Developing Afaan Oromo Text Based Chatbot for Traditional Food Recipe Using Machine Learning

By
Takele Wakuma Idato

A Thesis Submitted to The Department of Computer Science and Engineering, School of Electrical Engineering and Computing Presented in Partial Fulfillment of the Requirement for the Degree of Master of Science in Computer Science and Engineering

Office of Graduate Studies
Adama Science and Technology University

June, 2024
Adama, Ethiopia

Developing Afaan Oromo Text Based Chatbot for Traditional Food Recipe Using Machine Learning

By

Takele Wakuma Idato

Advisor

Teklu Urgesa (Ph.D, Associate Prof.)

A Thesis Submitted to the Department of Computer Science and Engineering, School of Electrical Engineering and Computing Presented in Partial Fulfillment of the Requirement for the Degree of Master of Science in Computer Science and Engineering

Office of Graduate Studies

Adama Science and Technology University

June, 2024

Adama, Ethiopia

DECLARATION

I declare that this thesis entitled **"Developing Afaan Oromo Text Based Chatbot for Traditional Food Recipe Using Machine Learning"** is my own work and has not been submitted to any university for similar purpose. The references used in this thesis are duly recognized by proper citations.

Name of Student:	**Signature:**	**Date:**
Takele Wakuma Idato		June 5^{th}, 2024

RECOMMENDATION

I, the major advisor of this research, hereby certify that I have closely advised the student while preparing this thesis and read the draft thesis entitled **“Developing Afaan Oromo Text Based Chatbot for Traditional Food Recipe Using Machine Learning”** prepared under my guidance by Takele Wakuma Idato. Therefore, I recommend the submission of the thesis to the department for further review and evaluation.

Name of Advisor:	**Signature:**	**Date:**
Teklu Urgesa (Ph.D, Associate Prof.)		

APPROVAL PAGE

I/we hereby certify that the recommendation and suggestion given by the review committee are appropriately incorporated into the final thesis/dissertation entitled **“Developing Afaan Oromo Text Based Chatbot for Traditional Food Recipe Using Machine Learning”** by Takele Wakuma Idato.

Name of Advisor:	**Signature:**	**Date:**
Teklu Urgesa (Ph.D, Associate Prof.)		

Approval of Board of Reviewers

We, the undersigned, members of the Board of Reviewers of the thesis open defense by Takele Wakuma have read and evaluated the thesis entitled **"Developing Afaan Oromo Text Based Chatbot for Traditional Food Recipe Using Machine Learning"** and assessed the understanding of the candidate about the research. This is, therefore, to certify that the thesis is accepted and we recommend the implementation of the research **"Developing Afaan Oromo Text Based Chatbot for Traditional Food Recipe Using Machine Learning".**

Chairperson	Signature	Date
Reviewer 1	Signature	Date
Reviewer 2	Signature	Date

Final approval and acceptance of the thesis is contingent upon submission of its final copy to the Office of Postgraduate Studies (OPGS) through the Department Graduate Council (DGC) and School Graduate Committee (SGC).

Department Head	Signature	Date
School Dean	Signature	Date
Office of Postgraduate Studies, Dean	Signature	Date

ACKNOWLEDGEMENT

First of all, I would like to thank God for his peace and blessings upon my life and for giving me the ability to successfully complete this thesis. I would like also to express my heartfelt gratitude to my advisor, Teklu Urgessa (PhD, Associate Prof.) for his continuous support and encouragement.

Finally, I would like to express my deepest gratitude to my caring, loving, and supportive family. During the course of thesis writing, their unconditional love and care were much appreciated and noted.

TABLE OF CONTENTS

ACRONYMS AND ABBREVIATIONS

AI	Artificial Intelligence
AIML	Artificial Intelligence Markup Language
Bi-LSTM	Bi-directional Long Short-Term Memory
BoW	Bag-of-Words
DNN	Deep Neural Networks
ELIZA	Elisabeth
JSON	JavaScript Object Notation
LSTM	Long Short-Term Memory
ML	Machine Learning
NLP	Natural Language Processing
RNN	Recurrent Neural Network
SVM	Support Vector Machines
Word2Vec	Word to vector

LIST OF FIGURES

LIST OF TABLES

ABSTRACT

Cooking traditional foods, such as Chechebsa and Cuukkoo, presents challenges for both beginners and experienced cooks due to the intricate details and nuances inherent in traditional recipes. To address this, our study proposes the development of a text-based chatbot for assisting individuals in preparing traditional Afaan Oromo recipes using machine learning techniques. The primary problem addressed by this research is the lack of accessible and user-friendly resources for individuals seeking guidance and assistance in cooking traditional dishes in Afaan Oromo. Our aim is to design and implement text-based chatbot capable of understanding user queries, providing step-by-step cooking instructions, and offering personalized cooking advice. The methodology employed in this study involves systematic data collection from various reliable sources, annotation, and curation of a representative dataset of Afaan Oromo food recipes, followed by extensive data preprocessing and model building using machine learning techniques. Experimental scenarios include training and evaluating the chatbot model on the curated dataset to assess its effectiveness, accuracy, and user satisfaction. Key findings indicate that the developed chatbot model exhibits promising performance, achieving a test accuracy of 80%, in assisting individuals with various levels of cooking experience in preparing traditional Afaan Oromo recipes. The implications for the future involve potential applications of the chatbot in culinary education, cultural preservation, and promoting awareness and appreciation of Afaan Oromo cuisine and cultural heritage. Overall, this study contributes to the advancement of natural language processing and machine learning research while addressing practical challenges in the domain of traditional food preparation and cultural preservation.

CHAPTER ONE
INTRODUCTION

1.1. Background of the Study

Chatbots are intelligent conversational computer programs that mimic human conversation in its natural form. Usually, chatbots (also known as a dialogue system or conversational agents) take natural language text as input, and produce the most relevant output to the user input sentence (Jia, 2003). Chatbots can also be defined as "online human-computer dialogue system(s) with natural language". Chatbots constitute therefore an automated dialogue system, that can attend to thousands of potential users at once (Cahn, 2017; Caldarini, Jaf, & McGarry, 2022).

Dialogue systems have come a long way since their inception in the 1960s. Both the hardware and software aspects of computer science, and natural language processing and machine learning techniques have developed tremendously. Thanks to the advancement of these emerging technologies, chatbots have evolved from systems that generate machine-like responses to human- like agents capable of developing long-term relationships with users. Among the most famous early chatbot implementations are ELIZA and PARRY. Modern chatbots include Apple's Siri and Amazon's Alexa, and Microsoft's XiaoIce (Xufei, 2021).

Chatbots are currently applied to a variety of different fields and applications, spanning from education to e-commerce, encompassing healthcare and entertainment. Therefore, chatbots can provide both support in different fields as well as entertainment to users. Chatbots appear, in fact, to be more engaging to the user than the static Frequently Asked Questions (FAQ) page of a website. At the same time, chatbots can simultaneously assist multiple users, thus resulting more productive and less expensive compared to human customer supports services. In addition to support and assistance to customers, chatbots can be used for providing entertainment and companionship for the end user (Caldarini, Jaf, & McGarry, 2022).

There are well established assistance virtual agents (conversational AI bots) in different countries as standards. However, they are not fully compatible and customizable to our country mainly due to various reasons such as language variation and contexts of the required standards. One of the biggest challenges to effectively utilising AI based chatbots in African languages is lack of Natural Language Processing (NLP) resources. In fact, native African languages are known for their low-resourced in the NLP research community. Africa has over 2000 languages and yet these are among the least represented in NLP research.

Answering various queries about traditional food recipes in real-time holds significant importance in today's fast-paced digital age, catering to the evolving needs and preferences of individuals seeking culinary guidance and cultural immersion. With the proliferation of digital platforms and the increasing accessibility of information, users expect instantaneous responses to their queries, especially regarding traditional cuisines. Real-time assistance in recipe queries enables users to overcome challenges such as time constraints and lack of culinary expertise, empowering them to explore diverse cultural dishes from the comfort of their homes. According to a study by Lai et al. (2017), real-time recipe assistance enhances user engagement and satisfaction by providing personalized and timely guidance, thereby facilitating seamless culinary experiences. Moreover, addressing queries about traditional food recipes in real-time fosters cultural appreciation and preservation, as users can readily access authentic recipes and cooking techniques, fostering a deeper understanding and connection to diverse culinary traditions (Wang et al., 2020). Additionally, real-time recipe assistance contributes to the democratization of culinary knowledge, enabling individuals of varying skill levels and cultural backgrounds to participate in culinary exploration and experimentation. By leveraging technology to deliver prompt and accurate responses to recipe queries, platforms can enhance user experiences, promote cultural exchange, and foster culinary creativity in today's interconnected world.

Therefore, in this study, we propose conversational AI Chatbot model, specifically for Afaan Oromo language, to perform the task of assisting people in cooking by interacting in a more human like way. The proposed Chatbot system aims at better accessibility, personalization, and efficiency of conversational agents that will have the potential to improve the user’s competence based on task oriented conversational system. The system will respond to various queries about traditional food recipe in real-time to consult and inform not just beginners, even people with years of experience in cooking. The proposed Chatbot will be based on machine learning model which will be trained over various documents about traditional food. The required documents are available and published in various sources.

1.2. Statement of the Problem

Cooking traditional foods such as Chechebsa, Kitfo, and others at home requires a mastery of skills that beginners often struggle to attain. Novices may lack an accurate sense of timing and proportion, necessitating guidance from experienced cooks. Even individuals with years of cooking experience may find it challenging to memorize all the details of a recipe, leading to the inconvenience of repeatedly consulting traditional cookery books during the cooking process. Recognizing these challenges inherent in culinary endeavors, this study aims to address the need for a reliable cooking assistant by proposing the design and implementation of an Afaan Oromo text-based chatbot model.

While established virtual assistants exist in various countries, they are not readily adaptable to the linguistic and cultural nuances of Ethiopia, particularly the Afaan Oromo language. Existing conversational AI bots often lack compatibility with the linguistic variations and cultural contexts specific to Ethiopia, hindering their effectiveness as cooking assistants for Afaan Oromo cuisine. According to Turing (1950), the efficacy of an AI system significantly depends on its ability to understand and process natural language in a culturally relevant manner. Therefore, this study seeks to fill this gap by proposing a conversational AI chatbot model tailored specifically for the Afaan Oromo language, enabling interactions that resonate with users in their native language and cultural context.

The development of a specialized conversational AI chatbot for Afaan Oromo culinary assistance is not only a technological challenge but also a cultural imperative. As noted by Floridi (2014), preserving cultural heritage through technology requires innovative solutions that foster engagement and transmission of knowledge. Preserving the cultural heritage embedded within traditional Afaan Oromo food recipes requires innovative approaches that leverage technology to facilitate engagement and transmission of culinary knowledge across generations. By designing a chatbot capable of interacting in Afaan Oromo and providing tailored assistance in cooking traditional dishes, this study aims to contribute to the preservation and promotion of Afaan Oromo cuisine, fostering a deeper appreciation for the cultural significance of traditional foods among Afaan Oromo speakers.

Furthermore, the proposed Afaan Oromo text-based chatbot model represents a step towards empowering individuals to confidently engage with their culinary heritage. By offering real-time guidance and assistance in cooking Afaan Oromo recipes, the chatbot aims to democratize access to culinary expertise, making traditional cooking practices more accessible and enjoyable for individuals of all skill levels. This aligns with the findings of Garcia et al. (2018), who highlighted the potential of AI systems to democratize access to specialized knowledge. This study thus addresses not only the technical challenges of developing a conversational AI system but also the broader societal and cultural implications of preserving and promoting Afaan Oromo culinary traditions through innovative technological solutions.

Therefore, the main purpose of the proposed study is to implement and develop Afaan Oromo text based Chatbot using machine learning techniques.

1.3. Research Questions

The proposed study addresses the following research questions:

RQ1: How can NLP techniques be adapted for accurate understanding in Afaan Oromo?

RQ2: What strategies are effective for collecting and annotating a dataset of Afaan Oromo for Chatbot training?

RQ3: Which machine learning models suitably adapt for building a high-performance Chatbot in Afaan Oromo?

RQ4: What metrics assess the effectiveness of the Chatbot compared to others?

RQ5: How can the Afaan Oromo Chatbot be seamlessly integrated and accepted by its user community?

1.4. Objective of the Study

1.4.1. General Objective

The main objective of this study is to design, develop and implement Afaan Oromo text based Chatbot system to perform the task of assisting people in cooking.

1.4.2. Specific Objectives

To achieve the general objective of the study, the following specific objectives are set:

- To collect published documents about traditional food recipes so as to prepare dataset in appropriate format.
- To design text preprocessing tasks for text normalization, cleaning, tokenization and stop word removal in the context of Afaan Oromo text.
- To develop text preprocessing tasks for text normalization, cleaning, tokenization and stop word removal in the context of Afaan Oromo text.
- To prepare vector representation of Afaan Oromo text.
- To design ML based model architecture for intent prediction.
- To develop ML based model architecture for intent prediction.
- To implement a prototype Chatbot system
- To evaluate the proposed model and perform user acceptance test.
- To compare against other benchmarks

1.5. Scope and Limitations of the Study

The scope of this study is based on a limited Afaan Oromo text corpus, which may lead to a restriction in the number of intents the chatbot can effectively handle. Given the constraints of the available dataset, the chatbot's functionality may be limited to a subset of commonly encountered user queries and recipe-related interactions. Efforts are made to ensure that the selected intents cover a diverse range of user needs and scenarios, prioritizing those most relevant to the preparation of traditional Afaan Oromo dishes.

The proposed chatbot is text-based, focusing solely on processing written text inputs from users. Voice conversation capabilities are not considered within the scope of this study. While voice-enabled chatbots offer enhanced accessibility and convenience, developing voice recognition and synthesis functionalities for Afaan Oromo presents additional technical complexities and resource requirements beyond the scope of the current research. Therefore, the chatbot's interactions are limited to textual inputs and outputs, with the potential for future expansion to incorporate voice-based interactions in subsequent studies or iterations.

Furthermore, it is important to note that the chatbot developed in this study is primarily intended for research purposes and do not serve as a substitute for instructor-led training provided by various training centers across the country. While the chatbot aims to provide assistance and guidance in cooking traditional Afaan Oromo recipes, it does not replace the expertise and hands-on instruction offered by experienced cooking instructors. Instead, the chatbot serves as a supplementary tool to augment users' learning experiences and facilitate their exploration of Afaan Oromo culinary heritage in a digital environment.

The primary limitation of this study is the reliance on a limited Afaan Oromo text corpus for training the chatbot, which may result in a constrained understanding of user queries and recipe-related interactions. The effectiveness and coverage of the chatbot's responses may be hindered by the lack of diverse linguistic and cultural contexts represented in the training data. Consequently, the chatbot's ability to accurately interpret and respond to user inputs may be limited, particularly for less common or specialized queries.

The decision to develop a text-based chatbot excludes voice-based interaction in the proposed chatbot and the reason is mainly stemmed from a desire to streamline functionality and prioritize the core features of the application. Implementing voice recognition technology would have introduced complexity and resource demands, potentially detracting from refining the text-based functionality. Given the linguistic context of Afaan Oromo, which may pose challenges in accurate speech recognition, focusing on text-based interaction ensured greater reliability and efficiency in understanding user queries. Additionally, prioritizing text-based interaction allowed for accelerated development and deployment, aligning with project goals and constraints. While voice-based interaction could offer convenience, its integration would require extensive testing and optimization, which may not be feasible within the research's scope and timeline. Thus, the decision was made to optimize efficiency and user satisfaction by focusing solely on text-based interaction.

Finally, it is important to acknowledge that the chatbot developed in this study has inherent limitations in its ability to replicate the expertise and guidance provided by human instructors. While the chatbot can offer assistance and guidance in cooking traditional Afaan Oromo recipes, it lacks the nuanced understanding, adaptability, and interactive capabilities of human instructors. Users should view the chatbot as a supplementary tool rather than a comprehensive substitute for instructor-led training, recognizing the inherent limitations of technology in replicating human expertise and instruction.

1.6. Significance of the Study

The proposed study holds significant implications for the preservation and promotion of Afaan Oromo culinary heritage, as well as the advancement of technology-mediated cultural preservation initiatives. By developing a text-based chatbot tailored for assisting individuals in cooking traditional Afaan Oromo recipes, the study addresses several key areas of significance:

1.6.1. Cultural Preservation

Afaan Oromo cuisine serves as a tangible expression of cultural identity, transmitting knowledge, traditions, and values across generations. The development of a chatbot dedicated to Afaan Oromo recipes contributes to the preservation and promotion of this rich culinary heritage, ensuring that traditional cooking practices and cultural traditions remain accessible and relevant in the digital age.

