200  
20/20 [=====] - 0s 4ms/step - loss: 0.9788 -  
accuracy: 0.7245  
Epoch 157/200  
20/20 [=====] - 0s 4ms/step - loss: 0.5724 -  
accuracy: 0.8367

Epoch 158/200  
20/20 [=====] - 0s 4ms/step - loss: 0.7829 -  
accuracy: 0.7857  
Epoch 159/200  
20/20 [=====] - 0s 5ms/step - loss: 0.6118 -  
accuracy: 0.7551  
Epoch 160/200  
20/20 [=====] - 0s 4ms/step - loss: 0.6834 -  
accuracy: 0.8061  
Epoch 161/200  
20/20 [=====] - 0s 4ms/step - loss: 0.6445 -  
accuracy: 0.7857  
Epoch 162/200  
20/20 [=====] - 0s 4ms/step - loss: 0.9644 -  
accuracy: 0.7449  
Epoch 163/200  
20/20 [=====] - 0s 4ms/step - loss: 0.7760 -  
accuracy: 0.7245  
Epoch 164/200  
20/20 [=====] - 0s 4ms/step - loss: 0.8336 -  
accuracy: 0.7449  
Epoch 165/200  
20/20 [=====] - 0s 4ms/step - loss: 0.7780 -  
accuracy: 0.7653  
Epoch 166/200  
20/20 [=====] - 0s 4ms/step - loss: 0.8563 -  
accuracy: 0.7449  
Epoch 167/200  
20/20 [=====] - 0s 5ms/step - loss: 1.0178 -  
accuracy: 0.6939  
Epoch 168/200  
20/20 [=====] - 0s 4ms/step - loss: 0.6782 -  
accuracy: 0.8163  
Epoch 169/200  
20/20 [=====] - 0s 4ms/step - loss: 0.7929 -  
accuracy: 0.7245  
Epoch 170/200  
20/20 [=====] - 0s 4ms/step - loss: 0.6698 -  
accuracy: 0.7551  
Epoch 171/200  
20/20 [=====] - 0s 4ms/step - loss: 0.8326 -  
accuracy: 0.7857  
Epoch 172/200  
20/20 [=====] - 0s 4ms/step - loss: 0.5731 -  
accuracy: 0.7857  
Epoch 173/200  
20/20 [=====] - 0s 4ms/step - loss: 0.7139 -  
accuracy: 0.7857  
Epoch 174/200

20/20 [=====] - 0s 4ms/step - loss: 0.4801 -  
accuracy: 0.8367  
Epoch 175/200  
20/20 [=====] - 0s 4ms/step - loss: 0.8064 -  
accuracy: 0.7653  
Epoch 176/200  
20/20 [=====] - 0s 4ms/step - loss: 0.7583 -  
accuracy: 0.7959  
Epoch 177/200  
20/20 [=====] - 0s 4ms/step - loss: 0.7289 -  
accuracy: 0.7653  
Epoch 178/200  
20/20 [=====] - 0s 5ms/step - loss: 0.7916 -  
accuracy: 0.7245  
Epoch 179/200  
20/20 [=====] - 0s 4ms/step - loss: 0.8034 -  
accuracy: 0.7143  
Epoch 180/200  
20/20 [=====] - 0s 4ms/step - loss: 0.9538 -  
accuracy: 0.7449  
Epoch 181/200  
20/20 [=====] - 0s 4ms/step - loss: 0.7520 -  
accuracy: 0.7755  
Epoch 182/200  
20/20 [=====] - 0s 4ms/step - loss: 0.7745 -  
accuracy: 0.7551  
Epoch 183/200  
20/20 [=====] - 0s 4ms/step - loss: 0.7902 -  
accuracy: 0.7653  
Epoch 184/200  
20/20 [=====] - 0s 4ms/step - loss: 0.9509 -  
accuracy: 0.7143  
Epoch 185/200  
20/20 [=====] - 0s 4ms/step - loss: 0.8444 -  
accuracy: 0.7551  
Epoch 186/200  
20/20 [=====] - 0s 4ms/step - loss: 0.9513 -  
accuracy: 0.7245  
Epoch 187/200  
20/20 [=====] - 0s 10ms/step - loss: 0.7703 -  
accuracy: 0.7143  
Epoch 188/200  
20/20 [=====] - 0s 4ms/step - loss: 0.7722 -  
accuracy: 0.7653  
Epoch 189/200  
20/20 [=====] - 0s 4ms/step - loss: 0.7945 -  
accuracy: 0.7857  
Epoch 190/200  
20/20 [=====] - 0s 5ms/step - loss: 0.4775 -  
accuracy: 0.8571

```

Epoch 191/200
20/20 [=====] - 0s 4ms/step - loss: 0.7423 -
accuracy: 0.7551
Epoch 192/200
20/20 [=====] - 0s 3ms/step - loss: 0.8125 -
accuracy: 0.7551
Epoch 193/200
20/20 [=====] - 0s 3ms/step - loss: 0.8549 -
accuracy: 0.7653
Epoch 194/200
20/20 [=====] - 0s 4ms/step - loss: 0.9063 -
accuracy: 0.6837
Epoch 195/200
20/20 [=====] - 0s 3ms/step - loss: 0.5778 -
accuracy: 0.7857
Epoch 196/200
20/20 [=====] - 0s 4ms/step - loss: 0.8108 -
accuracy: 0.8367
Epoch 197/200
20/20 [=====] - 0s 4ms/step - loss: 0.9329 -
accuracy: 0.7245
Epoch 198/200
20/20 [=====] - 0s 4ms/step - loss: 0.8790 -
accuracy: 0.7653
Epoch 199/200
20/20 [=====] - 0s 4ms/step - loss: 0.5453 -
accuracy: 0.8469
Epoch 200/200
20/20 [=====] - 0s 4ms/step - loss: 0.7183 -
accuracy: 0.7449

```

In [1]:

```

import matplotlib.pyplot as plt
# Get the loss history from the training
loss = hist.history['loss']
# Create a list of epochs to use as the x-axis
epochs = range(1, len(loss) + 1)
# Plot the loss
plt.plot(epochs, loss, 'b', label='Training Loss')
plt.title('Training Loss')
plt.xlabel('Epochs')
plt.ylabel('Loss')
plt.legend()
plt.show()

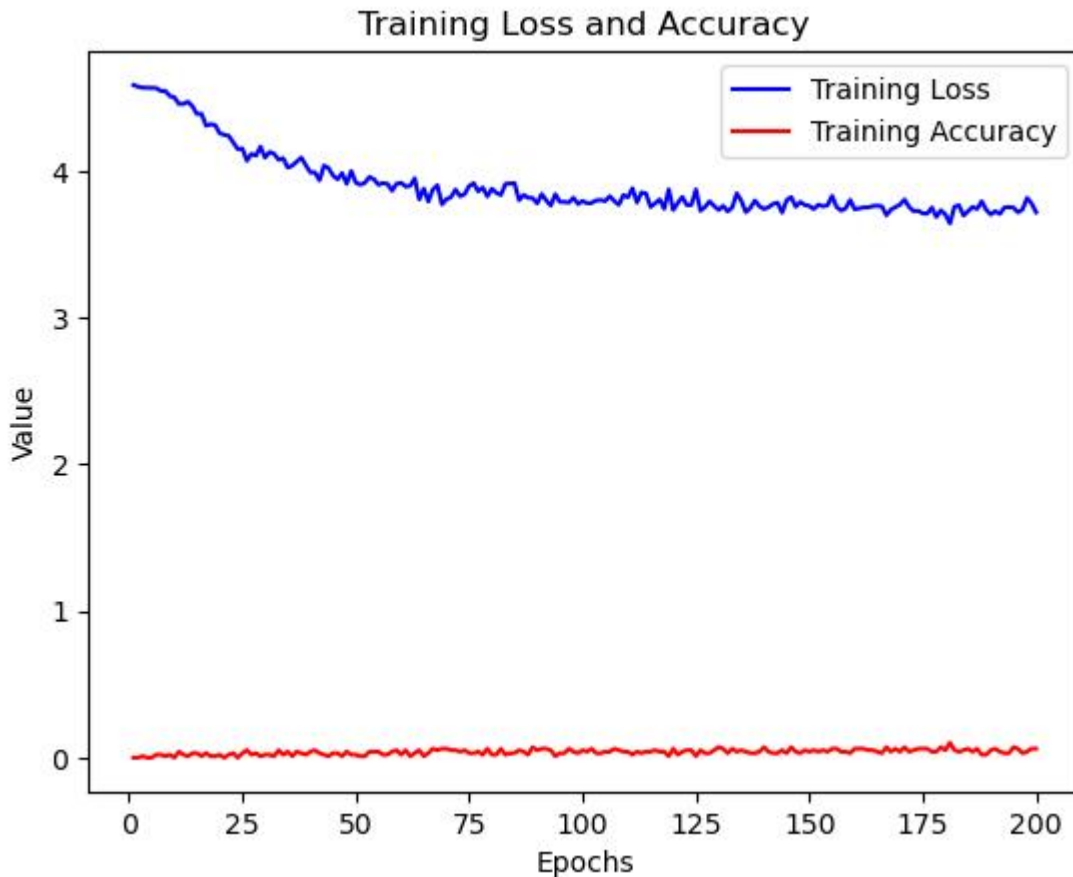
```

In [31]:

```

import matplotlib.pyplot as plt
# Get the training history
train_loss = hist.history['loss']
train_accuracy = hist.history['accuracy']
# Generate the x-axis values (epochs)
epochs = range(1, len(train_loss) + 1)
# Plot the training loss and accuracy on the same graph
plt.plot(epochs, train_loss, 'b', label='Training Loss')
plt.plot(epochs, train_accuracy, 'r', label='Training Accuracy')
plt.title('Training Loss and Accuracy')
plt.xlabel('Epochs')
plt.ylabel('Value')
plt.legend()
plt.show()

```



In [32]:

```

import matplotlib.pyplot as plt
# Split the training dataset into training and validation sets
validation_split = 0.2
num_validation_samples = int(validation_split * len(train_x))
train_x_partial = train_x[:-num_validation_samples]
train_y_partial = train_y[:-num_validation_samples]
val_x = train_x[-num_validation_samples:]
val_y = train_y[-num_validation_samples:]
# Create the model
model = Sequential()
model.add(Dense(128, input_shape=(len(train_x[0]),),
activation='relu'))
model.add(Dropout(0.5))
model.add(Dense(64, activation='relu'))
model.add(Dropout(0.5))
model.add(Dense(len(train_y[0]), activation='softmax'))
# Compile the model
sgd = SGD(lr=0.01, decay=1e-6, momentum=0.9, nesterov=True)
model.compile(loss='categorical_crossentropy', optimizer=sgd,
metrics=['accuracy'])
# Train the model and keep track of the training and validation loss
history = model.fit(np.array(train_x_partial), np.array(train_y_partial),
validation_data=(np.array(val_x), np.array(val_y)),
epochs=200, batch_size=5, verbose=1)
# Plot the training and validation loss
train_loss = history.history['loss']
val_loss = history.history['val_loss']
epochs = range(1, len(train_loss) + 1)
plt.plot(epochs, train_loss, 'b', label='Training Loss')
plt.plot(epochs, val_loss, 'r', label='Validation Loss')
plt.title('Training and Validation Loss')
plt.xlabel('Epochs')
plt.ylabel('Loss')
plt.legend()
plt.show()
Epoch 1/200

```

16/16 [=====] - 1s 20ms/step - loss: 4.5986 -  
accuracy: 0.0127 - val\_loss: 4.5941 - val\_accuracy: 0.0000e+00  
Epoch 2/200  
16/16 [=====] - 0s 7ms/step - loss: 4.5888 -  
accuracy: 0.0000e+00 - val\_loss: 4.6130 - val\_accuracy: 0.0000e+00  
Epoch 3/200  
16/16 [=====] - 0s 7ms/step - loss: 4.5945 -  
accuracy: 0.0000e+00 - val\_loss: 4.6349 - val\_accuracy: 0.0000e+00  
Epoch 4/200  
16/16 [=====] - 0s 7ms/step - loss: 4.5684 -  
accuracy: 0.0127 - val\_loss: 4.6533 - val\_accuracy: 0.0000e+00  
Epoch 5/200  
16/16 [=====] - 0s 7ms/step - loss: 4.5770 -  
accuracy: 0.0127 - val\_loss: 4.6718 - val\_accuracy: 0.0000e+00  
Epoch 6/200  
16/16 [=====] - 0s 7ms/step - loss: 4.5614 -  
accuracy: 0.0127 - val\_loss: 4.6921 - val\_accuracy: 0.0000e+00  
Epoch 7/200  
16/16 [=====] - 0s 6ms/step - loss: 4.5545 -  
accuracy: 0.0127 - val\_loss: 4.7089 - val\_accuracy: 0.0000e+00  
Epoch 8/200  
16/16 [=====] - 0s 7ms/step - loss: 4.5492 -  
accuracy: 0.0127 - val\_loss: 4.7315 - val\_accuracy: 0.0000e+00  
Epoch 9/200  
16/16 [=====] - 0s 6ms/step - loss: 4.5217 -  
accuracy: 0.0253 - val\_loss: 4.7588 - val\_accuracy: 0.0000e+00  
Epoch 10/200  
16/16 [=====] - 0s 8ms/step - loss: 4.5283 -  
accuracy: 0.0000e+00 - val\_loss: 4.7781 - val\_accuracy: 0.0000e+00  
Epoch 11/200  
16/16 [=====] - 0s 7ms/step - loss: 4.5204 -  
accuracy: 0.0380 - val\_loss: 4.8041 - val\_accuracy: 0.0000e+00  
Epoch 12/200  
16/16 [=====] - 0s 7ms/step - loss: 4.5184 -  
accuracy: 0.0253 - val\_loss: 4.8281 - val\_accuracy: 0.0000e+00  
Epoch 13/200  
16/16 [=====] - 0s 7ms/step - loss: 4.4943 -  
accuracy: 0.0253 - val\_loss: 4.8508 - val\_accuracy: 0.0000e+00  
Epoch 14/200  
16/16 [=====] - 0s 7ms/step - loss: 4.4709 -  
accuracy: 0.0000e+00 - val\_loss: 4.8811 - val\_accuracy: 0.0000e+00  
Epoch 15/200  
16/16 [=====] - 0s 7ms/step - loss: 4.4861 -  
accuracy: 0.0000e+00 - val\_loss: 4.9110 - val\_accuracy: 0.0000e+00  
Epoch 16/200  
16/16 [=====] - 0s 6ms/step - loss: 4.4359 -  
accuracy: 0.0253 - val\_loss: 4.9397 - val\_accuracy: 0.0000e+00  
Epoch 17/200  
16/16 [=====] - 0s 7ms/step - loss: 4.4530 -  
accuracy: 0.0253 - val\_loss: 4.9774 - val\_accuracy: 0.0000e+00