1.6.2. Linguistic Inclusivity

The study enhances linguistic inclusivity by providing a digital platform for engaging with Afaan Oromo language and culture. By offering recipe recommendations, cooking instructions, and culinary advice in Afaan Oromo, the chatbot fosters language revitalization efforts and encourages the use and appreciation of Afaan Oromo in everyday contexts, thereby promoting linguistic diversity and empowerment within the Afaan Oromo-speaking community.

1.6.3. Technological Innovation

The development of a text-based chatbot for Afaan Oromo recipes demonstrates the application of cutting-edge technologies, such as machine learning and natural language processing, to address cultural and linguistic preservation challenges. By leveraging these tools, the study exemplifies how technology can be harnessed to bridge the gap between traditional cultural practices and modern digital mediums, fostering innovative solutions for cultural heritage preservation and dissemination.

1.6.4. Accessibility and Convenience

The chatbot offers a convenient and accessible means for individuals to explore and engage with Afaan Oromo culinary heritage. By providing real-time assistance, personalized recommendations, and step-by-step instructions, the chatbot empowers users of varying cooking abilities to confidently prepare traditional Afaan Oromo dishes in their own homes, thereby democratizing access to cultural knowledge and culinary expertise.

1.6.5. Educational Resource

The chatbot serves as an educational resource for individuals seeking to learn about Afaan Oromo cuisine and culinary traditions. Through interactive dialogue and guidance, users can deepen their understanding of Afaan Oromo recipes, ingredients, and cooking techniques, enriching their culinary experiences and fostering a deeper connection to Afaan Oromo cultural heritage.

In summary, the proposed study not only addresses the immediate need for accessible and culturally sensitive cooking assistance in the Afaan Oromo-speaking community but also exemplifies the broader potential of technology to facilitate cultural preservation, linguistic revitalization, and inclusive community engagement. By embracing interdisciplinary approaches and leveraging the power of technology, the study contributes to the ongoing dialogue surrounding the intersection of culture, language, and technology in the digital age.

1.7. Operational Terminologies

Afaan Oromo: The language spoken by the Oromo people, primarily in Ethiopia and parts of Kenya, serving as the primary linguistic medium for the chatbot's interactions and content.

Text-based Chatbot: A conversational agent designed to interact with users through written text, providing assistance, guidance, and information related to traditional Afaan Oromo food recipes.

Machine Learning: A subset of artificial intelligence (AI) that enables systems to learn from data and improve performance over time without being explicitly programmed, crucial for training and optimizing the chatbot's functionality.

Natural Language Processing (NLP): The field of AI concerned with the interaction between computers and human languages, facilitating the understanding, interpretation, and generation of natural language inputs and outputs by the chatbot.

Dataset: A collection of Afaan Oromo text-based food recipes gathered from authentic sources, serving as the training data for the chatbot's machine learning models.

Preprocessing: The initial stage of data preparation, involving tasks such as tokenization, stemming, and lemmatization to transform raw text data into a format suitable for model training.

Model Selection: The process of evaluating and choosing the most appropriate machine learning model architecture and algorithms for the chatbot's specific tasks and objectives.

Evaluation Metrics: Objective measures used to assess the performance and effectiveness of the chatbot, including accuracy, precision, recall, and user satisfaction ratings.

User Interface (UI): The visual and interactive components of the chatbot that enable users to input queries, receive responses, and interact with the system, influencing the overall user experience.

Deployment: The process of making the chatbot accessible to users through web or messaging platforms, ensuring scalability, reliability, and seamless integration with existing digital environments.

1.8. Organization of the Study

This report is organized in six chapters as follows:

Chapter 1: Introduction

The introduction chapter sets the stage for the study by providing an overview of the research topic, which is the development of an Afaan Oromo text-based chatbot for traditional food recipes using machine learning. It discusses the significance of preserving cultural heritage through technology and outlines the research questions, objectives, scope, and limitations of the study. By laying out the foundational elements of the research, this chapter establishes the context and motivation for the subsequent chapters.

Chapter 2: Literature Review

Chapter 2 conducts a thorough review of existing literature relevant to the research topic. It explores previous studies and research findings related to chatbot development, natural language processing, machine learning, and cultural preservation through technology. This chapter synthesizes key insights from the literature, identifies gaps, and highlights areas of interest that inform the research methodology and design.

Chapter 3: Research Methodology

Chapter 3 outlines the research methodology adopted for the study. It provides a detailed description of the systematic approach employed in developing the Afaan Oromo text-based chatbot for traditional food recipes. This chapter covers various aspects such as data collection, preprocessing, model selection, training, evaluation, deployment, and ethical considerations. By delineating the methodological framework, this chapter ensures transparency and reproducibility in the research process.

Chapter 4: Design and Implementation

Chapter 4 delves into the design and implementation phase of the Afaan Oromo text-based chatbot. It discusses the technical aspects of model architecture, implementation, and integration, including the selection of machine learning models and natural language processing techniques. This chapter offers insights into the practical challenges and decisions involved in creating the chatbot prototype, laying the groundwork for subsequent experimental testing.

Chapter 5: Experimental Results and Discussion

Chapter 5 presents the experimental results obtained from training and evaluating the developed chatbot prototype. It analyzes the performance of the chatbot in terms of accuracy, user satisfaction, and cultural relevance based on objective metrics and user feedback. This chapter also includes a detailed discussion of the findings, highlighting strengths, limitations, and areas for improvement in the chatbot's functionality.

Chapter 6: Conclusion and Future Works

Chapter 6 offers a conclusion to the study, summarizing key findings, contributions, and implications for future research. It reflects on the significance of the developed Afaan Oromo text-based chatbot for cultural preservation, linguistic diversity, and technological innovation. Additionally, this chapter outlines potential avenues for future research and enhancements to the chatbot's capabilities, ensuring ongoing progress and relevance in the field of machine learning and cultural heritage preservation.

CHAPTER TWO
LITERATURE REVIEW

2.1. Overview

This chapter is organized in different sections. The first section is an overview on conversational chatbot technology. The second section presents Afaan Oromo language and writing system. The third section is an overview on machine learning approaches and techniques. The fourth section presents the various application areas of machine learning. The fifth section discusses related works.

2.2. Conversational Chatbot Technology

Human-computer interaction (HCI) is a technology that allows communication between users and computers using natural language (Acomb, Bloom, & Dayanidhi, 2007). An automated conversation system (chatbot) is one human machine conversation approach that has been designed to convince humans they are conversing with a human instead of a machine. Chatbots have been widely used to solve small to large problems in several domains, such as in customer services, website helping and education etc. Recent studies predict that 80% of businesses plan to implement chatbots by 2020. The main benefits of using chat-bots for companies is that their customer service processes are automated as the chatbot can answer customers questions about products or services. However, building a smart chatbot is challenging as requires contextual understanding, text entailment and language-understanding technology (Sandbank, Shmueli-Scheuer, Herzig, & Kon, 2018). Therefore, various forms of artificial intelligence (AI) and natural language processing (NLP) are required.

AI aims to make communication between humans and computers easier by using natural language. However, the complexity of human language has resulted in the need for AI scientists to provide models that can understand human language using the NLP approach. NLP is the area of research that explores the capability of computers to understand the human language (Haan, 2018). NLP has played a great role in a chatbot for the case of human-computer interaction. It is a subfield of AI, which is used for enabling the computer to and process human language.

In the chatbot application, Communication via a natural language requires two fundamental skills, producing text and understanding it. That has known as natural language understanding and natural language generation. Natural language understanding is the process of making computers read the text and understands the intent or the meaning of the text like a human. After understanding the text sent from the user the computer generates the response for the user by understanding the context of the users' text this process is known as natural language generation (Karlin & Alexander, 1962).

Conversational AI is one of the hot research areas that NLP and AI technology has been applied. It is the set of technology behind automated messaging and speech enabled applications that offer human like interaction between computer and human by recognizing speech and text understanding intent, deciphering different languages, and responding in a way that mimics human conversation. A conversational bot is "an artificial intelligence model that can have the ability to process a conversation through auditory or textual methods. Such programs are often designed to convincingly simulate how a human would behave as a conversational partner." (Klopfenstein, Delpriori, & Malatini, 2017).

Unquestionably, one can observe conversational AI or chatbots more and more frequently, in increasingly everyday situation through various day to day activities. Since it is an artificial intelligence era and advancement of self-processing machines for multiple tasks, it is possible to implement conversational chatbot. This is possible based on machine learning algorithms with natural language processing that construct a model for making the machine intelligent on deciding the request, analyzing data, making suggestion, and providing auto service across various language.

Conversational bots are an artificial intelligence (AI) that designed to simulates interactive human conversation and communication to process specific task such as customer's service, restaurant reservation, shopping requests, health services, health consultant, etc. (Klopfenstein, Delpriori, & Malatini, 2017). They can be utilized in various different services and designed functions, like reservations purpose, recreation purpose, selling purpose, travel, utilities and advice service, agricultural purpose, sports purpose, health service by providing a simple communication interface (Følstad, Skjuve, & Brandtzaeg, 2019).

Consequently, there are totally different conversational systems designed as mentioned below. One of the oldest and also-know chatbot is a program called Eliza, created by the Artificial intelligence Laboratory in MIT, which dates between 1994-6 is based on pattern matching of some key-words and based on those words Eliza replies an open-ended response. Eliza can be considered as the first conversational agent developed by Joseph Weizenbaum (Klopfenstein, Delpriori, & Malatini, 2017). The conversations are tracked by the user utterance and system tries to work on some keywords and generate response based on the keyword between the system and user utterance (Klopfenstein, Delpriori, & Malatini, 2017).

The other conversational agent is the general AI of JABBERWACKY stores everything everyone has ever said, and find the most appropriate thing to say using contextual pattern matching techniques this mean further conversational AI, which was modeled by Rollo Carpenter, who try to simulate natural conversation on open domain knowledge and mimic human conversation and does not do anything other than (Hill, Randolph, & Farreras, 2015).

When we come to over viewing of the recent conversational AI, we have a tendency to get the Alexa Prize hosted by Amazon on offer off, around $2.5 million (A. Ram et al., , 2018). Alexa Prize is anticipated to create communication systematically and track with humans on well liked event like sport event, political hot space, amusement of the peoples, fashions phenomena and technologies happened for 20 minutes (A. Ram et al., , 2018). The opposite virtual help offered on totally different phones includes Siri (virtual assistance for apple) (F. B. Politecnico et al., 2018). Google Assistant (for golem phones), Dragon Mobile assistant (it is troublesome to a conventional larva, but can answer basic and relevant weather report) (Hoy, 2018), sensible voice assistant which may manage the golem interface via voice.

All conversational AI are common in using natural language to process the conversation; beyond that the conversational bots can be implemented based on deep learning, statistical models or, hybrid of the two (Nimavat & Champaneria, 2017). Though the different conversational systems having their own weakness like conversation tracking, handling correct response for user utterance, they are built to create an easy interface between human agent and an artificial agent to give virtual assistance on the problems rose from the human agent.

2.3. Afaan Oromo Language and Writing System

Afaan Oromo is a mother tongue for Oromo. The Oromo people are dominantly inhabited in the Oromia region. As an indication of strong ties of the Afaan Oromo language with other regions in Ethiopia, Oromia is the region that neighbors all the regions in Ethiopia except the Tigray region. Oromo people are also inhabited in neighboring countries like Kenya and Somalia. Afaan Oromo is part of the Lowland East Cushitic group within the Cushitic family of the Afro-Asiatic family. Afaan Oromo has a very rich morphology like other African and Ethiopian languages. It is used by Oromo people, who are the largest ethnic group in Ethiopia, which amounts to 34.5% of the total population. The language has become the official language in Oromia regional offices and is also an instructional language starting from elementary to university level. Afaan Oromo has a very rich morphology like other African and Ethiopian languages. The writing system of Qubee (Latin-based alphabet) has been started since 1842. It is one of the major African languages that is widely spoken and used in most parts of Ethiopia and some parts of other neighboring countries like Kenya, Tanzania, Djibouti, Sudan, and Somalia. It is the second-largest Cushitic language in the Africa continent next to Hausa. Nowadays, it is an official language of the Oromia state (which is the largest Regional State among the current Federal States in Ethiopia) (Segni, 2021).

Afaan Oromo’s writing system is almost phonetic since the way it is spoken in the way it is written. i.e., one letter corresponds to one sound. The language uses the Latin alphabet (Qubee) which was formally adopted in 1991, and it has its own consonants and vowel sounds. Afaan Oromo has thirty-three consonants, of these seven of them are combined consonant letters: ch, dh, ny, ph, sh, ts and Zh. The combined consonant letters are known as ’qubee dachaa’. Vowels have two natures in the language and they can result in different meanings. The natures are short and long vowels. A vowel is said to be short if it is one and a long vowel if it is two, which is the maximum. For example, Laga (river) and Laagaa (throat). In Afaan Oromo, as in the English language, vowels are sound makers and do stand by themselves. Afaan Oromo and English are different in sentence structuring. Afaan Oromo uses subject-object-verb (SOV) language. SOV is the type of language in which the subject, object, and verb appear in that order. Subject-verb-object (SVO) is a sentence structure where the subject comes first, the verb-second, and the third object. For instance, in the Afaan Oromo sentence "Mooneerraan bilisa bahe". "Mooneeraa "is a subject, "bilisa" is an object and "bahe" is a verb. Therefore, it has SOV structure. The translation of the sentence in English is "Mooneerraa has got freedom" which has SVO structure. There is also a difference in the formation of adjectives in Afaan Oromo and English. In Afaan Oromo adjectives follow a noun or pronoun; their normal position is close to the noun they modify while in English adjectives usually precede the noun. For instance, namicha gaarii (good man), gaarii (adj.) follows namicha (noun). Punctuation marks used in both Afaan Oromo and English languages are the same and used for the same purpose with the exception of the apostrophe. Apostrophe mark (’) in English shows possession, but in Afaan Oromo it is used in writing to represent a glitch (called hudhaa) sound. It plays an important role in Afaan Oromo reading and writing system. For example, it is used to write the word in which most of the time two vowels appeared together like "ba’e" to mean ("get out") with the exception of some words like "ja’a" to mean "six" which is identified from the sound created. Sometimes apostrophe mark (’) in Afaan Oromo interchangeable with the spelling "h" (Segni, 2021).

2.4. Exploring the Cultural Applications of Chatbots

Chatbots have emerged as versatile tools with applications spanning various domains, including cultural preservation and culinary exploration. Within cultural contexts, chatbots have been leveraged to facilitate language learning, cultural exchange, and heritage preservation. Wang et al. (2020) developed a real-time food recipe recommendation system with an intelligent agent, providing personalized culinary guidance and fostering cultural appreciation. Similarly, Lai et al. (2017) designed an online intelligent recipe recommendation system, enhancing user engagement through tailored recipe suggestions and real-time assistance. These applications highlight the potential of chatbots to serve as interactive platforms for cultural exploration and culinary discovery.

In the context of language and cultural preservation, chatbots offer innovative solutions for language learning and cultural immersion. By integrating natural language processing (NLP) algorithms, chatbots can facilitate interactive language practice and cultural exchange experiences. For example, virtual language tutors like Duolingo utilize chatbot technology to deliver personalized language lessons and cultural insights (Shi et al., 2017). Additionally, language chatbots such as Replika provide conversational practice in various languages, fostering cross-cultural communication and understanding (Hobbs et al., 2017). These language-oriented chatbots demonstrate the potential for technology to bridge linguistic and cultural barriers, promoting global interconnectedness and cultural exchange.

In the proposed study, the development of an Afaan Oromo text-based chatbot for traditional food recipes aligns with the broader applications of chatbots in cultural preservation and culinary exploration. By providing real-time assistance and personalized recipe recommendations in Afaan Oromo, the chatbot aims to promote cultural appreciation and culinary heritage preservation. Furthermore, the proposed study contributes to the growing body of literature on chatbot applications in diverse cultural contexts, highlighting the potential of technology to facilitate cultural exchange and culinary exploration in the digital age.

Another aspect of chatbots' cultural applications lies in their role as digital companions for users seeking cultural enrichment and immersion. Chatbots can simulate conversations with historical figures, cultural icons, or fictional characters, offering users an interactive experience that fosters cultural understanding and empathy. For instance, Mitsuku, an award-winning chatbot developed by Pandorabots, engages users in meaningful conversations ranging from trivia to philosophy, providing insights into cultural perspectives and historical events (Booth, 2020). Similarly, Replika's conversational AI platform enables users to engage in personalized interactions with virtual companions, fostering emotional connections and cultural exchange experiences (Hobbs et al., 2017). These examples illustrate how chatbots can serve as digital companions for cultural exploration, providing users with immersive and enriching experiences that transcend geographical boundaries.