Epoch 18/200  
16/16 [=====] - 0s 6ms/step - loss: 4.4137 -  
accuracy: 0.0253 - val\_loss: 5.0180 - val\_accuracy: 0.0000e+00  
Epoch 19/200  
16/16 [=====] - 0s 7ms/step - loss: 4.4061 -  
accuracy: 0.0253 - val\_loss: 5.0551 - val\_accuracy: 0.0000e+00  
Epoch 20/200  
16/16 [=====] - 0s 6ms/step - loss: 4.3324 -  
accuracy: 0.0253 - val\_loss: 5.1017 - val\_accuracy: 0.0000e+00  
Epoch 21/200  
16/16 [=====] - 0s 6ms/step - loss: 4.3136 -  
accuracy: 0.0506 - val\_loss: 5.1674 - val\_accuracy: 0.0000e+00  
Epoch 22/200  
16/16 [=====] - 0s 7ms/step - loss: 4.2597 -  
accuracy: 0.0633 - val\_loss: 5.2210 - val\_accuracy: 0.0000e+00  
Epoch 23/200  
16/16 [=====] - 0s 7ms/step - loss: 4.2259 -  
accuracy: 0.0380 - val\_loss: 5.2896 - val\_accuracy: 0.0000e+00  
Epoch 24/200  
16/16 [=====] - 0s 7ms/step - loss: 4.2599 -  
accuracy: 0.0253 - val\_loss: 5.3370 - val\_accuracy: 0.0000e+00  
Epoch 25/200  
16/16 [=====] - 0s 6ms/step - loss: 4.1240 -  
accuracy: 0.0759 - val\_loss: 5.4083 - val\_accuracy: 0.0000e+00  
Epoch 26/200  
16/16 [=====] - 0s 7ms/step - loss: 4.1633 -  
accuracy: 0.0380 - val\_loss: 5.4811 - val\_accuracy: 0.0000e+00  
Epoch 27/200  
16/16 [=====] - 0s 6ms/step - loss: 4.0735 -  
accuracy: 0.0506 - val\_loss: 5.5435 - val\_accuracy: 0.0000e+00  
Epoch 28/200  
16/16 [=====] - 0s 6ms/step - loss: 4.0967 -  
accuracy: 0.0380 - val\_loss: 5.6000 - val\_accuracy: 0.0000e+00  
Epoch 29/200  
16/16 [=====] - 0s 8ms/step - loss: 4.0284 -  
accuracy: 0.0127 - val\_loss: 5.6470 - val\_accuracy: 0.0000e+00  
Epoch 30/200  
16/16 [=====] - 0s 7ms/step - loss: 4.0163 -  
accuracy: 0.0506 - val\_loss: 5.6930 - val\_accuracy: 0.0000e+00  
Epoch 31/200  
16/16 [=====] - 0s 7ms/step - loss: 3.9744 -  
accuracy: 0.0506 - val\_loss: 5.7499 - val\_accuracy: 0.0000e+00  
Epoch 32/200  
16/16 [=====] - 0s 7ms/step - loss: 4.0179 -  
accuracy: 0.0127 - val\_loss: 5.7844 - val\_accuracy: 0.0000e+00  
Epoch 33/200  
16/16 [=====] - 0s 15ms/step - loss: 4.0130 -  
accuracy: 0.0380 - val\_loss: 5.8593 - val\_accuracy: 0.0000e+00  
Epoch 34/200

16/16 [=====] - 0s 9ms/step - loss: 3.9709 -  
accuracy: 0.0380 - val\_loss: 5.9557 - val\_accuracy: 0.0000e+00  
Epoch 35/200  
16/16 [=====] - 0s 6ms/step - loss: 3.9070 -  
accuracy: 0.0127 - val\_loss: 6.0447 - val\_accuracy: 0.0000e+00  
Epoch 36/200  
16/16 [=====] - 0s 6ms/step - loss: 3.9773 -  
accuracy: 0.0633 - val\_loss: 6.0571 - val\_accuracy: 0.0000e+00  
Epoch 37/200  
16/16 [=====] - 0s 7ms/step - loss: 3.9362 -  
accuracy: 0.0380 - val\_loss: 6.0872 - val\_accuracy: 0.0000e+00  
Epoch 38/200  
16/16 [=====] - 0s 7ms/step - loss: 3.8671 -  
accuracy: 0.0506 - val\_loss: 6.1527 - val\_accuracy: 0.0000e+00  
Epoch 39/200  
16/16 [=====] - 0s 7ms/step - loss: 3.8983 -  
accuracy: 0.0253 - val\_loss: 6.2623 - val\_accuracy: 0.0000e+00  
Epoch 40/200  
16/16 [=====] - 0s 6ms/step - loss: 3.8654 -  
accuracy: 0.0380 - val\_loss: 6.3194 - val\_accuracy: 0.0000e+00  
Epoch 41/200  
16/16 [=====] - 0s 6ms/step - loss: 3.8811 -  
accuracy: 0.0633 - val\_loss: 6.3494 - val\_accuracy: 0.0000e+00  
Epoch 42/200  
16/16 [=====] - 0s 6ms/step - loss: 3.7906 -  
accuracy: 0.0633 - val\_loss: 6.3938 - val\_accuracy: 0.0000e+00  
Epoch 43/200  
16/16 [=====] - 0s 6ms/step - loss: 3.7915 -  
accuracy: 0.0506 - val\_loss: 6.4741 - val\_accuracy: 0.0000e+00  
Epoch 44/200  
16/16 [=====] - 0s 7ms/step - loss: 3.8477 -  
accuracy: 0.0253 - val\_loss: 6.5023 - val\_accuracy: 0.0000e+00  
Epoch 45/200  
16/16 [=====] - 0s 6ms/step - loss: 3.8572 -  
accuracy: 0.0506 - val\_loss: 6.4643 - val\_accuracy: 0.0000e+00  
Epoch 46/200  
16/16 [=====] - 0s 7ms/step - loss: 3.8881 -  
accuracy: 0.0000e+00 - val\_loss: 6.4410 - val\_accuracy: 0.0000e+00  
Epoch 47/200  
16/16 [=====] - 0s 6ms/step - loss: 3.8457 -  
accuracy: 0.0633 - val\_loss: 6.4991 - val\_accuracy: 0.0000e+00  
Epoch 48/200  
16/16 [=====] - 0s 6ms/step - loss: 3.7634 -  
accuracy: 0.0380 - val\_loss: 6.6013 - val\_accuracy: 0.0000e+00  
Epoch 49/200  
16/16 [=====] - 0s 6ms/step - loss: 3.7669 -  
accuracy: 0.0506 - val\_loss: 6.6865 - val\_accuracy: 0.0000e+00  
Epoch 50/200  
16/16 [=====] - 0s 6ms/step - loss: 3.8488 -  
accuracy: 0.0253 - val\_loss: 6.7247 - val\_accuracy: 0.0000e+00