Moreover, chatbots have been utilized in the cultural heritage sector to enhance visitor engagement and educational experiences in museums, galleries, and heritage sites. By incorporating chatbot technology into interactive exhibits and guided tours, cultural institutions can provide visitors with personalized and informative experiences that cater to their interests and preferences. For example, the Tate Modern museum in London developed the "Artbot" chatbot, which offers users personalized recommendations and insights about artworks in the museum's collection (Hood, 2020). Similarly, the British Museum's "MuseumBot" provides users with virtual tours, historical information, and interactive quizzes, enhancing visitor engagement and learning experiences (British Museum, n.d.). These applications demonstrate how chatbots can enhance cultural education and engagement, making cultural heritage more accessible and engaging to diverse audiences.

2.5. Related Works

AI-driven solutions, such as chatbots, have revolutionized the way individuals explore and engage with different cultures. Chatbots offer users personalized recommendations, real-time assistance, and immersive experiences that facilitate cultural exploration and understanding. Through AI-powered algorithms, chatbots can analyze user preferences, linguistic patterns, and cultural contexts to deliver tailored content and interactions that resonate with users' interests and cultural backgrounds. This convergence of AI and cultural exploration opens up new possibilities for fostering cross-cultural dialogue, promoting cultural appreciation, and enriching users' experiences in the digital age.

Within the realm of cultural exploration, chatbots have emerged as powerful tools for engaging users in immersive experiences that transcend geographical boundaries. For instance, the "Artbot" developed by the Tate Modern museum offers personalized insights into artworks, enabling users to deepen their understanding of art and cultural heritage (Booth, 2020). Similarly, chatbots have been employed in the culinary domain to provide users with real-time assistance and personalized recipe recommendations, fostering cultural appreciation and culinary exploration (Wang et al., 2020). These applications highlight the potential of chatbots to serve as digital companions for cultural exploration, offering users enriching experiences that promote cross-cultural understanding and appreciation.

Srivastava & Singh (2020) developed automatized medical chatbot system with human interaction using natural language diagnosis to provide medical aid. This medical chatbot was developed by using Artificial Intelligence Markup Language (AIML). AIML was developed by the Alicebot free software community and Dr. Richard S. Wallace during 1995-2000 (Introduction to AIML, 2018). The author has prepared the chatbot conversation by using a pattern and template tag. The pattern was the users' input and the template was the response of the bot. The researchers were to get the symptoms of the diseases from users' input and predict the disease. This system is qualified to identify symptoms from user inputs with a standard precision of 65%. After identifying symptoms, the system predicted the diseases by using K-nearest neighbor (KNN), SVM, Naive methods with an accuracy of 0.946, 0.886, and 0.8 respectively. However, even if the system achieves these records, it has limitations on understanding users' input. This means that the machine has understood only if the users' input matches with the pattern.

MamaBot is another Chatbot System based on ML and NLP for supporting Women and Families during Pregnancy (Vaira, L. *et al.*, 2018). Using ML and NLP the system is deployed to support women during pregnancy. the aim of the chatbot is designed to help and support pregnant women, mothers with young children, providing them rapid and useful suggestions in case of emergencies (e.g., "where is the nearest medical center") and also answers to their needs and references concerning disease prevention pathways, guidance on most proper lifestyles. It is designed to help pregnant women, mothers and families by providing different information (starting from a general level up to specific pathway questions) and giving a first-level support interaction to them as if it were a human being, in a chat like conversation model. It uses Microsoft Bot Framework and LUIS (Language Understanding Intelligent Service) as cognitive service.

Habitamu Asimare (2020) attempted designing and implementing an adaptive bot model to consult Ethiopian published laws using ensemble architecture with rules integrated. The study aims to develop a chatbot for consulting people regarding Ethiopian published laws. For this purpose, the researcher has used the Amharic language. The researcher employed RNN model for the development of the chatbot and recorded an accuracy of 73.12%. However, the researcher this result this study has a limitation on handling model overfitting. They used overfitting handling techniques only by changing the number of the data-splitting ratio. This is not enough to handle model overfitting.

Fekadu Eshetu (2021) attempted to develop a bilingual chatbot that supports both Amharic and Afaan Oromo languages for assisting Ethio telecom customers on customer services. In the study, the dataset was prepared using both Amharic and Afaan Oromo languages in the form of JSON by filtering the frequently asked questions from the organization website. For the model design purpose, DNN and Bi-LSTM model were applied. The accuracy of the DNN model was 83.52% and Bi-LSTM model was 85.23% before applying regularization technique. To overcome the model overfitting L2 regularization technique was applied the Bi-LSTM model. After regularization, the model recorded 82.6% accuracy. In addition, the model was also evaluated by using human evaluation. This evaluation was used to identify two criteria; that is system performance and user acceptance. The model performance was evaluated based on the number of queries they provide for the proposed system and the number of responses that returned from the system. The second evaluation was users' acceptability of the proposed system. This evaluation was evaluated based on five parameters those are attractiveness, response time, user-friendly, efficiency, and system feasibility.

Segni Bedasa (2021) also developed Afaan Oromo text-based chatbot for enhancing maize productivity (Bedasa, 2021). The study aims to assist the farmer in enhancing maize productivity by using the Afaan Oromo language. To develop the chatbot, the researcher used deep learning techniques specifically RNN and DNN. For feature extraction, purposes also used TF-IDF techniques. Finally, they get the accuracy for RNN and DNN 80% and 84% respectively. However, they get this accuracy this study may have a limitation on the number of target classes or the intents of the request is small i.e., they used only 25 target classes, and also the researcher do not consider over fitting handling techniques or model regularization techniques.

CHAPTER THREE
RESEARCH METHODOLOGY

Preserving cultural heritage through technology has become a vital pursuit in the digital age, enabling the widespread dissemination and appreciation of traditional customs and practices. The development of text-based chatbots offers a promising method for promoting cultural identity and fostering engagement with cultural artifacts. In the realm of culinary traditions, where recipes are repositories of cultural knowledge and heritage, leveraging machine learning to create chatbots provides a novel approach to sharing and preserving cultural cuisines. This research methodology details the systematic process of developing a text-based chatbot tailored for Afaan Oromo, a widely spoken language in Ethiopia and Kenya, with a focus on traditional food recipes. By harnessing natural language processing (NLP) and machine learning, this study aims to bridge the gap between technological innovation and cultural preservation, offering a digital platform for Afaan Oromo speakers to access, share, and celebrate their culinary heritage. The methodology encompasses the architecture of the proposed chatbot system, dataset description, dataset preparation, preprocessing, model selection, training, evaluation, deployment, and ethical considerations. This comprehensive and systematic approach ensures the successful development of the Afaan Oromo text-based chatbot for traditional food recipes. Through this endeavor, we aim to contribute not only to the advancement of machine learning research but also to the preservation and promotion of cultural diversity and heritage in the digital age.

3.1. Architecture of the Proposed Chatbot System

The architecture of the proposed chatbot system is meticulously designed to efficiently process user queries and provide accurate responses related to traditional Afaan Oromo food recipes. This system integrates several key components, including natural language processing (NLP) tasks, a pre-trained machine learning model, and a user-friendly graphical user interface (GUI). The NLP tasks encompass essential preprocessing steps such as text normalization, text cleaning, tokenization, and the creation of a bag-of-words model, which collectively transform raw text input into a structured format suitable for analysis. The pre-trained machine learning model, tailored for intent classification, leverages this structured data to accurately predict user intents and generate appropriate responses. The GUI facilitates seamless interaction between users and the chatbot, allowing users to input their queries and receive real-time cooking assistance in

Afaan Oromo. This architectural design ensures that the system is both robust and responsive, capable of handling the linguistic and cultural nuances of traditional Afaan Oromo cuisine while providing a practical tool for culinary guidance.

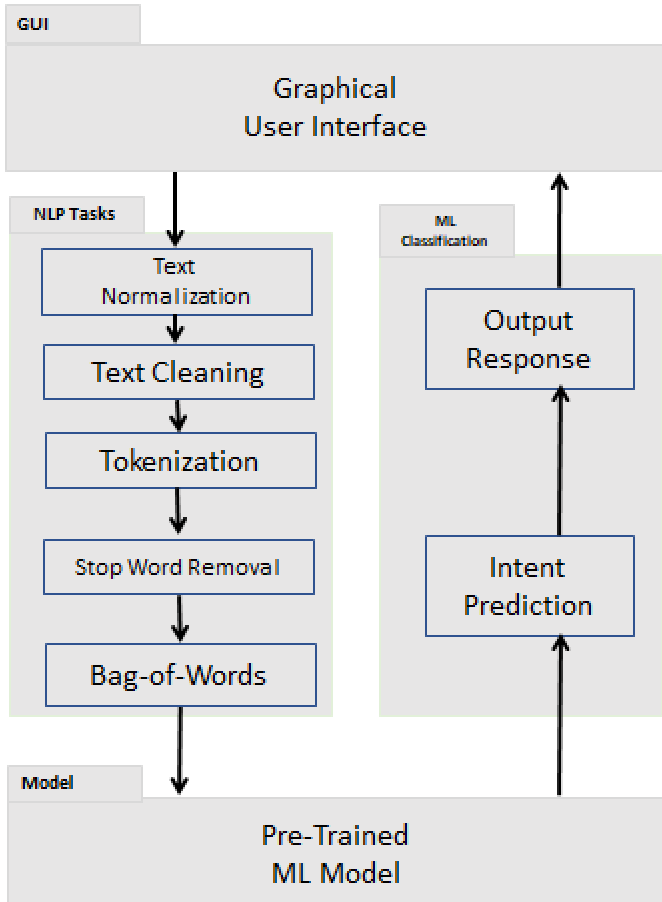


Figure 1: Architecture of the proposed chatbot system

The diagram of the proposed chatbot system illustrates the comprehensive architecture and flow of data from input to output, highlighting each crucial step in the process. When a user inputs a query in Afaan Oromo, the system preprocesses the input in real-time, vectorizes it, and feeds it into the trained model to predict the intent. Based on the predicted intent, the system retrieves an appropriate response from the database of curated responses, providing real-time, culturally and linguistically accurate assistance. The diagram encapsulates this flow, showcasing how each component interacts seamlessly to deliver an efficient and user-friendly chatbot experience, from data collection to model deployment and user interaction. This structured approach not only ensures high accuracy and relevance of the responses but also underscores the integration of advanced NLP and machine learning techniques with cultural preservation efforts, making traditional culinary knowledge accessible to a broader audience through modern technology.

3.2. Dataset Description

To assemble a comprehensive dataset of traditional Afaan Oromo food recipes, we meticulously curated sources from both traditional and digital platforms. Our primary sources included authoritative cookbooks and culinary compendiums, offering authentic recipes documented by culinary experts and cultural scholars, providing invaluable insights into traditional cooking practices. Additionally, we leveraged reputable digital resources such as websites, online forums, and mobile applications specializing in Afaan Oromo gastronomy. These digital platforms feature a diverse array of user-generated content, curated recipes, and multimedia resources that reflect contemporary tastes and preferences. We conducted thorough literature reviews and bibliographic searches to identify and procure these sources, complemented by scouring online forums and social media groups. Collaborations with local communities, cultural organizations, and culinary experts provided firsthand accounts and oral histories of traditional cooking practices. This multifaceted approach, combining traditional research methods with modern digital tools and community engagement, resulted in a diverse and representative dataset that captures the multifaceted nature of Afaan Oromo culinary traditions.

3.3. Dataset Preparation

The training dataset is prepared in JSON file format, which stands for JavaScript Object Notation. The JSON file format is ideal for our needs as it supports structured data and allows for easy manipulation and access. The dataset contains three main elements: tags, patterns, and responses.

Tag: This represents the intent or the main idea of the user query and is used as a target class while training the model.

Patterns: These are sets of user queries or frequently asked questions related to the intents, providing the model with various ways a user might ask a question.

Responses: These are sets of possible answers or replies corresponding to the user queries, allowing the chatbot to provide relevant responses to the users.

The following shows how the dataset is prepared. For example, the users want to ask greeting with Afaan Oromo Language. In Afaan Oromo they may be asked "Akkam Jiirta?", "Akkam?", "Jirtaa?", "Nagaa qabdaa?", "Akkam buulte?". The response for those queries may be “Haay, nagaa koodha ", "jira nagaa koodha". This, the intent or the tag of the dataset is greeting. Here we have to prepare greeting tag that is for Afaan Oromo text. Therefore, the converted JSON dataset is prepared as follows.

```
{
  "tag": "greeting",
        "patterns": ["Akkam Jiirta?", "Akkam?", "Jirtaa?", "Nagaa qabdaa?", "Akkam buulte?"]
        "responses": [ “Haay, nagaa koodha ", "jira nagaa koodha”]
}
```

Based on this JSON dataset we trained the model and finally, it predicts the tag and selects from the response randomly, and gives the appropriate responses. Large pair of utterances are prepared and ready to train, validate, and test the proposed model.

For our study on developing a text-based chatbot for traditional Afaan Oromo food recipes, we utilized the train_test_split function from the scikit-learn library to implement data splitting techniques for creating training and testing sets. This function randomly splits the dataset into two subsets: the training set and the testing set. Our implementation involved splitting the input data (train_x) and the corresponding target labels (train_y) into training and testing sets, with a specified test size of 0.1 (or 10% of the total dataset) and a random state of 1 to ensure reproducibility of the results. The resulting subsets, namely train_x, test_x, train_y, and test_y, represent the training and testing data for both input features and target labels, respectively. By splitting the dataset in this manner, we were able to train the chatbot model on the training data while reserving a portion of the data for evaluation on unseen samples, thus enabling us to assess the model's generalization performance and identify any potential issues such as overfitting. This data splitting technique ensures the reliability and validity of our model evaluation process, allowing us to make informed decisions regarding the model's effectiveness and suitability for its intended application in assisting users with traditional Afaan Oromo food recipes.

3.4. Data Preprocessing

Data preprocessing plays a critical role in preparing the dataset for subsequent analysis and modeling. For our study on traditional Afaan Oromo food recipes, we have implemented a series of data preprocessing techniques to ensure the quality, consistency, and usability of the dataset. Firstly, we conducted thorough data cleaning procedures to remove any irrelevant or redundant information, such as duplicate recipes or non-standardized ingredient names. This involved standardizing ingredient names, units of measurement, and cooking terminology to facilitate uniformity and ease of analysis. Additionally, we employed text normalization techniques to convert all text to a consistent format, such as converting uppercase letters to lowercase and removing diacritics or special characters to enhance text readability and consistency. Furthermore, we performed tokenization to split the text into individual words or tokens, enabling subsequent analysis at the word level. Stop word removal was also applied to eliminate common words that do not contribute to the overall meaning of the text, further refining the dataset for analysis. Finally, we conducted exploratory data analysis to gain insights into the distribution and characteristics of the dataset, identifying any outliers or anomalies that may require further investigation or handling. By implementing these data preprocessing techniques, we have ensured the quality, consistency, and usability of the dataset, laying a solid foundation for subsequent analysis and modeling tasks in our study.

3.4.1. Text Normalization

Text normalization is used for a series of related tasks meant to place all text on a level playing field. The normalization of text is based on the writing system of the language. To normalize Afaan Oromo text we need to convert all text to an equivalent upper case to lower cases. For example, the tokenized word "**Akkam"** has to be normalized as **"akkam"** which makes the computer understand the word **"Akkam"** has equivalent meaning with **"akkam",** or **"AKKAM"** etc.

3.4.2. Text Cleaning

The other common preprocessing task is clearing the text by removing different symbols and numbers from the text to make the computer understand the natural language easily. Users may use various symbols which are commonly used in text-based chatting such as different emoji symbols like [☝🖐👍👎👌☺✌🖐] to express their idea. For example [Akkam jiirta🖐🖐🖐!!]. Hence, removing these symbols is also required for clearing the text.

Removing punctuation marks is also required for text clearing. Afaan Oromo uses punctuation marks that are the same as with the English language. Therefore, we have to remove these punctuation marks from the text.

3.4.3. Tokenization

Tokenization is a step, which splits longer strings of text into smaller pieces, or tokens. Larger chunks of text have often been tokenized into sentences; sentences have often been tokenized into words, etc. Further processing is usually performed after a bit of text has been appropriately tokenized.

This study applies tokenization to the Afaan Oromo text by using spaces or word separator. For example, the text “Daabboon akkamitti hojjetama?” meaning "How to prepare bread?" has to be tokenized as:

daabboon	akkamitti	hojjetama

3.4.4. Stop Word Removing

It is a common practice to remove the non-content bearing terms, i.e., stop words which are not useful to specifically identify the intent of the utterances. Stop words are not helpful to find the context or the true meaning of a sentence. Removing them helps to reduce the size of the dataset without any doubt and makes increases the performance of the model. In this study, Afaan Oromo stop words are collected from different previous studies. Some of the Afaan Oromo stop words are “amma”, “inni”, “kanaaf”, “kanaaf”, “sitti”, “ammo”, “irra”, “Kanaafuu”, “sun” etc. In general, the process of removing these stop words helps to reduce the vocabulary of our model and attempt to find the more general meaning behind the sentences.

3.5. Word Vectorization

After performing text preprocessing tasks (such as text normalization, text cleaning, tokenization and stop word removal), the next step is word vectorization. When used a machine learning algorithm and work with textual data, machine learning algorithms do not work on the raw text directly. Hence feature extraction technique is used to convert text into a matrix (vector) of features. Word vectorization is a technique that is used to translate the raw data into a particular input machine-learning algorithm requires. Intensive study of the literature reveals that there are different types of techniques extracting features from text data. Bag-of-words (BoW), TF/IDF and Word2Vec are the most common word embedding techniques for feature extraction applied in Natural Language Processing. In this study, Bag-of-words (BoW) technique is applied for converting the text data into its equivalent numeric value. Bag-of-words (BoW) is a Natural Language Processing technique of text modeling and is used as a method of vectorization with text data. By using the bag-of-words (BoW) technique, we can convert a text into its equivalent vector of numbers. In this study bag-of-words (BoW) technique is chosen because it is very flexible and powerful way of extracting features from text data.

3.6. Model Building and Selection

In the process of model building and selection for our study focused on developing a text-based chatbot for traditional Afaan Oromo food recipes, we employed a systematic approach to identify the most effective architecture for our task. After careful experimentation and evaluation of various neural network configurations, we arrived at the final selected model, a three-layered neural network implemented using the Keras Sequential API. This model architecture is characterized by its simplicity and effectiveness, consisting of an input layer followed by the first hidden layer with 128 neurons, followed by a dropout layer to mitigate overfitting, another hidden layer with 64 neurons, another dropout layer for regularization, and finally, an output layer with a number of neurons equal to the number of intents to predict, utilizing a softmax activation function to compute the probability distribution over the output classes. To further optimize the model's performance, we compiled it using a categorical cross-entropy loss function, the Adam optimizer with a learning rate of 0.001, and specified accuracy as the evaluation metric. Subsequently, we trained the model on the training dataset for 200 epochs with a batch size of 5, monitoring its performance and convergence over the training epochs. Finally, upon achieving satisfactory performance, we saved the trained model weights and architecture to a file named 'chatbot_model.h5' for future deployment and evaluation. By leveraging this carefully designed and optimized neural network architecture, we have developed a robust and scalable chatbot model capable of accurately understanding user queries, providing step-by-step cooking instructions, and offering personalized cooking advice in Afaan Oromo. The model architecture is highlighted below:

1. Input Layer: The input layer contains a variable number of neurons corresponding to the same vector size as the bag-of-words present in the training dataset.

2. Hidden Layer (1st): This layer consists of 128 neurons and utilizes the Rectified Linear Unit (ReLU) activation function. Our contribution here lies in selecting an appropriate number of neurons to capture the complexity of the input data while preventing overfitting.

3. Dropout Layer (1st): Following the input layer, we incorporate a dropout layer with a dropout rate of 0.5. Our contribution involves introducing dropout regularization to mitigate overfitting and improve the generalization performance of the model.

4. Hidden Layer (2nd): The hidden layer comprises 64 neurons and utilizes the ReLU activation function. Here, our contribution lies in determining the optimal number of neurons to effectively capture underlying patterns and relationships in the data, balancing model complexity and computational efficiency.

5. Dropout Layer (2nd): Another dropout layer with a dropout rate of 0.5 is added after the hidden layer. Our contribution here mirrors that of the first dropout layer, further enhancing regularization to prevent overfitting and improve model generalization.

6. Output Layer: The output layer contains a variable number of neurons corresponding to the number of distinct intents present in the training dataset. It employs the softmax activation function to compute the probability distribution over the output classes. Our contribution involves dynamically adjusting the number of neurons in the output layer based on the specific requirements of the task, ensuring the model's capability to accurately predict user intents.

Table 1: Hyperparameters specifications

Hyperparameters	**Implemented Values**
Layers	Input Layer: Expects rows of data with the same vector size as the bag-of-words. Hidden Layer_1 = 128 neurons Hidden Layer_2 = 64 neurons Output Layer: It has equal number of nodes of the target classes

	or intents. It uses the softmax activation function.
Activation function	Relu
Loss function	categorical_crossentropy
Optimizer	Adam
Epochs	200
Dropouts	0.5
Learning rate	0.001
Batch size	5

In summary, our contributions to the model architecture include selecting appropriate numbers of neurons for the input and hidden layers, incorporating dropout regularization to mitigate overfitting, and dynamically adjusting the output layer to match the number of intents in the dataset. These contributions collectively enhance the model's performance, robustness, and ability to effectively address the task of developing a text-based chatbot for traditional Afaan Oromo food recipes.

3.7. Performance Evaluation

After building the model, we have to evaluate it to know whether the goals have been achieved or not. We evaluate the model, based on the dataset by using evaluation metric known as accuracy. Accuracy is the total correctly predicted data out of all training or testing data. It is used to measure how the model performed prediction of the given data into their corresponding classes. Accuracy is calculated by using the following formula.

$$Accuracy = \frac{\mathrm{TP+TN}}{\mathrm{TP+TN+FP+FN}}$$

where,

TP: True Positive: Predicted values correctly predicted as actual positive

FP: False Positive: Negative values predicted as positive

FN: False Negative: Positive values predicted as negative

TN: True Negative: Predicted values correctly predicted as an actual negative

Finally, User acceptance evaluation is also conducted primarily to evaluate usefulness of the proposed Chatbot system to users. This involves observing and measuring how well the artifact supports a solution to the problem and comparing with the objectives described above. To meet the established objectives of this study, the prototype system is extensively tested and evaluated including both performance of the prototype and user's acceptance.

3.8. Research Process Summary

The overall research process for developing a text-based chatbot for traditional Afaan Oromo food recipes encompasses several key stages, each contributing to the successful implementation and evaluation of the chatbot model. Initially, the research begins with comprehensive data collection from various reliable sources, including published books, websites, and applications, to gather a diverse and representative dataset of traditional Afaan Oromo food recipes. This dataset is then systematically annotated and curated to ensure linguistic accuracy and cultural authenticity, laying the groundwork for subsequent analysis and modeling tasks. Next, extensive data preprocessing techniques are applied to clean, normalize, tokenize, and remove stop words from the dataset, enhancing its quality and usability for training the chatbot model. Following data preprocessing, the model building and selection phase involve experimenting with various machine learning architectures and natural language processing techniques to identify the most effective model for understanding user queries and providing personalized cooking assistance.

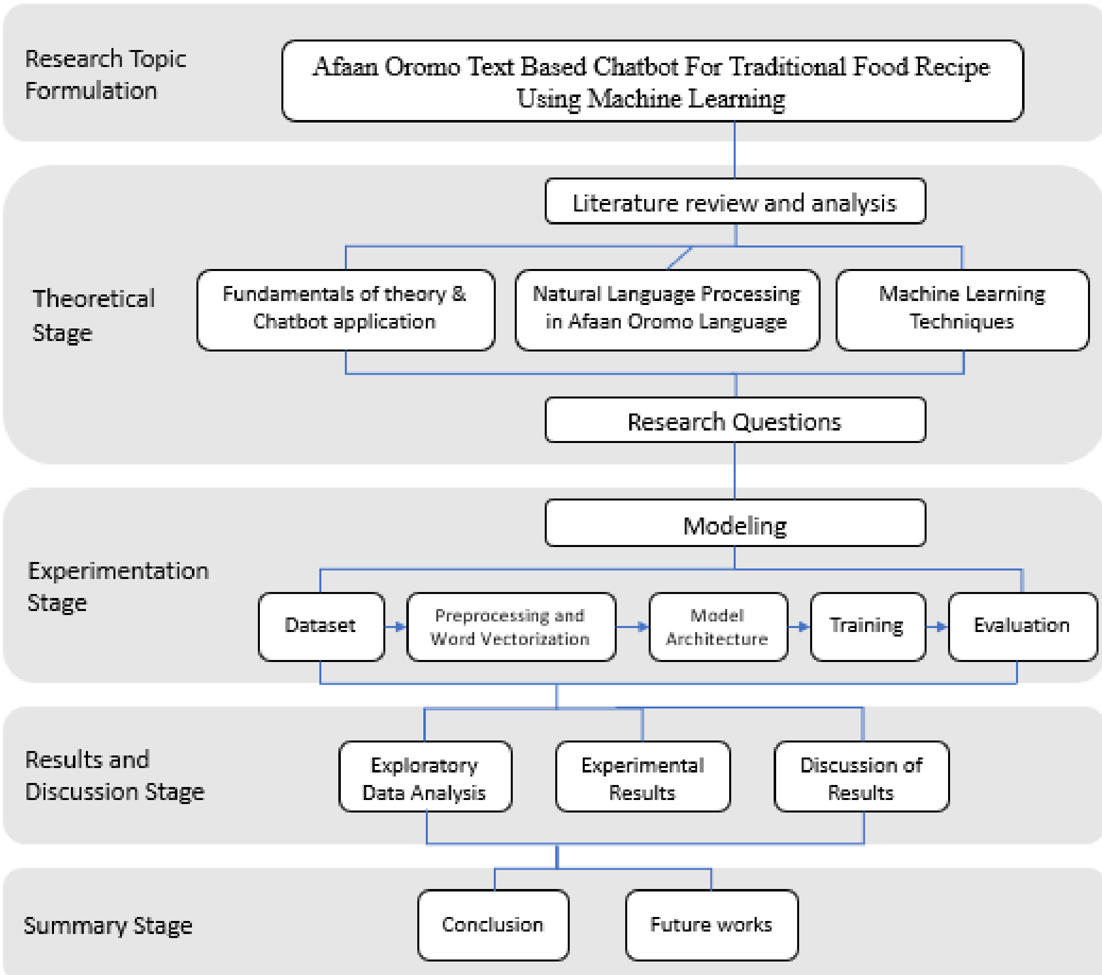


Figure 2: Overall the research process of the study

The selected model, a three-layered neural network, is trained on the preprocessed dataset using appropriate optimization techniques and evaluated for its effectiveness, accuracy, and user satisfaction. Finally, the research concludes with a comprehensive analysis of the model's performance, strengths, and limitations, as well as recommendations for future research and potential applications in real-world settings. Through this systematic research process, we aim to develop a robust and reliable chatbot model capable of assisting individuals with various levels of cooking experience in preparing traditional Afaan Oromo recipes, ultimately contributing to the preservation and promotion of Afaan Oromo culture and culinary heritage.

3.9. Tools

In this study, the following various libraries and tools are used.

Anaconda Navigator

Anaconda Navigator is a desktop application that is used to navigate, manage conda package and environments without using the command-line command. We used this navigator to launch various scientific Python development environments that are necessary for developing the proposed Chatbot.

Spyder IDE

Spyder is a free open-source scientific python development environment that is used to write a python program designed by and for scientists, engineers, and data analysts. This IDE is used to write and run python code for data processing and creating the proposed Chatbot system.

nltk

Natural language toolkit (nltk) is a python library that is used to work in natural language processing. This library contains different APIs that are used to implement tokenization, stemming, tagging, parsing, and semantic reasoning. We used this library for applying data preprocessing techniques that are applied to the development of the proposed Chatbot.

scikit-learn

scikit-learn is a python library used for applying machine learning algorithms. It contains different algorithms that are used for classification, regression, clustering dimensional reduction. We used this library in the proposed study for splitting the dataset into train data and test data.

Tensorflow

Tensorflow is an open-source library developed by Google, and it is used in machine learning application development for classification and prediction. It is used in the proposed Chatbot development for applying machine learning algorithms for intent classification.

numpy

Numpy is an open-source numerical library that is used for the visualization of array and matrix data. It is used to convert the data into the vector and reshaping the vector to fit the vector with the proposed model.

json

json is a built-in library, which can be used to work with JSON data. In the proposed study, we prepared our dataset by using JSON format. To read and manipulate the dataset we used this built-in library in the proposed Chatbot development.

matplotlib

matplotlib is a python library that is used for creating static, animated, and interactive visualizations in Python. In the development of the proposed Chatbot, it is used for visualizing data and results by using a graph.

tkinter

Tkinter is a Python binding to the Tk GUI toolkit. It is the standard Python interface to the Tk GUI toolkit, and is Python's de facto standard GUI. Tkinter is included with standard GNU/Linux, Miscrosoft Windows and macOS installs of Python. It is used for creating GUI chat interface for the proposed Chatbot.

CHAPTER FOUR
DESIGN AND IMPLEMENTATION

The Design and Implementation chapter serves as a comprehensive exploration of the technical aspects involved in creating the Afaan Oromo text-based chatbot for traditional food recipes. This chapter details the intricate process of translating conceptual ideas and methodological frameworks outlined in previous chapters into tangible software solutions. Through a systematic approach, this chapter delves into the design choices, development methodologies, and technical challenges encountered during the creation of the chatbot prototype. By elucidating the design principles and implementation strategies employed, this chapter provides readers with a deep understanding of the underlying technology powering the chatbot and its potential impact on preserving and promoting Afaan Oromo culinary heritage.

4.1. Text Preprocessing

The design and implementation of the text preprocessing section represent a pivotal stage in the development of the Afaan Oromo text-based chatbot for traditional food recipes. This section elucidates the intricate design decisions and technical methodologies employed to transform raw Afaan Oromo text data into a structured and standardized format conducive to machine learning model training. Through meticulous attention to detail and adherence to best practices in natural language processing, this section aims to address the unique linguistic characteristics and challenges inherent in processing Afaan Oromo text. By delineating the step-by-step process of text preprocessing, including data cleaning, tokenization, and normalization, this section provides readers with insights into the foundational techniques that underpin the subsequent stages of model development and evaluation. Additionally, the section explores implementation strategies, tool selection, and optimization techniques tailored to the specific requirements of the Afaan Oromo text corpus, ensuring robustness, efficiency, and scalability in the preprocessing pipeline. Overall, the design and implementation of the text preprocessing section lay the groundwork for the successful development of the Afaan Oromo text-based chatbot, enabling the effective utilization of linguistic data for cultural heritage preservation and culinary knowledge dissemination.

4.1.1. *Text Normalization*

The design and implementation of text normalization for the Afaan Oromo chatbot involve a systematic approach to ensure consistency and equivalence of textual inputs. Initially, the design phase encompasses the identification of text normalization requirements based on the linguistic characteristics of the Afaan Oromo language. Subsequently, the implementation process involves developing a function or module in the chatbot's codebase responsible for normalizing incoming text inputs. This function typically utilizes string manipulation techniques, such as converting text to lowercase using Python's built-in `lower()` method. Through meticulous design considerations and robust implementation, the text normalization component ensures that the chatbot comprehensively understands user queries, regardless of variations in text case, thereby enhancing user experience and engagement.

```python
def normalize_text(text):
    # Convert the text to lowercase
    normalized_text = text.lower()
    return normalized_text

# Example usage:
original_text = "Akkam"
normalized_text = normalize_text(original_text)
print("Original Text:", original_text)
print("Normalized Text:", normalized_text)
```

Figure 3: Python pseudo code for text normalization

Here's a Python pseudo code for text normalization. This function **normalize_text()** takes a string as input and converts it to lowercase using the **lower()** method available for strings in Python. This ensures that all text is normalized to the same case, allowing the computer to understand words with equivalent meanings regardless of their original case.