Epoch 51/200  
16/16 [=====] - 0s 6ms/step - loss: 3.6986 -  
accuracy: 0.0506 - val\_loss: 6.7667 - val\_accuracy: 0.0000e+00  
Epoch 52/200  
16/16 [=====] - 0s 7ms/step - loss: 3.8289 -  
accuracy: 0.0506 - val\_loss: 6.7924 - val\_accuracy: 0.0000e+00  
Epoch 53/200  
16/16 [=====] - 0s 7ms/step - loss: 3.8348 -  
accuracy: 0.0380 - val\_loss: 6.8380 - val\_accuracy: 0.0000e+00  
Epoch 54/200  
16/16 [=====] - 0s 7ms/step - loss: 3.7555 -  
accuracy: 0.0633 - val\_loss: 6.8273 - val\_accuracy: 0.0000e+00  
Epoch 55/200  
16/16 [=====] - 0s 6ms/step - loss: 3.8211 -  
accuracy: 0.0253 - val\_loss: 6.8589 - val\_accuracy: 0.0000e+00  
Epoch 56/200  
16/16 [=====] - 0s 7ms/step - loss: 3.7346 -  
accuracy: 0.0380 - val\_loss: 6.8282 - val\_accuracy: 0.0000e+00  
Epoch 57/200  
16/16 [=====] - 0s 6ms/step - loss: 3.7306 -  
accuracy: 0.0380 - val\_loss: 6.8364 - val\_accuracy: 0.0000e+00  
Epoch 58/200  
16/16 [=====] - 0s 7ms/step - loss: 3.7669 -  
accuracy: 0.0380 - val\_loss: 6.8876 - val\_accuracy: 0.0000e+00  
Epoch 59/200  
16/16 [=====] - 0s 7ms/step - loss: 3.7427 -  
accuracy: 0.0253 - val\_loss: 6.8988 - val\_accuracy: 0.0000e+00  
Epoch 60/200  
16/16 [=====] - 0s 7ms/step - loss: 3.7134 -  
accuracy: 0.0380 - val\_loss: 6.9321 - val\_accuracy: 0.0000e+00  
Epoch 61/200  
16/16 [=====] - 0s 8ms/step - loss: 3.7310 -  
accuracy: 0.0633 - val\_loss: 6.9782 - val\_accuracy: 0.0000e+00  
Epoch 62/200  
16/16 [=====] - 0s 6ms/step - loss: 3.6526 -  
accuracy: 0.0253 - val\_loss: 7.0347 - val\_accuracy: 0.0000e+00  
Epoch 63/200  
16/16 [=====] - 0s 6ms/step - loss: 3.6816 -  
accuracy: 0.0380 - val\_loss: 7.0788 - val\_accuracy: 0.0000e+00  
Epoch 64/200  
16/16 [=====] - 0s 7ms/step - loss: 3.7073 -  
accuracy: 0.0380 - val\_loss: 7.1136 - val\_accuracy: 0.0000e+00  
Epoch 65/200  
16/16 [=====] - 0s 6ms/step - loss: 3.7236 -  
accuracy: 0.0633 - val\_loss: 7.1250 - val\_accuracy: 0.0000e+00  
Epoch 66/200  
16/16 [=====] - 0s 6ms/step - loss: 3.7362 -  
accuracy: 0.0380 - val\_loss: 7.0440 - val\_accuracy: 0.0000e+00  
Epoch 67/200

16/16 [=====] - 0s 6ms/step - loss: 3.6809 -  
accuracy: 0.0127 - val\_loss: 7.0623 - val\_accuracy: 0.0000e+00  
Epoch 68/200  
16/16 [=====] - 0s 7ms/step - loss: 3.6605 -  
accuracy: 0.0506 - val\_loss: 7.1189 - val\_accuracy: 0.0000e+00  
Epoch 69/200  
16/16 [=====] - 0s 7ms/step - loss: 3.6868 -  
accuracy: 0.0380 - val\_loss: 7.1779 - val\_accuracy: 0.0000e+00  
Epoch 70/200  
16/16 [=====] - 0s 6ms/step - loss: 3.6375 -  
accuracy: 0.0380 - val\_loss: 7.2291 - val\_accuracy: 0.0000e+00  
Epoch 71/200  
16/16 [=====] - 0s 8ms/step - loss: 3.6863 -  
accuracy: 0.0127 - val\_loss: 7.2946 - val\_accuracy: 0.0000e+00  
Epoch 72/200  
16/16 [=====] - 0s 7ms/step - loss: 3.7272 -  
accuracy: 0.0380 - val\_loss: 7.3100 - val\_accuracy: 0.0000e+00  
Epoch 73/200  
16/16 [=====] - 0s 6ms/step - loss: 3.7364 -  
accuracy: 0.0380 - val\_loss: 7.2698 - val\_accuracy: 0.0000e+00  
Epoch 74/200  
16/16 [=====] - 0s 7ms/step - loss: 3.7268 -  
accuracy: 0.0253 - val\_loss: 7.2366 - val\_accuracy: 0.0000e+00  
Epoch 75/200  
16/16 [=====] - 0s 7ms/step - loss: 3.6688 -  
accuracy: 0.0127 - val\_loss: 7.2721 - val\_accuracy: 0.0000e+00  
Epoch 76/200  
16/16 [=====] - 0s 7ms/step - loss: 3.6519 -  
accuracy: 0.0506 - val\_loss: 7.3268 - val\_accuracy: 0.0000e+00  
Epoch 77/200  
16/16 [=====] - 0s 7ms/step - loss: 3.6498 -  
accuracy: 0.0506 - val\_loss: 7.3228 - val\_accuracy: 0.0000e+00  
Epoch 78/200  
16/16 [=====] - 0s 6ms/step - loss: 3.6291 -  
accuracy: 0.0633 - val\_loss: 7.3266 - val\_accuracy: 0.0000e+00  
Epoch 79/200  
16/16 [=====] - 0s 7ms/step - loss: 3.6396 -  
accuracy: 0.0759 - val\_loss: 7.4171 - val\_accuracy: 0.0000e+00  
Epoch 80/200  
16/16 [=====] - 0s 6ms/step - loss: 3.6692 -  
accuracy: 0.0506 - val\_loss: 7.4707 - val\_accuracy: 0.0000e+00  
Epoch 81/200  
16/16 [=====] - 0s 7ms/step - loss: 3.6515 -  
accuracy: 0.0380 - val\_loss: 7.4600 - val\_accuracy: 0.0000e+00  
Epoch 82/200  
16/16 [=====] - 0s 7ms/step - loss: 3.6552 -  
accuracy: 0.0759 - val\_loss: 7.5256 - val\_accuracy: 0.0000e+00  
Epoch 83/200  
16/16 [=====] - 0s 7ms/step - loss: 3.6407 -  
accuracy: 0.0633 - val\_loss: 7.5216 - val\_accuracy: 0.0000e+00