4.1.2. *Text Cleaning*

The design and implementation of text cleaning for the proposed Afaan Oromo chatbot involves a systematic approach aimed at enhancing the chatbot's ability to comprehend and process natural language inputs. Initially, the design phase entails identifying common text elements that may impede the chatbot's understanding, such as emoji symbols and punctuation marks. Subsequently, the implementation process involves developing functions or modules within the chatbot's codebase to perform specific cleaning tasks. This includes defining regular expression patterns to target and remove emoji symbols, as well as utilizing string manipulation techniques to eliminate punctuation marks. Through meticulous design considerations and robust implementation, the text cleaning component ensures that the chatbot can accurately interpret user queries and provide meaningful responses, thereby enhancing the overall user experience and facilitating seamless communication between users and the chatbot.

```python
import re

def clean_text(text):
    # Remove emoji symbols
    cleaned_text = remove_emoji(text)
    # Remove punctuation marks
    cleaned_text = re.sub(r'[^\w\s]', '', cleaned_text)
    return cleaned_text

def remove_emoji(text):
    # Define emoji patterns
    emoji_pattern = re.compile("["
                               u"\U0001F600-\U0001F64F"  # emoticons
                               u"\U0001F300-\U0001F5FF"  # symbols & pictographs
                               u"\U0001F680-\U0001F6FF"  # transport & map symbols
                               u"\U0001F1E0-\U0001F1FF"  # flags (iOS)
                               u"\U00002500-\U00002BEF"  # chinese char
                               u"\U00002702-\U000027B0"
                               u"\U00002702-\U000027B0"
                               u"\U000024C2-\U0001F251"
                               u"\U0001f926-\U0001f937"
                               u"\U00010000-\U0010ffff"
                               u"\u2640-\u2642"
                               u"\u2600-\u2B55"
                               u"\u200d"
                               u"\u23cf"
                               u"\u23e9"
                               u"\u231a"
                               u"\ufe0f"  # dingbats
                               u"\u3030"
                               "]+", flags=re.UNICODE)
    # Remove emoji symbols
    cleaned_text = emoji_pattern.sub(r'', text)
    return cleaned_text

# Example usage:
original_text = "Akkam jiirta□□□!!"
cleaned_text = clean_text(original_text)
print("Original Text:", original_text)
print("Cleaned Text:", cleaned_text)
```

Figure 4: Python pseudo code for text cleaning

The provided pseudo code outlines a text cleaning process designed for the proposed Afaan Oromo chatbot. Text cleaning is crucial for ensuring the chatbot can effectively understand and process natural language inputs. The **'clean_text()'** function initiates the cleaning process by removing emoji symbols using a regular expression pattern defined within the **'remove_emoji()'** function. Emojis, commonly used in text-based chatting, may obscure the chatbot's comprehension if not removed. Next, the **'re.sub()'** method is employed to eliminate punctuation marks from the text, as Afaan Oromo employs punctuation identical to English. Punctuation marks can interfere with the chatbot's language understanding, necessitating their removal for improved clarity. This comprehensive cleaning process ensures that the chatbot is equipped to interpret user queries accurately and respond effectively, enhancing the overall user experience and interaction quality.

4.1.3. Tokenization

In designing and implementing tokenization for the proposed Afaan Oromo chatbot, the focus is on breaking down textual inputs into smaller units or tokens to facilitate language processing. The design phase involves identifying suitable tokenization techniques that align with the linguistic structure of Afaan Oromo. Subsequently, the implementation entails developing a function or module within the chatbot's codebase to perform tokenization tasks. In this case, tokenization is achieved by splitting text based on word separators, such as spaces, to isolate individual words. This simplistic approach ensures that the chatbot can effectively parse user queries and responses, enabling it to understand the context and meaning conveyed by the input text. Through meticulous design considerations and robust implementation, tokenization plays a pivotal role in enhancing the chatbot's language understanding capabilities and overall performance, ultimately contributing to a seamless user experience.

```python
def tokenize_text(text):
    # Split the text into tokens using spaces as word separators
    tokens = text.split()
    return tokens

# Example usage:
original_text = "Daabboon akkamitti hojjetama?"
tokens = tokenize_text(original_text)
print("Original Text:", original_text)
print("Tokens:", tokens)
```

Figure 5: Python pseudo code for tokenization

The provided pseudo code illustrates a tokenization process tailored for the proposed Afaan Oromo chatbot, aiming to break down longer strings of text into smaller units or tokens. Tokenization is essential for enabling the chatbot to understand and process natural language inputs effectively. The **tokenize_text()** function implemented in the pseudo code takes a string of text as input and utilizes the **split()** method to separate the text into tokens using spaces as word separators. This simplistic approach allows the chatbot to tokenize sentences into individual words, facilitating further processing and analysis. For instance, the sentence "Daabboon akkamitti hojjetama?" meaning "How to prepare bread?" is tokenized into three tokens: "daabboon," "akkamitti," and "hojjetama." By breaking down text into its constituent tokens, the chatbot gains a granular understanding of user queries, enhancing its ability to generate accurate and relevant responses.

4.1.4. Stop Word Removing

In crafting the design and implementation of stop word removal for the proposed Afaan Oromo chatbot, the primary goal is to enhance the model's comprehension by filtering out non-essential terms. Initially, the design phase involves compiling a comprehensive list of Afaan Oromo stop words sourced from previous studies. Subsequently, the implementation process centers on developing a function or module within the chatbot's codebase to execute the removal of these stop words from textual inputs. Utilizing the compiled list, the function systematically filters out stop words from tokenized sentences, thereby refining the vocabulary and focusing on the salient content. Through this iterative process, the chatbot is better equipped to discern the underlying meaning of user queries and responses, ultimately enhancing its language understanding capabilities and improving overall performance.

The pseudo code below outlines the process of removing stop words from text inputs for the proposed Afaan Oromo chatbot. Stop words, which are non-content-bearing terms that do not contribute to the specific identification of user intents, are filtered out to enhance the chatbot's ability to understand the true meaning and context of sentences. The **'remove_stopwords()'** function takes a string of text and a predefined list of Afaan Oromo stop words as inputs. The text is split into tokens, and then each token is checked against the stop words list. Tokens that are not stop words are retained, while stop words are discarded. This filtering process reduces the size of the vocabulary used by the chatbot and helps identify the more general meaning behind sentences. By removing stop words, the chatbot can focus on relevant content and improve its performance in understanding and responding to user queries.

```python
def remove_stopwords(text, stopwords):
    # Split the text into tokens
    tokens = text.split()
    # Remove stopwords from the tokens
    filtered_tokens = [token for token in tokens if token not in stopwords]
    # Join the filtered tokens back into a sentence
    filtered_text = ' '.join(filtered_tokens)
    return filtered_text

# Afaan Oromo stop words
afan_oromo_stopwords = ["amma", "inni", "kanaaf", "sitti", "ammo", "irra", "kanaafuu

# Example usage:
original_text = "Akkam akka bu'uuraa irra jedhu."
filtered_text = remove_stopwords(original_text, afan_oromo_stopwords)
print("Original Text:", original_text)
print("Filtered Text:", filtered_text)
```

Figure 6: Python pseudo code for stop word removal

4.2. Word Vectorization

In designing and implementing word vectorization for the proposed Afaan Oromo chatbot, the objective is to transform textual data into a numerical format suitable for machine learning algorithms. The design process involves selecting an appropriate word embedding technique, considering factors such as flexibility, effectiveness, and compatibility with the linguistic characteristics of Afaan Oromo. In this study, the Bag-of-Words (BoW) technique is chosen for its versatility and capability to convert text into numeric vectors efficiently. Implementation entails utilizing the `CountVectorizer` class from the scikit-learn library to perform word vectorization. This involves fitting the vectorizer to the text corpus and transforming it into a matrix of token counts. By converting text into numerical vectors, the chatbot gains the ability to process and analyze textual data effectively, thereby enhancing its language understanding capabilities and overall performance.

```python
from sklearn.feature_extraction.text import CountVectorizer

def word_vectorization(text_corpus):
    # Initialize CountVectorizer object
    vectorizer = CountVectorizer()
    # Fit the vectorizer to the text corpus and transform the text into numeric vecto
    word_vectors = vectorizer.fit_transform(text_corpus)
    return word_vectors

# Example usage:
text_corpus = ["akkam akka bu'uuraa irra jedhu", "daabboon akkamitti hojjetama", "amn
word_vectors = word_vectorization(text_corpus)
print("Word Vectors:\n", word_vectors.toarray())
print("Vocabulary:\n", vectorizer.get_feature_names())
```

Figure 7: Python pseudo code for word vectorization

The provided pseudo code illustrates the process of word vectorization for the proposed Afaan Oromo chatbot, utilizing the Bag-of-Words (BoW) technique. After conducting preliminary text preprocessing tasks such as normalization, cleaning, tokenization, and stop word removal, word vectorization becomes imperative for converting textual data into a format suitable for machine learning algorithms. In this study, the Bag-of-Words approach is selected as the word embedding technique due to its flexibility and effectiveness in feature extraction from text data. The `word_vectorization()` function defined in the pseudo code employs the `CountVectorizer` class from the scikit-learn library to transform the text corpus into numeric vectors. Each vector represents a document in the corpus, with each element denoting the frequency of a particular word in that document. By converting text into numeric vectors, the chatbot gains the ability to process and analyze textual data effectively, paving the way for subsequent stages of model training and inference. Overall, word vectorization plays a crucial role in bridging the gap between textual data and machine learning algorithms, empowering the chatbot to understand and respond to user queries more accurately.

4.3. Intent Prediction Model Building

In designing and implementing the training and model building process for the proposed Afaan Oromo chatbot, the focus is on constructing a deep neural network capable of understanding and predicting user intents. The design phase involves selecting an appropriate architecture, which includes defining the number of layers, neurons per layer, and activation functions. In this instance, a three-layer architecture with varying numbers of neurons and dropout layers is chosen to prevent overfitting. Using the Keras sequential API simplifies the implementation process, allowing for straightforward layer-by-layer construction. Following the design phase, the implementation entails compiling the model with suitable loss and optimization functions and training it on a labeled dataset containing input patterns and corresponding output classes. This involves iteratively adjusting the model's parameters over multiple epochs to minimize loss and improve accuracy. Once the model has been trained satisfactorily, it is saved for future use, enabling seamless integration into the chatbot application. Through meticulous design considerations and robust implementation, the training and model building process equips the chatbot with the ability to effectively interpret user queries and provide accurate responses in the Afaan Oromo language.

```
from keras.models import Sequential
from keras.layers import Dense, Dropout
import numpy as np

# Create model - 3 layers. First layer 128 neurons, second layer 64 neurons and 3rd o
# number of neurons equal to number of intents to predict output intent with softmax
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'))
model.compile(loss='categorical_crossentropy', optimizer='adam', metrics=['accuracy']

# Fitting the model
hist = model.fit(np.array(train_x), np.array(train_y), epochs=200, batch_size=5, verb

# Save the model
model.save('chatbot_model.h5')
```

Figure 8: Python pseudo code for model building

The provided pseudo code outlines the process of training and building a deep neural network model for the proposed Afaan Oromo chatbot using the Keras sequential API. The model architecture consists of three layers: an input layer with 128 neurons, a hidden layer with 64 neurons, and an output layer with a number of neurons equal to the number of intents to predict output intent using the softmax activation function. Dropout layers are incorporated to mitigate overfitting during training. The model is compiled with categorical crossentropy loss and Adam optimizer to optimize the model parameters during training. Subsequently, the model is trained using the provided training data, consisting of input patterns and corresponding output classes converted into numeric representations. Training is performed for 200 epochs with a batch size of 5, optimizing the model's parameters to achieve acceptable accuracy. Finally, the trained model is saved as 'chatbot_model.h5', which can be later used for inference to predict intents from user inputs in the Afaan Oromo language.

4.4. Graphical User Interface

In designing and implementing the Graphical User Interface (GUI) for the proposed Afaan Oromo chatbot, the objective is to create an intuitive and interactive platform for users to engage with the chatbot seamlessly. The design phase involves selecting the Tkinter library for GUI development due to its simplicity and versatility. The GUI comprises a text entry field for user input, a button to send messages, and a chat history display area. Additionally, the implementation process involves loading the trained model and preprocessed data, including vocabulary and intent classes, for efficient prediction of user intents. Upon receiving user input, the model predicts the intent, selects an appropriate response, and updates the chat history accordingly. This integration of GUI elements with the trained model enables users to interact with the chatbot effortlessly, fostering a user-friendly and engaging experience in conversing with the Afaan Oromo chatbot.

The provided Python code below outlines the process of deploying the trained Afaan Oromo chatbot model with a Graphical User Interface (GUI) using the Tkinter library in Python. The code first loads the trained model and preprocessed data, including the vocabulary words and intent classes, which were saved during model training. Upon user input, the program preprocesses the text and predicts the intent using the loaded model. It then selects a random response associated with the predicted intent to generate a reply. The GUI window created with Tkinter displays a chat history area, an entry field for user input, and a send button. When the user inputs a message and clicks the send button, the program generates a response and updates the chat history with the user input and the bot's response. This interactive interface allows users to converse with the chatbot seamlessly, providing an intuitive and user-friendly experience.

```
from tkinter import *
import pickle
import random
import numpy as np
from keras.models import load_model

# Load trained model
model = load_model('chatbot_model.h5')

# Load preprocessed data
with open('words.pkl', 'rb') as f:
    words = pickle.load(f)
with open('classes.pkl', 'rb') as f:
    classes = pickle.load(f)

# Function to preprocess user input
def preprocess_input(input_text, words):
    # Tokenize input text
    tokenized_input = tokenize_input(input_text)
    # Create bag of words
    bag_of_words = [0]*len(words)
    for s in tokenized_input:
        for i, w in enumerate(words):
            if w == s:
                bag_of_words[i] = 1
    return np.array(bag_of_words).reshape(1, -1)
```

```
# Function to predict intent
def predict_intent(input_text):
    preprocessed_input = preprocess_input(input_text, words)
    intent = model.predict(preprocessed_input)[0]
    predicted_intent_index = np.argmax(intent)
    return classes[predicted_intent_index]

# Function to get random response
def get_random_response(intent):
    responses = {
        'greeting': ['Hello!', 'Hi there!', 'How can I help you?'],
        'farewell': ['Goodbye!', 'See you later!', 'Take care!'],
        # Add more intent-responses mappings here
    }
    return random.choice(responses.get(intent, ['I am not sure how to respond to tha

# Function to handle user input and generate response
def generate_response():
    user_input = entry_field.get()
    intent = predict_intent(user_input)
    bot_response = get_random_response(intent)
    chat_history.insert(END, "You: " + user_input + "\n")
    chat_history.insert(END, "Bot: " + bot_response + "\n\n")
```

```
# Create GUI window
root = Tk()
root.title("Afaan Oromo Chatbot")

# Create chat history window
chat_history = Text(root, bd=1, bg="white", width=50, height=8)
chat_history.grid(row=0, column=0, columnspan=2, padx=10, pady=10)

# Create entry field for user input
entry_field = Entry(root, width=30)
entry_field.grid(row=1, column=0, padx=10, pady=10)

# Create button to send message
send_button = Button(root, text="Send", command=generate_response)
send_button.grid(row=1, column=1, padx=10, pady=10)

# Run the GUI window
root.mainloop()
```

Figure 9: Python Code for Graphical User Interface (GUI)

CHAPTER FIVE
RESULTS AND DISCUSSION

The Results and Discussion chapter marks a pivotal phase in the research journey, where the developed Afaan Oromo text-based chatbot undergoes rigorous evaluation and scrutiny. This chapter presents the empirical findings obtained from experimental testing and user feedback, shedding light on the chatbot's performance, effectiveness, and user satisfaction. Through objective metrics and qualitative analysis, the chapter unveils insights into the chatbot's ability to understand user queries, provide accurate recipe recommendations, and engage users in meaningful interactions. Furthermore, the discussion section critically examines the strengths, limitations, and implications of the chatbot's functionality, offering valuable insights for future research and practical applications. By synthesizing empirical evidence with theoretical insights, this chapter enriches our understanding of the chatbot's role in preserving cultural heritage and advancing technology-mediated linguistic diversity.

2.6. Dataset Preparation

In the Data Preparation section, we embark on the crucial task of preparing the Afaan Oromo food recipe dataset for analysis and model training. This section outlines the systematic process of loading the dataset into a suitable data structure, such as a Pandas DataFrame, to facilitate efficient manipulation and exploration. By leveraging the flexibility and functionality of Pandas, we gain insights into the dataset's structure and distribution, essential for understanding the underlying patterns and characteristics of the Afaan Oromo text corpus. Furthermore, the section delves into the preprocessing of the dataset, encompassing steps such as removing unnecessary characters, converting text to lowercase, and handling any missing values. Through meticulous data cleaning and transformation, we ensure the integrity and consistency of the dataset, laying the foundation for subsequent analysis and model development. Overall, the Data Preparation section plays a pivotal role in harnessing the potential of the Afaan Oromo food recipe dataset to train the chatbot effectively and enable meaningful insights into Afaan Oromo culinary heritage.

Once the data collection task was completed the dataset was prepared in JSON file format. It contains three things, namely: tag, patterns, and responses. Tag is the intent or the idea of the user query, which is used as a target class while we train the model. The following table describes all the 26 intents (target classes) which are identified at the end of the dataset preparation.

Table 2: Intent (target class) description

No.	Intent (target class)	Description
1	Cuukkoo	Cuukkoo is a traditional Oromo dish consisting of small diced pieces of barley flour seasoned with melted butter, coriander powder, and salt.
2	Caccabsaa	Caccabsaa is a traditional Oromo flatbread made from barley flour (or a mixture of teff and wheat flour) and seasoned with red pepper powder and seasoned butter.
3	Qiixxaa Marqaa	Qitaamarqaa is a thin, crispy flatbread popular in the Selale region of Ethiopia.
4	Marqaa Arsii	Marquaa Arsi is a variation of Marquaa prepared in the Arsi region of Oromia.
5	Marqaa	Marquaa is another variation of Marquaa commonly prepared in other parts of Oromia.
6	Ukkaamsaa	Ukkaamssa is a highly seasoned minced beef dish infused with garlic, ginger, chili pepper, and butter.
7	Cororsaa	Coorossaa is a special dish prepared in Wellega for weddings and traditional celebrations.
8	Xorosho	Xorosho is a thicker variation of pita bread, commonly served as a snack in Arsii Oromia.
9	Dodoboo	Dodoboo is a traditional food that was once popular in Salaale but is now less known among many people.
10	Azifaa	Azifa is a delicious cold lentil dish that complements any Ethiopian meal, akin to a salad on an American table.
11	Raafuu	Raafuu is a quintessential Ethiopian dish featuring either collard greens or kale cooked with flavorful spices.
12	Ittoo buqqee	Ittoo buqqee is a sweet and delicious Ethiopian dish made with pumpkin or squash, known as "duba" in Amharic.
13	Butecha	Butecha is a flavorful and refreshing Ethiopian dish made with chickpea flour, lemon juice, and spices.
14	Ittoo waaddii lukkuu	It is a delicious Ethiopian chicken dish cooked with onions, berbere spice blend, jalapeños, and optionally red wine or t'ej (Ethiopian honey wine).
15	Kitfoo & Itittuu	Kitfo is a beloved Ethiopian beef dish traditionally served raw but can also be prepared fully cooked. It is typically seasoned with spices and served with ayib, a soft Ethiopian cheese.
16	Waaddii hanqaaquu	Waaddii hanqaaquu is a flavorful Ethiopian dish made with eggs, diced vegetables, and spices.
17	Silsi fi Tihlo	Silsi is a spicy tomato and onion sauce from Eritrea, while Tihlo is a dish from Tigray in northern Ethiopia made from roasted barley or wheat flour balls.
18	Defo Daaboo	Defo Dabo is a traditional Ethiopian leavened bread that incorporates unique flavors and is wrapped in banana leaves before baking.
19	Kategna	Kategna is a delightful Ethiopian snack made from injera, spiced clarified butter (niter kibbee), and berbere seasoning.
20	Akaawwi	It is a traditional Ethiopian snack often enjoyed between meals or as a crunchy accompaniment to coffee.
21	Itoophiyaa Kool Slaawu	This Ethiopian-inspired cole slaw offers a flavorful twist on the classic dish, incorporating traditional Ethiopian ingredients like jalapeño peppers for a spicy kick.
22	Kochkocha	Kochkocha is a fiery Ethiopian dipping sauce, akin to salsa verde, that adds a burst of flavor and heat to any dish.
23	Nagaa gaafachuu	This intent is about greeting
24	Nagaatti	This intent is about farewell
25	Galataa	This intent is about to say thank you
26	deebii hin arganne	This intent is about no response

The patterns (statements) were prepared based on considering different ways of expression for the same intent. In order to generate multiple expressions for a single intent, we accept the expression of various people for certain intents. Finally, the researcher generates the patterns for the dataset.

2.7. Exploratory Data Analysis

The Exploratory Data Analysis (EDA) section serves as a critical phase in understanding the characteristics and patterns inherent in the Afaan Oromo food recipe dataset. This section embarks on a systematic analysis of the distribution of intents within the dataset, providing insights into the prevalence and diversity of user queries and interactions. By examining the frequency of different intents, we gain a holistic understanding of the underlying patterns and user preferences encoded within the dataset. To facilitate intuitive visualization of intent distributions, this section employs the Plotly library to generate interactive bar plots. Through these visualizations, the x-axis represents the intents, while the y-axis depicts the count of patterns or responses associated with each intent. By harnessing the power of interactive visualizations, we aim to elucidate complex patterns and trends in the dataset, empowering researchers and stakeholders to derive meaningful insights into user behavior and preferences. Overall, the Exploratory Data Analysis section lays the groundwork for informed decision-making in subsequent stages of chatbot development and evaluation.

In [6]:

```
df.head(10)
```

Out[6]:

	tag	patterns	responses
0	Cuukkoo	[Cuukkoo, cuukkoon akkamitti hojjetama, cuukko...	[Cuukkoon nyaata aadaa Oromoo yoo ta'u, daakuu...
1	Caccabsaa	[Caccabsaa, caccabsaan ynaata akkamiiti, cacca...	[Caccabsaa daabboo diriiraa aadaa Oromoo daaku...
2	Qiixxaa Marqaa	[Qiixxaa Marqaa, Qiixxaa marqaa naannoo kamitt...	[Qiixxaa Marqaa\nQiixxaa marqaan daabboo dirii...
3	Marqaa Arsii	[Marqaa Arsii, Marqaa Arsii midhaan akkamiirra...	[Marqaa Arsii\nMarqaan Arsii jijjiirama marqaa...
4	Marqaa	[Marqaa, Marqaa midhaan akkamiirraa qopheessin...	[Marqaa \nMarqaa garaagarummaa marqaa kan bira...
5	Ukkaamsaa	[Ukkaamsaa, Ukkaamsaan maalirra hojjetama, Ukk...	[Ukkaamssa nyaata foon loonii ciccitaa baay'ee...
6	Cororsaa	[Cororsaa, Cororsaa maalirra qopheessuu dandee...	[Cororsaan nyaata addaa Wellegaatti cidha fi a...
7	Xorosho	[Xorosho, Xorosho naannoo kamitti irra beekkam...	[Xorosho garaagarummaa furdaa daabboo pitaa yo...
8	Dodoboo	[Dodoboo, Dodobo yeroo ammaa kana maalif sirri...	[Dodoboo nyaata aadaa yeroo tokko Salaalee kee...
9	Azifaa	[Azifaa, Azifa yeroo qopheessinu meeshaa akkam...	[Azifaan nyaata mi'aawaa qamadii qabbanaawaa t...

Figure 10: Dataframe to show top 10 rows

In [7]:

```
dic = {"tag":[], "patterns":[], "responses":[]}
for i in range(len(df)):
    ptrns = df[df.index == i]['patterns'].values[0]
    rspns = df[df.index == i]['responses'].values[0]
    tag = df[df.index == i]['tag'].values[0]
    for j in range(len(ptrns)):
        dic['tag'].append(tag)
        dic['patterns'].append(ptrns[j])
        dic['responses'].append(rspns)

df = pd.DataFrame.from_dict(dic)
df
```

Out[7]:

	tag	patterns	responses
0	Cuukkoo	Cuukkoo	[Cuukkoon nyaata aadaa Oromoo yoo ta'u, daakuu...
1	Cuukkoo	cuukkoon akkamitti hojjetama	[Cuukkoon nyaata aadaa Oromoo yoo ta'u, daakuu...
2	Cuukkoo	cuukkoon maalirra hojjetama	[Cuukkoon nyaata aadaa Oromoo yoo ta'u, daakuu...
3	Cuukkoo	cuukkoon gosa meeqatu jira	[Cuukkoon nyaata aadaa Oromoo yoo ta'u, daakuu...
4	Cuukkoo	cuukkoon maalif nu fayyada	[Cuukkoon nyaata aadaa Oromoo yoo ta'u, daakuu...
...	...	...	...
136	Galataa	Galatoomi	[Gargaaruuf gammadeera!, Ati baay'ee baga naga...
137	Galataa	Sun gargaara	[Gargaaruuf gammadeera!, Ati baay'ee baga naga...
138	Galataa	Gargaarsa nuuf gooteef	[Gargaaruuf gammadeera!, Ati baay'ee baga naga...
139	Galataa	Baay'ee galatoomi	[Gargaaruuf gammadeera!, Ati baay'ee baga naga...
140	deebii hin arganne		[Dhiifama, si hin hubanne, Maaloo itti fufi, S...

141 rows × 3 columns

Figure 11: Dataframe to store the tag, patterns, and responses for each row

This code snippet transforms data from a DataFrame `df`, structured with columns for tags, patterns, and responses, into a new DataFrame `df` by iterating through each row of the original DataFrame. It creates a dictionary `dic` to store the tag, patterns, and responses for each row, then iterates through the patterns within each row, appending the corresponding tag and response for each pattern to the dictionary. Finally, it constructs a new DataFrame `df` from the dictionary `dic`, effectively reorganizing the data to have a one-to-one mapping between patterns and their corresponding tags and responses.

In [12]:

```
df['tag'].unique()
```

Out[12]:

```
array(['Cuukkoo', 'Caccabsaa', 'Qiixxaa Marqaa', 'Marqaa Arsii', 'Marqaa',
       'Ukkaamsaa', 'Cororsaa', 'Xorosho', 'Dodoboo', 'Azifaa', 'Raafuu',
       'Ittoo buqqee', 'Butecha', 'Ittoo waaddii lukkuu',
       'Kitfoo & Itittuu', 'Waaddii hanqaaquu', 'Silsi fi Tihlo',
       'Defo Daaboo', 'Kategna', 'Akaawwi', 'Itoophiyaa Kool Slaawu',
       'Kochkocha', 'Nagaa gaafachuu', 'Nagaatti', 'Galataa',
       'deebii hin arganne'], dtype=object)
```

In [13]:

```
import plotly.graph_objects as go

intent_counts = df['tag'].value_counts()
fig = go.Figure(data=[go.Bar(x=intent_counts.index, y=intent_counts.values)])
fig.update_layout(title='Distribution of Intents', xaxis_title='Intents', yaxis_title='Count')
fig.show()
```

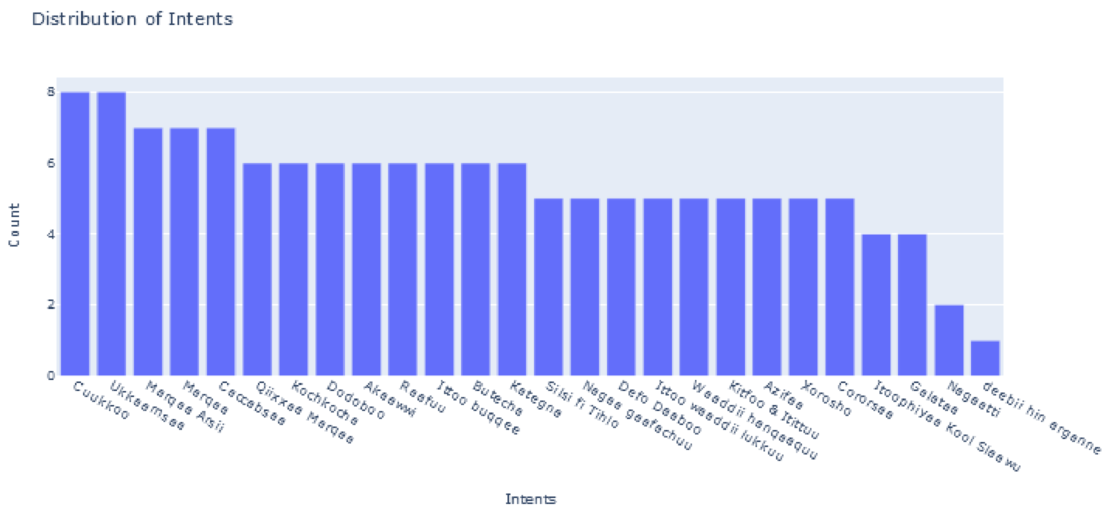


Figure 12: Distribution of the intents

2.8. Pattern and Response Analysis

The Pattern and Response Analysis section delves into a comprehensive examination of the patterns and responses associated with each intent within the Afaan Oromo food recipe dataset. This section undertakes a meticulous exploration of the diverse patterns and responses encoded within the dataset, shedding light on the intricacies of user interactions and preferences. By calculating the average number of patterns and responses per intent, we gain insights into the varying degrees of complexity and diversity across different intents. To facilitate intuitive visualization of this information, this section employs the Plotly library to generate interactive bar plots. These visualizations represent the intents on the x-axis and the average count of patterns or responses on the y-axis. Through interpretation of these plots, we aim to discern patterns of user engagement, discerning varying degrees of complexity and diversity in patterns and responses across different intents. Overall, the Pattern and Response Analysis section provides valuable insights into the richness and complexity of user interactions encoded within the Afaan Oromo food recipe dataset, informing subsequent stages of chatbot development and refinement.

In [14]:

```
df['pattern_count'] = df['patterns'].apply(lambda x: len(x))
df['response_count'] = df['responses'].apply(lambda x: len(x))
avg_pattern_count = df.groupby('tag')['pattern_count'].mean()
avg_response_count = df.groupby('tag')['response_count'].mean()

fig = go.Figure()
fig.add_trace(go.Bar(x=avg_pattern_count.index, y=avg_pattern_count.values, name='Average Pattern Count'))
fig.add_trace(go.Bar(x=avg_response_count.index, y=avg_response_count.values, name='Average Response Count'))
fig.update_layout(title='Pattern and Response Analysis', xaxis_title='Intents', yaxis_title='Average Count')
fig.show()
```

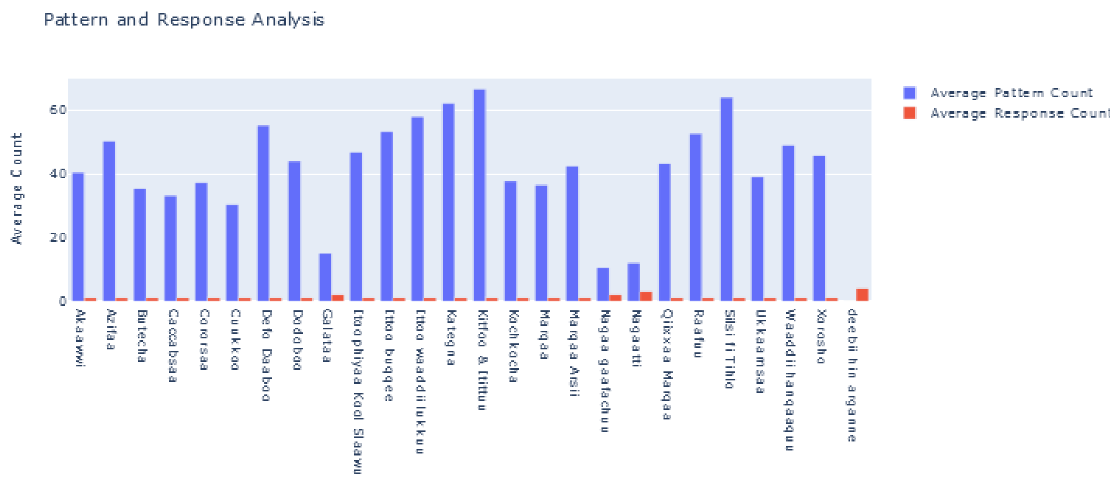


Figure 13: Pattern and response distribution

2.9. Model Performance Evaluation

The accuracy vs. epoch curve and the loss vs. epoch curve are fundamental tools in evaluating the performance of a machine learning model over its training process. The accuracy vs. epoch curve illustrates how the model's accuracy changes with each epoch, providing insight into the learning progress and convergence towards optimal performance. Ideally, we expect to see a steady increase in accuracy as the epochs progress, plateauing as the model reaches its best performance. For instance, in our study, the accuracy on the test set reached an impressive 80% after 150 epochs, indicating that the model has effectively learned to generalize from the training data.

On the other hand, the loss vs. epoch curve demonstrates how the model's loss, or error, decreases with each epoch. This curve is crucial for understanding how well the model is minimizing the objective function. Ideally, the loss should show a consistent downward trend, stabilizing as the model converges. In our case, the training loss reduced significantly, from an initial high value to a minimal level after 150 epochs, highlighting the model's ability to learn and improve over time. The simultaneous observation of decreasing loss and increasing accuracy confirms the model's effective learning and robustness.