Epoch 84/200  
16/16 [=====] - 0s 6ms/step - loss: 3.6674 -  
accuracy: 0.0506 - val\_loss: 7.5314 - val\_accuracy: 0.0000e+00  
Epoch 85/200  
16/16 [=====] - 0s 6ms/step - loss: 3.6047 -  
accuracy: 0.0380 - val\_loss: 7.5201 - val\_accuracy: 0.0000e+00  
Epoch 86/200  
16/16 [=====] - 0s 7ms/step - loss: 3.7258 -  
accuracy: 0.0380 - val\_loss: 7.4754 - val\_accuracy: 0.0000e+00  
Epoch 87/200  
16/16 [=====] - 0s 7ms/step - loss: 3.5990 -  
accuracy: 0.0380 - val\_loss: 7.4994 - val\_accuracy: 0.0000e+00  
Epoch 88/200  
16/16 [=====] - 0s 7ms/step - loss: 3.6420 -  
accuracy: 0.0506 - val\_loss: 7.5758 - val\_accuracy: 0.0000e+00  
Epoch 89/200  
16/16 [=====] - 0s 7ms/step - loss: 3.6542 -  
accuracy: 0.0759 - val\_loss: 7.6303 - val\_accuracy: 0.0000e+00  
Epoch 90/200  
16/16 [=====] - 0s 7ms/step - loss: 3.7078 -  
accuracy: 0.0506 - val\_loss: 7.6832 - val\_accuracy: 0.0000e+00  
Epoch 91/200  
16/16 [=====] - 0s 6ms/step - loss: 3.6241 -  
accuracy: 0.0380 - val\_loss: 7.6729 - val\_accuracy: 0.0000e+00  
Epoch 92/200  
16/16 [=====] - 0s 6ms/step - loss: 3.6395 -  
accuracy: 0.0506 - val\_loss: 7.7216 - val\_accuracy: 0.0000e+00  
Epoch 93/200  
16/16 [=====] - 0s 7ms/step - loss: 3.6694 -  
accuracy: 0.0633 - val\_loss: 7.7149 - val\_accuracy: 0.0000e+00  
Epoch 94/200  
16/16 [=====] - 0s 7ms/step - loss: 3.6064 -  
accuracy: 0.0253 - val\_loss: 7.6866 - val\_accuracy: 0.0000e+00  
Epoch 95/200  
16/16 [=====] - 0s 6ms/step - loss: 3.6094 -  
accuracy: 0.0380 - val\_loss: 7.6826 - val\_accuracy: 0.0000e+00  
Epoch 96/200  
16/16 [=====] - 0s 7ms/step - loss: 3.5929 -  
accuracy: 0.0380 - val\_loss: 7.7238 - val\_accuracy: 0.0000e+00  
Epoch 97/200  
16/16 [=====] - 0s 8ms/step - loss: 3.5672 -  
accuracy: 0.0380 - val\_loss: 7.7170 - val\_accuracy: 0.0000e+00  
Epoch 98/200  
16/16 [=====] - 0s 7ms/step - loss: 3.6566 -  
accuracy: 0.0633 - val\_loss: 7.7288 - val\_accuracy: 0.0000e+00  
Epoch 99/200  
16/16 [=====] - 0s 7ms/step - loss: 3.5655 -  
accuracy: 0.0633 - val\_loss: 7.7282 - val\_accuracy: 0.0000e+00  
Epoch 100/200

16/16 [=====] - 0s 7ms/step - loss: 3.6425 -  
accuracy: 0.0253 - val\_loss: 7.6570 - val\_accuracy: 0.0000e+00  
Epoch 101/200  
16/16 [=====] - 0s 8ms/step - loss: 3.5555 -  
accuracy: 0.0759 - val\_loss: 7.7016 - val\_accuracy: 0.0000e+00  
Epoch 102/200  
16/16 [=====] - 0s 8ms/step - loss: 3.6806 -  
accuracy: 0.0380 - val\_loss: 7.7608 - val\_accuracy: 0.0000e+00  
Epoch 103/200  
16/16 [=====] - 0s 7ms/step - loss: 3.6604 -  
accuracy: 0.0506 - val\_loss: 7.7438 - val\_accuracy: 0.0000e+00  
Epoch 104/200  
16/16 [=====] - 0s 7ms/step - loss: 3.6105 -  
accuracy: 0.0886 - val\_loss: 7.7284 - val\_accuracy: 0.0000e+00  
Epoch 105/200  
16/16 [=====] - 0s 7ms/step - loss: 3.5698 -  
accuracy: 0.0633 - val\_loss: 7.8078 - val\_accuracy: 0.0000e+00  
Epoch 106/200  
16/16 [=====] - 0s 8ms/step - loss: 3.5576 -  
accuracy: 0.0633 - val\_loss: 7.8680 - val\_accuracy: 0.0000e+00  
Epoch 107/200  
16/16 [=====] - 0s 8ms/step - loss: 3.6806 -  
accuracy: 0.0127 - val\_loss: 7.8066 - val\_accuracy: 0.0000e+00  
Epoch 108/200  
16/16 [=====] - 0s 8ms/step - loss: 3.6002 -  
accuracy: 0.0253 - val\_loss: 7.7642 - val\_accuracy: 0.0000e+00  
Epoch 109/200  
16/16 [=====] - 0s 7ms/step - loss: 3.6821 -  
accuracy: 0.0253 - val\_loss: 7.7412 - val\_accuracy: 0.0000e+00  
Epoch 110/200  
16/16 [=====] - 0s 7ms/step - loss: 3.5941 -  
accuracy: 0.0380 - val\_loss: 7.7671 - val\_accuracy: 0.0000e+00  
Epoch 111/200  
16/16 [=====] - 0s 15ms/step - loss: 3.5221 -  
accuracy: 0.0380 - val\_loss: 7.9108 - val\_accuracy: 0.0000e+00  
Epoch 112/200  
16/16 [=====] - 0s 10ms/step - loss: 3.6386 -  
accuracy: 0.0253 - val\_loss: 7.9651 - val\_accuracy: 0.0000e+00  
Epoch 113/200  
16/16 [=====] - 0s 10ms/step - loss: 3.6264 -  
accuracy: 0.0506 - val\_loss: 7.9575 - val\_accuracy: 0.0000e+00  
Epoch 114/200  
16/16 [=====] - 0s 8ms/step - loss: 3.6399 -  
accuracy: 0.0380 - val\_loss: 7.9473 - val\_accuracy: 0.0000e+00  
Epoch 115/200  
16/16 [=====] - 0s 7ms/step - loss: 3.6731 -  
accuracy: 0.0253 - val\_loss: 7.9006 - val\_accuracy: 0.0000e+00  
Epoch 116/200  
16/16 [=====] - 0s 8ms/step - loss: 3.6212 -  
accuracy: 0.0380 - val\_loss: 7.7718 - val\_accuracy: 0.0000e+00