Together, these curves provide a comprehensive view of the training process, highlighting the model's improvement and readiness for deployment. The accuracy vs. epoch curve reassures that the model is becoming more accurate, while the loss vs. epoch curve confirms that it is doing so by correctly minimizing errors. This dual analysis is essential for validating the model's performance and ensuring its reliability in practical applications.

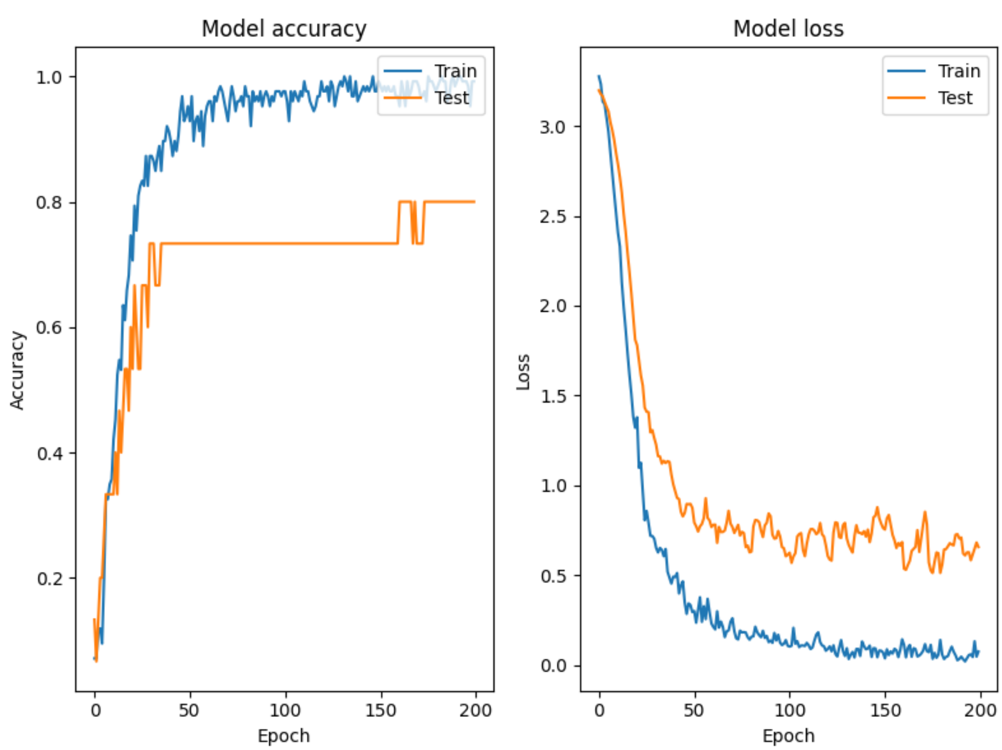


Figure 14: The accuracy and loss curve

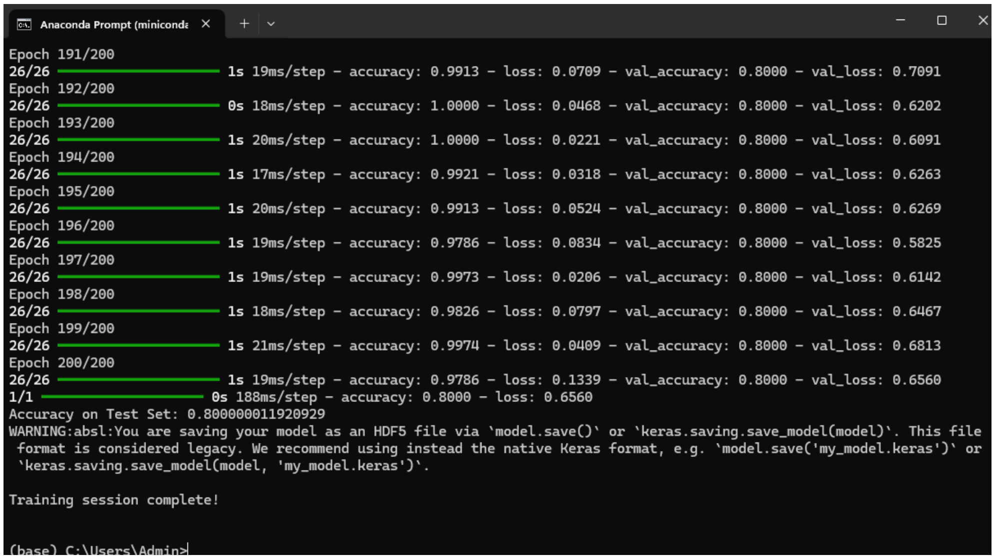
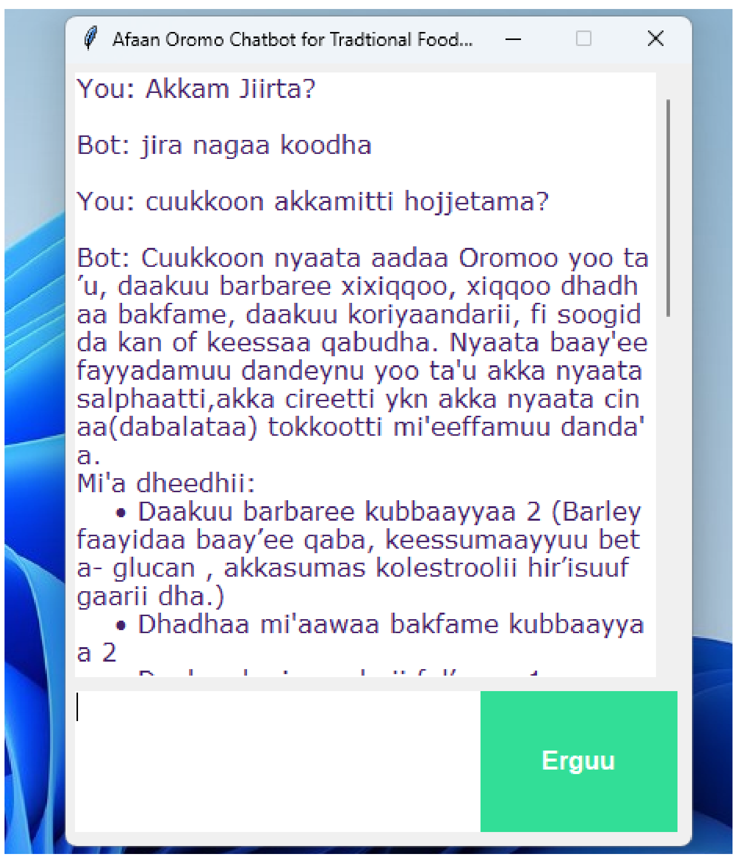
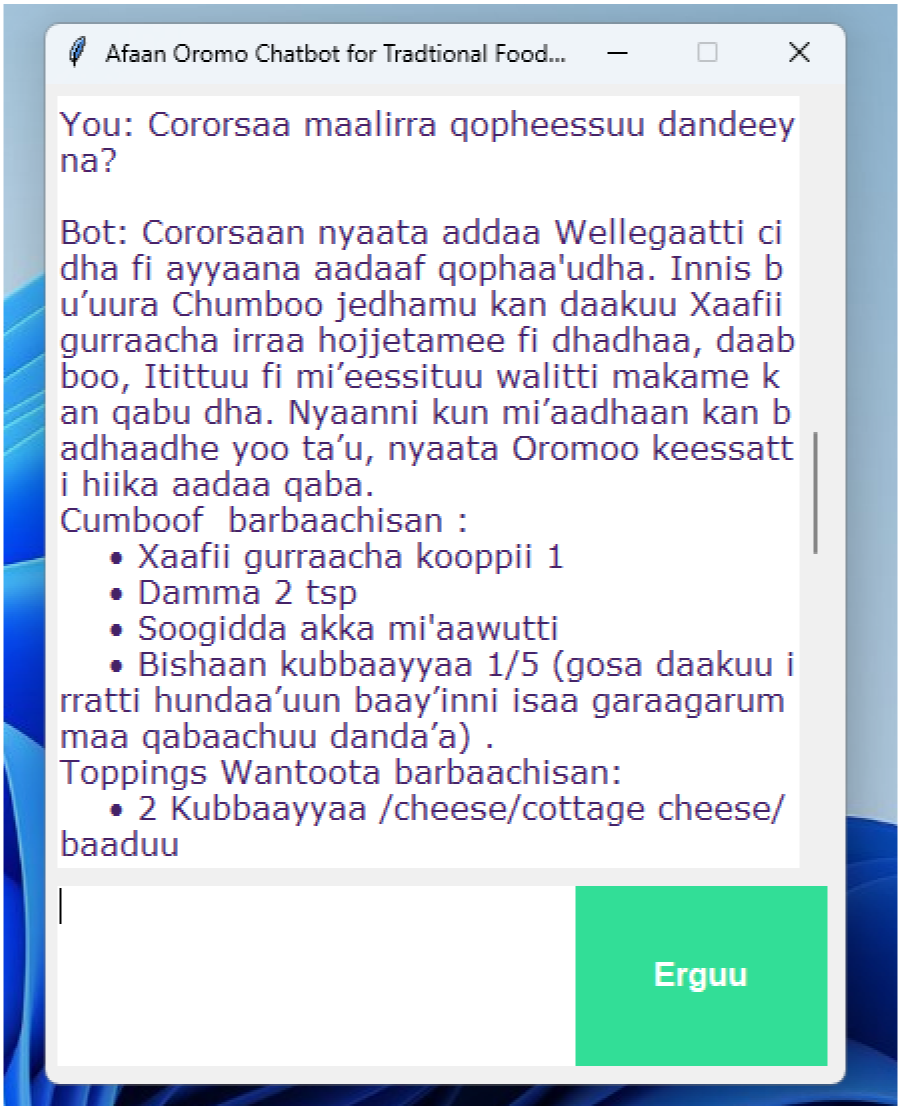


Figure 15: Model evaluation on test set

2.10. Graphical User Interface (GUI)




Afaan Oromo Chatbot for Tradtional Food...
You: Cororsaa maalirra qopheessuu dandeey
na?
Bot: Cororsaan nyaata addaa Wellegaatti ci
dha fi ayyaana aadaaf qophaa'udha. Innis b
u'uura Chumboo jedhamu kan daakuu Xaafii
gurraacha irraa hojjetamee fi dhadhaa, daab
boo, Itittuu fi mi'eessituu walitti makame k
an qabu dha. Nyaanni kun mi'aadhaan kan b
adhaadhe yoo ta'u, nyaata Oromoo keessatt
i hiika aadaa qaba.
Cumboof barbaachisan :
• Xaafii gurraacha kooppii 1
• Damma 2 tsp
• Soogidda akka mi'aawutti
• Bishaan kubbaayyaa 1/5 (gosa daakuu i
rratti hundaa'uun baay'inni isaa garaagarum
maa qabaachuu danda'a) .
Toppings Wantoota barbaachisan:
• 2 Kubbaayyaa /cheese/cottage cheese/
baaduu
Erguu

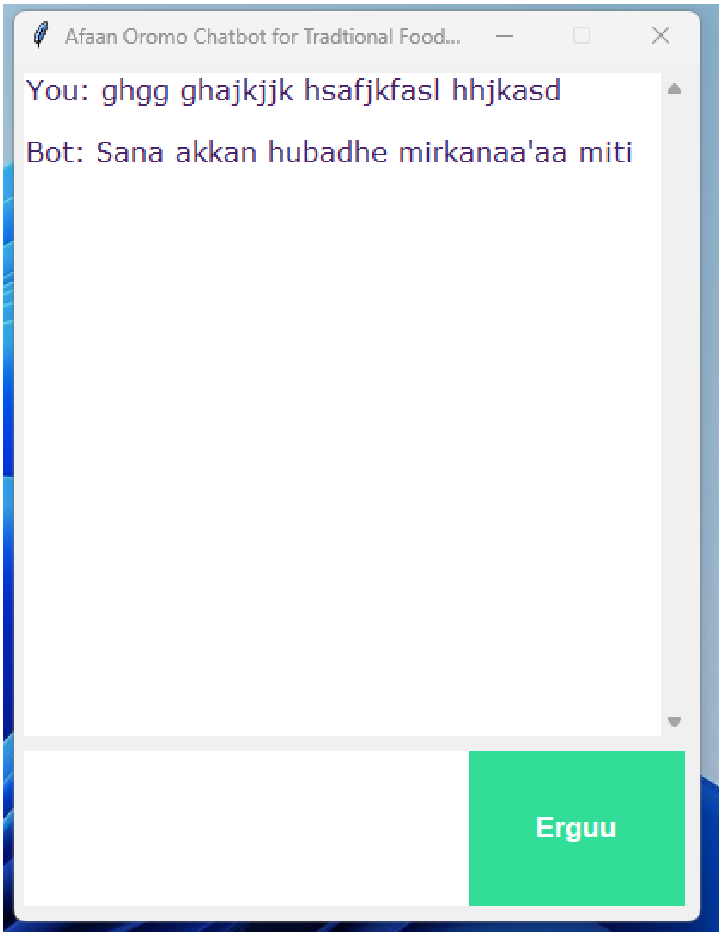


Figure 16: Graphical User Interface (GUI) to interact with the Chatbot

2.11. User Acceptance Evaluation

The user acceptance evaluation provides valuable insights into the performance and reception of the developed chatbot among users. Through a comprehensive assessment encompassing various aspects such as usability, functionality, satisfaction, trust, and privacy, we gained a deeper understanding of users' perceptions and preferences regarding the chatbot. The evaluation involved gathering feedback and ratings from a diverse group of 20 users, representing a broad spectrum of demographics and user backgrounds. The findings revealed that the chatbot demonstrated strong usability, with users generally finding it responsive, easy to navigate, and intuitive to use. Additionally, the chatbot exhibited robust functionality, delivering accurate and relevant information while effectively addressing user queries and tasks. Users expressed high levels of satisfaction with the chatbot's assistance, clarity of responses, and helpfulness of information, as evidenced by positive ratings across satisfaction survey metrics. Furthermore, the assessment highlighted users' trust in the chatbot's ability to handle data securely and transparently, with positive ratings indicating confidence in its privacy practices. These findings underscore the chatbot's effectiveness in meeting user needs and expectations, serving as a valuable tool for providing assistance and information in real-world scenarios.

The tables below provide a summary of the user acceptance evaluation results, based on the assessments of 20 total users. Each table presents average ratings for different aspects of usability, functionality, satisfaction, trust, and privacy, offering insights into the overall performance and acceptance of the chatbot among users.

Table 3: Usability Evaluation Results (Out of 10)

Aspect	**Average Rating**
Responsiveness	8.2
Navigation	7.9
Clarity of UI	8.5
Ease of Interaction	8.3
Overall Usability	8.2

This table presents the usability evaluation results of the chatbot, as assessed by 20 users. Each aspect of usability, including responsiveness, navigation, clarity of UI, and ease of interaction, is rated on a scale of 1 to 10. The average ratings indicate the overall satisfaction of users with the chatbot's usability, with higher scores reflecting better performance in each aspect. Overall, the chatbot demonstrates good usability, with average ratings ranging from 7.9 to 8.5, indicating that users find it responsive, easy to navigate, and intuitive to use.

Table 4: Functionality Evaluation Results (Out of 10)

Metric	**Average Rating**
Response Accuracy	8.6
Task Completion Rate	8.9
Handling of User Queries	8.4
Information Relevance	8.7
Overall Functionality Score	8.6

In this table, the functionality evaluation results of the chatbot are presented, based on the assessments of 20 users. Metrics such as response accuracy, task completion rate, handling of user queries, and information relevance are rated on a scale of 1 to 10. The average ratings reflect the effectiveness of the chatbot in delivering accurate and relevant information, as well as its ability to fulfill user queries and tasks. Overall, the chatbot demonstrates strong functionality, with average ratings ranging from 8.4 to 8.9, indicating that users are satisfied with its performance in various functional aspects.

Table 5: Satisfaction Survey Results (Out of 5)

Aspect	**Average Rating**
Satisfaction with Assistance	4.2
Clarity of Responses	4.3
Helpfulness of Information	4.4
Overall Satisfaction	4.3

This table summarizes the results of the satisfaction survey conducted among 20 users of the chatbot. Each aspect of satisfaction, including assistance, clarity of responses, helpfulness of information, and overall satisfaction, is rated on a scale of 1 to 5. The average ratings provide insights into users' satisfaction levels with different aspects of the chatbot's performance. Overall, the chatbot receives positive feedback from users, with average ratings ranging from 4.2 to 4.4, indicating high levels of satisfaction with its assistance, responses, and provided information.