Epoch 117/200  
16/16 [=====] - 0s 8ms/step - loss: 3.5361 -  
accuracy: 0.0633 - val\_loss: 7.8377 - val\_accuracy: 0.0000e+00

Epoch 118/200  
16/16 [=====] - 0s 8ms/step - loss: 3.6538 -  
accuracy: 0.0759 - val\_loss: 7.9125 - val\_accuracy: 0.0000e+00

Epoch 119/200  
16/16 [=====] - 0s 7ms/step - loss: 3.5419 -  
accuracy: 0.0759 - val\_loss: 8.0018 - val\_accuracy: 0.0000e+00

Epoch 120/200  
16/16 [=====] - 0s 7ms/step - loss: 3.6459 -  
accuracy: 0.0253 - val\_loss: 7.9855 - val\_accuracy: 0.0000e+00

Epoch 121/200  
16/16 [=====] - 0s 8ms/step - loss: 3.5856 -  
accuracy: 0.0759 - val\_loss: 7.9900 - val\_accuracy: 0.0000e+00

Epoch 122/200  
16/16 [=====] - 0s 7ms/step - loss: 3.6312 -  
accuracy: 0.0506 - val\_loss: 7.9656 - val\_accuracy: 0.0000e+00

Epoch 123/200  
16/16 [=====] - 0s 7ms/step - loss: 3.5613 -  
accuracy: 0.0633 - val\_loss: 8.0139 - val\_accuracy: 0.0000e+00

Epoch 124/200  
16/16 [=====] - 0s 7ms/step - loss: 3.5619 -  
accuracy: 0.0633 - val\_loss: 8.0268 - val\_accuracy: 0.0000e+00

Epoch 125/200  
16/16 [=====] - 0s 8ms/step - loss: 3.6305 -  
accuracy: 0.0380 - val\_loss: 8.0060 - val\_accuracy: 0.0000e+00

Epoch 126/200  
16/16 [=====] - 0s 7ms/step - loss: 3.5875 -  
accuracy: 0.0759 - val\_loss: 8.0208 - val\_accuracy: 0.0000e+00

Epoch 127/200  
16/16 [=====] - 0s 7ms/step - loss: 3.5398 -  
accuracy: 0.0253 - val\_loss: 8.0509 - val\_accuracy: 0.0000e+00

Epoch 128/200  
16/16 [=====] - 0s 7ms/step - loss: 3.5707 -  
accuracy: 0.0633 - val\_loss: 8.0631 - val\_accuracy: 0.0000e+00

Epoch 129/200  
16/16 [=====] - 0s 7ms/step - loss: 3.5681 -  
accuracy: 0.0506 - val\_loss: 8.0579 - val\_accuracy: 0.0000e+00

Epoch 130/200  
16/16 [=====] - 0s 10ms/step - loss: 3.5869 -  
accuracy: 0.0253 - val\_loss: 8.0230 - val\_accuracy: 0.0000e+00

Epoch 131/200  
16/16 [=====] - 0s 10ms/step - loss: 3.6695 -  
accuracy: 0.0506 - val\_loss: 7.9920 - val\_accuracy: 0.0000e+00

Epoch 132/200  
16/16 [=====] - 0s 8ms/step - loss: 3.5999 -  
accuracy: 0.0380 - val\_loss: 7.9790 - val\_accuracy: 0.0000e+00

Epoch 133/200

16/16 [=====] - 0s 8ms/step - loss: 3.6066 -  
accuracy: 0.0633 - val\_loss: 7.9927 - val\_accuracy: 0.0000e+00  
Epoch 134/200  
16/16 [=====] - 0s 9ms/step - loss: 3.5299 -  
accuracy: 0.0633 - val\_loss: 8.0784 - val\_accuracy: 0.0000e+00  
Epoch 135/200  
16/16 [=====] - 0s 8ms/step - loss: 3.5903 -  
accuracy: 0.0759 - val\_loss: 8.0919 - val\_accuracy: 0.0000e+00  
Epoch 136/200  
16/16 [=====] - 0s 8ms/step - loss: 3.6375 -  
accuracy: 0.0380 - val\_loss: 8.0198 - val\_accuracy: 0.0000e+00  
Epoch 137/200  
16/16 [=====] - 0s 9ms/step - loss: 3.5877 -  
accuracy: 0.0759 - val\_loss: 8.1310 - val\_accuracy: 0.0000e+00  
Epoch 138/200  
16/16 [=====] - 0s 9ms/step - loss: 3.5468 -  
accuracy: 0.0759 - val\_loss: 8.2049 - val\_accuracy: 0.0000e+00  
Epoch 139/200  
16/16 [=====] - 0s 9ms/step - loss: 3.5546 -  
accuracy: 0.0380 - val\_loss: 8.2701 - val\_accuracy: 0.0000e+00  
Epoch 140/200  
16/16 [=====] - 0s 9ms/step - loss: 3.5713 -  
accuracy: 0.0253 - val\_loss: 8.1958 - val\_accuracy: 0.0000e+00  
Epoch 141/200  
16/16 [=====] - 0s 9ms/step - loss: 3.5931 -  
accuracy: 0.0506 - val\_loss: 8.1582 - val\_accuracy: 0.0000e+00  
Epoch 142/200  
16/16 [=====] - 0s 8ms/step - loss: 3.5720 -  
accuracy: 0.0253 - val\_loss: 8.1992 - val\_accuracy: 0.0000e+00  
Epoch 143/200  
16/16 [=====] - 0s 8ms/step - loss: 3.6100 -  
accuracy: 0.0633 - val\_loss: 8.2473 - val\_accuracy: 0.0000e+00  
Epoch 144/200  
16/16 [=====] - 0s 8ms/step - loss: 3.5741 -  
accuracy: 0.0506 - val\_loss: 8.2527 - val\_accuracy: 0.0000e+00  
Epoch 145/200  
16/16 [=====] - 0s 8ms/step - loss: 3.5002 -  
accuracy: 0.1013 - val\_loss: 8.2412 - val\_accuracy: 0.0000e+00  
Epoch 146/200  
16/16 [=====] - 0s 8ms/step - loss: 3.5573 -  
accuracy: 0.0759 - val\_loss: 8.2816 - val\_accuracy: 0.0000e+00  
Epoch 147/200  
16/16 [=====] - 0s 8ms/step - loss: 3.5442 -  
accuracy: 0.0380 - val\_loss: 8.2879 - val\_accuracy: 0.0000e+00  
Epoch 148/200  
16/16 [=====] - 0s 7ms/step - loss: 3.5042 -  
accuracy: 0.0506 - val\_loss: 8.2869 - val\_accuracy: 0.0000e+00  
Epoch 149/200  
16/16 [=====] - 0s 7ms/step - loss: 3.6053 -  
accuracy: 0.0380 - val\_loss: 8.2872 - val\_accuracy: 0.0000e+00

Epoch 150/200  
16/16 [=====] - 0s 7ms/step - loss: 3.5594 -  
accuracy: 0.0506 - val\_loss: 8.1029 - val\_accuracy: 0.0000e+00  
Epoch 151/200  
16/16 [=====] - 0s 7ms/step - loss: 3.5559 -  
accuracy: 0.0380 - val\_loss: 8.0675 - val\_accuracy: 0.0000e+00  
Epoch 152/200  
16/16 [=====] - 0s 7ms/step - loss: 3.5548 -  
accuracy: 0.0380 - val\_loss: 8.0971 - val\_accuracy: 0.0000e+00  
Epoch 153/200  
16/16 [=====] - 0s 7ms/step - loss: 3.5668 -  
accuracy: 0.0633 - val\_loss: 8.2042 - val\_accuracy: 0.0000e+00  
Epoch 154/200  
16/16 [=====] - 0s 8ms/step - loss: 3.6444 -  
accuracy: 0.0253 - val\_loss: 8.2175 - val\_accuracy: 0.0000e+00  
Epoch 155/200  
16/16 [=====] - 0s 7ms/step - loss: 3.5047 -  
accuracy: 0.0759 - val\_loss: 8.2250 - val\_accuracy: 0.0000e+00  
Epoch 156/200  
16/16 [=====] - 0s 8ms/step - loss: 3.5009 -  
accuracy: 0.0759 - val\_loss: 8.2618 - val\_accuracy: 0.0000e+00  
Epoch 157/200  
16/16 [=====] - 0s 8ms/step - loss: 3.5494 -  
accuracy: 0.0633 - val\_loss: 8.3067 - val\_accuracy: 0.0000e+00  
Epoch 158/200  
16/16 [=====] - 0s 7ms/step - loss: 3.5398 -  
accuracy: 0.0380 - val\_loss: 8.3330 - val\_accuracy: 0.0000e+00  
Epoch 159/200  
16/16 [=====] - 0s 8ms/step - loss: 3.6495 -  
accuracy: 0.0380 - val\_loss: 8.2990 - val\_accuracy: 0.0000e+00  
Epoch 160/200  
16/16 [=====] - 0s 8ms/step - loss: 3.5734 -  
accuracy: 0.0380 - val\_loss: 8.2463 - val\_accuracy: 0.0000e+00  
Epoch 161/200  
16/16 [=====] - 0s 7ms/step - loss: 3.5832 -  
accuracy: 0.0506 - val\_loss: 8.2322 - val\_accuracy: 0.0000e+00  
Epoch 162/200  
16/16 [=====] - 0s 8ms/step - loss: 3.6338 -  
accuracy: 0.0506 - val\_loss: 8.1706 - val\_accuracy: 0.0000e+00  
Epoch 163/200  
16/16 [=====] - 0s 7ms/step - loss: 3.5460 -  
accuracy: 0.0759 - val\_loss: 8.2506 - val\_accuracy: 0.0000e+00  
Epoch 164/200  
16/16 [=====] - 0s 7ms/step - loss: 3.5830 -  
accuracy: 0.0253 - val\_loss: 8.3316 - val\_accuracy: 0.0000e+00  
Epoch 165/200  
16/16 [=====] - 0s 7ms/step - loss: 3.4508 -  
accuracy: 0.0886 - val\_loss: 8.3244 - val\_accuracy: 0.0000e+00  
Epoch 166/200

16/16 [=====] - 0s 7ms/step - loss: 3.5427 -  
accuracy: 0.0380 - val\_loss: 8.3453 - val\_accuracy: 0.0000e+00  
Epoch 167/200  
16/16 [=====] - 0s 7ms/step - loss: 3.5629 -  
accuracy: 0.0633 - val\_loss: 8.2850 - val\_accuracy: 0.0000e+00  
Epoch 168/200  
16/16 [=====] - 0s 7ms/step - loss: 3.5260 -  
accuracy: 0.0759 - val\_loss: 8.2891 - val\_accuracy: 0.0000e+00  
Epoch 169/200  
16/16 [=====] - 0s 8ms/step - loss: 3.5255 -  
accuracy: 0.0633 - val\_loss: 8.3601 - val\_accuracy: 0.0000e+00  
Epoch 170/200  
16/16 [=====] - 0s 8ms/step - loss: 3.5611 -  
accuracy: 0.0253 - val\_loss: 8.3679 - val\_accuracy: 0.0000e+00  
Epoch 171/200  
16/16 [=====] - 0s 7ms/step - loss: 3.6115 -  
accuracy: 0.0633 - val\_loss: 8.3664 - val\_accuracy: 0.0000e+00  
Epoch 172/200  
16/16 [=====] - 0s 7ms/step - loss: 3.5947 -  
accuracy: 0.0633 - val\_loss: 8.4590 - val\_accuracy: 0.0000e+00  
Epoch 173/200  
16/16 [=====] - 0s 8ms/step - loss: 3.5672 -  
accuracy: 0.0380 - val\_loss: 8.5419 - val\_accuracy: 0.0000e+00  
Epoch 174/200  
16/16 [=====] - 0s 7ms/step - loss: 3.4603 -  
accuracy: 0.0886 - val\_loss: 8.6118 - val\_accuracy: 0.0000e+00  
Epoch 175/200  
16/16 [=====] - 0s 7ms/step - loss: 3.5399 -  
accuracy: 0.0633 - val\_loss: 8.6556 - val\_accuracy: 0.0000e+00  
Epoch 176/200  
16/16 [=====] - 0s 7ms/step - loss: 3.5560 -  
accuracy: 0.0633 - val\_loss: 8.5505 - val\_accuracy: 0.0000e+00  
Epoch 177/200  
16/16 [=====] - 0s 7ms/step - loss: 3.6511 -  
accuracy: 0.0127 - val\_loss: 8.4278 - val\_accuracy: 0.0000e+00  
Epoch 178/200  
16/16 [=====] - 0s 8ms/step - loss: 3.4947 -  
accuracy: 0.0633 - val\_loss: 8.3929 - val\_accuracy: 0.0000e+00  
Epoch 179/200  
16/16 [=====] - 0s 7ms/step - loss: 3.5810 -  
accuracy: 0.0380 - val\_loss: 8.3626 - val\_accuracy: 0.0000e+00  
Epoch 180/200  
16/16 [=====] - 0s 7ms/step - loss: 3.5096 -  
accuracy: 0.0253 - val\_loss: 8.4039 - val\_accuracy: 0.0000e+00  
Epoch 181/200  
16/16 [=====] - 0s 6ms/step - loss: 3.6019 -  
accuracy: 0.0253 - val\_loss: 8.3858 - val\_accuracy: 0.0000e+00  
Epoch 182/200  
16/16 [=====] - 0s 6ms/step - loss: 3.5022 -  
accuracy: 0.0633 - val\_loss: 8.4263 - val\_accuracy: 0.0000e+00

Epoch 183/200  
16/16 [=====] - 0s 7ms/step - loss: 3.6043 -  
accuracy: 0.0380 - val\_loss: 8.4046 - val\_accuracy: 0.0000e+00  
Epoch 184/200  
16/16 [=====] - 0s 7ms/step - loss: 3.5665 -  
accuracy: 0.0506 - val\_loss: 8.4146 - val\_accuracy: 0.0000e+00  
Epoch 185/200  
16/16 [=====] - 0s 7ms/step - loss: 3.5818 -  
accuracy: 0.0506 - val\_loss: 8.4513 - val\_accuracy: 0.0000e+00  
Epoch 186/200  
16/16 [=====] - 0s 6ms/step - loss: 3.5109 -  
accuracy: 0.0633 - val\_loss: 8.4705 - val\_accuracy: 0.0000e+00  
Epoch 187/200  
16/16 [=====] - 0s 6ms/step - loss: 3.5299 -  
accuracy: 0.0380 - val\_loss: 8.5629 - val\_accuracy: 0.0000e+00  
Epoch 188/200  
16/16 [=====] - 0s 7ms/step - loss: 3.5079 -  
accuracy: 0.0506 - val\_loss: 8.5488 - val\_accuracy: 0.0000e+00  
Epoch 189/200  
16/16 [=====] - 0s 6ms/step - loss: 3.5709 -  
accuracy: 0.0506 - val\_loss: 8.5863 - val\_accuracy: 0.0000e+00  
Epoch 190/200  
16/16 [=====] - 0s 7ms/step - loss: 3.5054 -  
accuracy: 0.0633 - val\_loss: 8.6166 - val\_accuracy: 0.0000e+00  
Epoch 191/200  
16/16 [=====] - 0s 7ms/step - loss: 3.5320 -  
accuracy: 0.0380 - val\_loss: 8.5865 - val\_accuracy: 0.0000e+00  
Epoch 192/200  
16/16 [=====] - 0s 9ms/step - loss: 3.5725 -  
accuracy: 0.0633 - val\_loss: 8.6253 - val\_accuracy: 0.0000e+00  
Epoch 193/200  
16/16 [=====] - 0s 10ms/step - loss: 3.5349 -  
accuracy: 0.0506 - val\_loss: 8.6352 - val\_accuracy: 0.0000e+00  
Epoch 194/200  
16/16 [=====] - 0s 7ms/step - loss: 3.4939 -  
accuracy: 0.0506 - val\_loss: 8.6933 - val\_accuracy: 0.0000e+00  
Epoch 195/200  
16/16 [=====] - 0s 16ms/step - loss: 3.5275 -  
accuracy: 0.0380 - val\_loss: 8.6858 - val\_accuracy: 0.0000e+00  
Epoch 196/200  
16/16 [=====] - 0s 8ms/step - loss: 3.5149 -  
accuracy: 0.0127 - val\_loss: 8.6606 - val\_accuracy: 0.0000e+00  
Epoch 197/200  
16/16 [=====] - 0s 13ms/step - loss: 3.5085 -  
accuracy: 0.0380 - val\_loss: 8.5981 - val\_accuracy: 0.0000e+00  
Epoch 198/200  
16/16 [=====] - 0s 7ms/step - loss: 3.5904 -  
accuracy: 0.0380 - val\_loss: 8.5845 - val\_accuracy: 0.0000e+00  
Epoch 199/200

```
16/16 [=====] - 0s 7ms/step - loss: 3.6002 -  
accuracy: 0.0380 - val_loss: 8.4937 - val_accuracy: 0.0000e+00  
Epoch 200/200  
16/16 [=====] - 0s 7ms/step - loss: 3.5508 -  
accuracy: 0.0506 - val_loss: 8.5434 - val_accuracy: 0.0000e+00
```

```
In [17]:
```

```
# Save the modelmodel.save('model.counsellor_new')print("Model created")
```

```
INFO:tensorflow:Assets written to: model.counsellor_new\assets
```

```
Model created
```

```
In [ ]:
```

Do Not Copy, Lead City University, Nigeria

## Bio-data

### i. **Personal Data:**

Full name: Lucky Samuel-Ejomafuvwe  
Address: No 124 Apebi Street, Ijebu-Ode, Ogun-State  
E-mail: luckysamuel49@gmail.com  
Phone No: 08050308452, 08109887057  
Date of Birth: 27<sup>th</sup> December 1997  
Nationality: Nigerian

### ii. **Educational Background:**

#### **Educational Institutions Attended with Dates and Qualifications:**

2019: H.N.D Computer Science, (Gateway (ICT) Polytechnic, Ogun-State)  
2016: O.N.D Computer Science (Grace Polytechnic, Omu-Ijebu, Ogun-State)  
2014: WAEC Certificate (Varsity Grammer School, Ijebu-Ode, Ogun-State).

### iii. **Working Experience with Dates.**

July 2015 – Dec 2016 - Grace Polytechnic.  
December 2016 – June 2017 - Ogun State Internal Revenue Service (OGIRS).  
June 2017– March 2018 - Fastlink Telecommunications.  
October 2019 – October 2020 - NYSC. Hubbulah Islamic College.  
November 2020– August 2021 - Dragnet Solutions Limited.  
November 2021– Present - First Tech.

### iv. **Membership of Academic/Professional Bodies**

Nil.

### v. **Conferences Attended and Publications**

Nil.

### **The University Compliance Certification**

This is to certify that this thesis written by Samuel Ejomafuvwe LUCKY with the matriculation number LCU/PG/002201 in the **Department of Computer Science, Faculty of Natural and Applied Sciences**, Lead City University, Ibadan, Oyo State, Nigeria is in full compliance with the approved University Format and Style.

---

**Signature**

---

**Date**

*Do Not Copy, Lead City University, Nigeria*