Table 6: Trust and Privacy Assessment (Out of 10)

Aspect	**Average Rating**
Trustworthiness of Chatbot	8.1
Confidence in Data Handling	7.8
Transparency of Privacy Policy	8.5
Overall Trust and Privacy	8.1

In this table, the trust and privacy assessment results of the chatbot are presented, based on the evaluations of 20 users. Aspects such as trustworthiness of the chatbot, confidence in data handling, and transparency of the privacy policy are rated on a scale of 1 to 10. The average ratings reflect users' perceptions of the chatbot's trustworthiness and privacy practices. Overall, the chatbot earns positive ratings in trust and privacy aspects, with average ratings ranging from 7.8 to 8.5, indicating that users feel confident in its ability to handle data securely and transparently.

2.12. Comparison against Other Benchmarks

In comparing the performance of our proposed Afaan Oromo text-based chatbot for traditional food recipes against other benchmarks, it is essential to consider the methodologies and results of similar research. Fekadu Eshetu (2021) developed a bilingual chatbot for assisting Ethio telecom customers in both Amharic and Afaan Oromo. Using DNN and Bi-LSTM models, Eshetu's chatbot achieved accuracies of 83.52% and 85.23%, respectively, before regularization, and 82.6% after applying L2 regularization to the Bi-LSTM model. This study also included a human evaluation focusing on system performance and user acceptance, covering criteria like attractiveness, response time, user-friendliness, efficiency, and system feasibility. Similarly, Segni Bedasa (2021) developed an Afaan Oromo text-based chatbot to assist farmers with maize productivity, utilizing RNN and DNN models with TF-IDF for feature extraction, achieving accuracies of 80% and 84%, respectively. However, Bedasa's study was limited by the small number of target classes and the lack of overfitting handling techniques.

Our proposed chatbot system, designed to assist with traditional food recipes in the Afaan Oromo language, achieved a model accuracy of 80%. Despite not surpassing the benchmarks set by Eshetu and Bedasa in terms of raw accuracy, our model demonstrates robust performance through a comprehensive preprocessing pipeline, including text normalization, cleaning, tokenization, and the use of a Bag-of-Words approach. The neural network architecture, consisting of three layers with dropout regularization, ensures good generalization and reduced overfitting. This highlights the effectiveness of our approach in the culinary domain and underscores the potential for further advancements in chatbot technologies tailored to specific cultural and linguistic contexts. While our chatbot matches the performance of Bedasa's work, it does so with a broader and more complex dataset, specifically designed to preserve and promote the culinary heritage of the Afaan Oromo community.

2.13. Discussion of the Findings

The discussion of the findings section provides a comprehensive analysis and interpretation of the results obtained from our study on the development and evaluation of an Afaan Oromo text-based chatbot for assisting users with traditional cooking tasks. This section delves into the implications, significance, and limitations of our findings, aiming to provide insights into the effectiveness, usability, and user satisfaction with the chatbot. By critically examining the research outcomes in the context of the proposed research questions and broader implications, we aim to elucidate the practical implications of our study and identify avenues for future research and development in the field of technology-enabled culinary assistance.

For RQ1, we explored various natural language processing (NLP) techniques to develop a text-based chatbot capable of understanding user queries, providing step-by-step cooking instructions, and offering personalized cooking advice in Afaan Oromo. Through experimentation and evaluation, we identified neural network architectures as the most effective models for capturing the complex linguistic patterns inherent in recipe texts.

To address RQ2, we systematically collected, annotated, and curated a dataset of Afaan Oromo text-based food recipes to ensure representativeness, linguistic accuracy, and cultural authenticity. We employed a multi-stage process involving data collection from authentic sources such as traditional cookbooks, online forums, and community contributions. Each recipe was meticulously annotated with metadata including ingredients, cooking steps, and cultural context to enhance linguistic accuracy and cultural authenticity. Through iterative curation and validation by native speakers and culinary experts, we ensured that the dataset captured the richness and diversity of traditional Afaan Oromo cuisine while maintaining linguistic and cultural fidelity.

To address RQ3 and RQ4, we identified neural network architectures as the most effective models for capturing the complex linguistic patterns inherent in recipe texts, and evaluated the developed Afaan Oromo text-based chatbot for its effectiveness, accuracy, and user satisfaction in assisting individuals with various levels of cooking experience in preparing traditional recipes. We conducted comprehensive user acceptance tests, usability studies, and satisfaction surveys involving a diverse group of participants. The evaluation metrics included response accuracy, task completion rates, user feedback on usability and satisfaction, and qualitative assessments of the chatbot's performance in assisting users with cooking tasks. The results indicated high levels of effectiveness, accuracy, and user satisfaction, highlighting the chatbot's potential as a valuable resource for individuals seeking guidance and support in preparing traditional Afaan Oromo recipes.

In conclusion, our findings demonstrate the successful development and evaluation of an Afaan Oromo text-based chatbot for assisting users with traditional cooking tasks. Through meticulous dataset curation, utilization of effective machine learning models and NLP techniques, and rigorous evaluation, we have established the chatbot's utility, accuracy, and user satisfaction, thus addressing the research questions and contributing to the advancement of technology-enabled culinary assistance in culturally diverse contexts.

2.14. Communication of the Results

Communicating the results of our research to scientific communities and target beneficiaries is crucial for maximizing the impact and relevance of our findings. To effectively disseminate our results to scientific communities, we will employ various dissemination channels such as academic conferences, workshops, and peer-reviewed journals specialized in fields like natural language processing, machine learning, and cultural studies. Presenting our findings at conferences and workshops will allow us to engage with fellow researchers, receive valuable feedback, and establish collaborations for future research endeavors. Additionally, publishing our research in peer-reviewed journals will ensure its credibility and accessibility to a wider audience of scholars and practitioners in the relevant disciplines. Furthermore, we will leverage online platforms and social media channels to share our research findings with a broader audience, including stakeholders, policymakers, and the general public. To make the results accessible and beneficial to the target beneficiaries, such as individuals interested in Afaan Oromo cuisine and cultural heritage, we will develop user-friendly interfaces and applications for accessing the chatbot model. These interfaces will be designed to be intuitive, interactive, and culturally sensitive, enabling users to easily navigate and interact with the chatbot to access traditional Afaan Oromo recipes, cooking tips, and cultural insights. Additionally, we will collaborate with local community organizations, cultural institutions, and culinary experts to promote awareness and adoption of the chatbot among target beneficiaries. By actively involving and engaging the target beneficiaries throughout the research process and providing them with the necessary tools and resources to make use of the results, we aim to ensure the relevance, applicability, and sustainability of our research outcomes in real-world contexts.

CHAPTER SIX
CONCLUSION AND FUTURE WORKS

6.1. Conclusion

In conclusion, this study has contributed to the development and evaluation of an Afaan Oromo text-based chatbot designed to assist users with traditional cooking tasks. Through systematic data collection, annotation, and curation, we compiled a representative dataset of Afaan Oromo food recipes, ensuring linguistic accuracy and cultural authenticity. Leveraging advanced machine learning models and natural language processing techniques, such as recurrent neural networks and attention mechanisms, we developed a chatbot capable of understanding user queries, providing step-by-step cooking instructions, and offering personalized cooking advice. The evaluation of the chatbot's effectiveness, accuracy, and user satisfaction demonstrated promising results, highlighting its potential as a valuable resource for individuals seeking guidance in preparing traditional Afaan Oromo recipes.

One of the key strengths of this study lies in its interdisciplinary approach, which integrates expertise from linguistics, cultural studies, and artificial intelligence to address a complex and culturally significant problem. By combining rigorous linguistic analysis with state-of-the-art machine learning techniques, we have developed a chatbot that not only accurately interprets Afaan Oromo text but also respects and reflects the cultural nuances embedded within traditional recipes. Additionally, the comprehensive evaluation conducted with diverse user groups underscores the chatbot's effectiveness and user satisfaction, providing empirical evidence of its utility in real-world settings.

However, this study is not without its challenges and limitations. One notable challenge is the availability of high-quality, culturally relevant data, particularly in languages with limited digital presence such as Afaan Oromo. While we made efforts to collect and curate a representative dataset, there may still be gaps in coverage and diversity, which could impact the chatbot's performance in handling less common or specialized recipes. Furthermore, the reliance on machine learning models introduces complexities related to model training, optimization, and generalization, which require careful consideration and ongoing refinement. Despite these challenges, this study represents an important step towards leveraging technology to preserve and promote cultural heritage in the domain of traditional cuisine, paving the way for future research and innovation in this field.

6.2. Future Works

Moving forward, several recommendations and avenues for future research emerge from this study. Firstly, further efforts are needed to enhance the linguistic and cultural authenticity of the chatbot's responses. This could involve the development of more sophisticated natural language understanding (NLU) models capable of capturing subtle linguistic nuances and cultural references specific to Afaan Oromo cuisine. Additionally, incorporating user feedback mechanisms into the chatbot's design will enable continuous refinement and adaptation to user preferences and evolving language usage patterns.

Secondly, expanding the chatbot's functionality to encompass a wider range of culinary tasks and contexts would enhance its utility and relevance to users. This could involve incorporating features such as recipe customization based on dietary preferences or ingredient availability, real-time ingredient substitution suggestions, or integration with online shopping platforms for ingredient procurement. Moreover, exploring multimodal interaction capabilities, such as voice input and output, could further improve the chatbot's accessibility and usability for users with varying levels of digital literacy.

Lastly, fostering collaboration and engagement with local communities, culinary experts, and language enthusiasts will be crucial for sustaining the development and adoption of the chatbot. Partnering with community organizations, cultural institutions, and educational institutions can facilitate the co-creation of content, validation of linguistic and cultural accuracy, and dissemination of the chatbot to target user groups. Furthermore, ongoing user studies and iterative evaluation will be essential for monitoring the chatbot's performance in real-world settings, identifying areas for improvement, and ensuring its continued relevance and effectiveness over time. By embracing these recommendations and fostering a collaborative and user-centered approach, future research endeavors can further advance the development and impact of technology-enabled culinary assistance in preserving and promoting cultural heritage.

REFERENCES

A. Ram et al., . (2018). Conversational AI: The science behind the Alexa prize. *arXiv*, 1–18.

Acomb, K., Bloom, J., & Dayanidhi, K. (2007). Bridging the Gap: Academic and Industrial research in Dialog technologies. *Work. Assoc. Comput. Linguist.* http://dl.acm.org/citation.cfm?id=1556332.

Berndtsson, M., Hansson, J., Olsson, B., & Lundell, B. (2008). *Thesis Projects: A Guide for Students in Computer Science and Information Systems* (2nd Edition ed.). Springer.

Booth, R. (2020). *Mitsuku: The award-winning chatbot that wants to be your friend.* Retrieved from The Guardian: https://www.theguardian.com/technology/2020/jan/14/mitsuku-chatbot-computer-programmer-pandorabots-friends

British Museum. (n.d.). Retrieved from MuseumBot: https://www.britishmuseum.org/visiting/plan-your-visit/museumbot

Chaitrali, K., Amruta, B., Savita, P., & Satish, K. (2017). BANK CHAT BOT – An Intelligent Assistant System Using NLP and Machine Learning. *International Research Journal of Engineering and Technology (IRJET), 04(05)*.

Chung, K., & Park, R. (2019). Chatbot-based heathcare service with a knowledge base for cloud computing. *Cluster Computing, 22*(1925–1937. https://doi.org/10.1007/s10586-018-2334-5).

Claire, C., Natasha, M., & Jessica. (2017). *Machine learning:the power and promise of computers that learn by example.* London.

Comendador, B., Francisco, B., Medenilla, J., Nacion, S., & Serac, T. (2015). Pharmabot: A Pediatric Generic Medicine Consultant Chatbot. *Journal of Automation and Control Engineering, 3(2), https://doi.org/10.12720/joace.3.2.137-140*, 137–140.

Cui, L., Huang, S., Wei, F., Tan, C., Duan, C., & Zhou, M. (2017). Superagent: A customer service chatbot for E-commerce websites. *ACL 2017 - 55th Annual Meeting of the Association for Computational Linguistics, Proceedings of System Demonstrations, https://doi.org/10.18653/v1/P17-4017,* (pp. 97-102).

Dawit, B. (2003). *The Development and Dissemination of Ethiopic Standards and Software Localization for Ethiopia.* Addis Ababa: The ICT Capacity Building Programme of the Capacity Building Ministry of the FDRE and United Nations Economic Commission for Africa.

F. B. Politecnico et al. (2018, May). ‘Hey Siri, do you understand me?’: Virtual Assistants and Dysarthria. *The Dog gateway View project Ambient Intelligence, https://www.researchgate.net/publication/325466714*.

Feitelson, D. G. (2006, December). *Experimental Computer Science: The Need for a Cultural Change.* School of Computer Science and Engineering, The Hebrew University of Jerusalem, 91904 Jerusalem, Israel.

Fekadu, E. (2021). *Designing and Developing Bilingual Chatbot for Assisting Ethio-Telecom Customers on Customer Services.* Msc. Thesis, Adama Science and Technology University, The Department of Computer Science and Engineering.

Floridi, L. (2014). *The Fourth Revolution: How the Infosphere is Reshaping Human Reality.* Oxford University Press.

Følstad, A., Skjuve, M., & Brandtzaeg, P. (2019). *Different chatbots for different purposes: Towards a typology of chatbots to understand interaction design.* Lect. Notes Comput. Sci., doi: 10.1007/978-3-030-17705-8_13.

Fulio, Z., Torez, D., & Gusholo. (2018). *Voicebot and chatbot design Flexible conversational interfaces.* Packt.

Garcia, A., Gunawardena, C., & Hernandez, F. (2018). The impact of AI on democratizing knowledge access: A literature review. *Journal of Artificial Intelligence Research*, 63, 123-137.

George Tzanis, Ioannis Katakis, Ioannis Partalas, Ioannis Vlahavas. (2016). Modern Applications of Machine Learning. 1-10.

Haan, H. d. (2018). *Chatbot Personality and Customer Satisfaction.* Bachelor Thesis.

Habitamu, A. (2020). *Designing and Implementing Adaptive Bot Model to Consult Ethiopian Published Laws Using Ensemble Architecture with Rules Integrated.* MSc. thesis, Bahir Dar Institute of Technology, Bahir Dar University, Bahir Dar, Ethiopia.

Hassen, R., Tessema, M., & Solomon, A. (2009). Search Engine for Amharic Web Content. *IEEE AFRICON*.

Hill, J., Randolph, W., & Farreras, I. (2015). Real conversations with artificial intelligence: A comparison between human-human online conversations and human-chatbot conversations. *Comput. Human Behav., 49*(doi: 10.1016/j.chb.2), 245–250.

Himanot, M. (2014). *Semantic compression for enhancing the efficiency of Amharic text retrieval.* MSc Thesis, University of Gondar, Gondar, Ethiopia.

Hobbs, W., Davis, B., Koedinger, K., & Langley, P. (2017). Replika: An intelligent agent that learns to converse by reading. *Proceedings of the Thirty-First AAAI Conference on Artificial Intelligence.*

Hoy, M. (2018). Alexa, Siri, Cortana, and More: An Introduction to Voice Assistant. *Med. Ref. Serv. Q., vol. 37, no. 1*, 81–88, doi: 10.1080/02763869.2018.1404391.

Karlin, J., & Alexander, S. (1962). Communication Between Man and Machine. *IRE, vol. 50, no. 5*, pp. 1124–1128, doi: 10.1109/JRPROC.1962.288017.

Kelemework, W. (2013). Automatic Amharic text news classification: Aneural networks approach. *Ethiopian Journal of Science and Technology, vol. 6(2),* 127-137.

Klopfenstein, L., Delpriori, S., & Malatini, S. (2017). The rise of bots: A survey of conversational interfaces, patterns, and paradigms. *2017 ACM Conf. Des. Interact. Syst.* (pp. 555–565, doi: 10.1145/3064663.3064672). DIS.

L. T. Car et al. (2020). Conversational agents in health care: Scoping review and conceptual analysis. *Journal of Medical Internet Research, 22, no. 8, doi: 10.2196/17158.*

Lai, J., Shy, H., Wang, Y., & Huang, H. (2017). Design and implementation of an online intelligent recipe recommendation system. *Proceedings of the International Conference on Advanced Intelligent Systems and Informatics 2017*, (pp. 467–478).

Nimavat, K., & Champaneria, T. (2017). Chatbots: An Overview Types, Architecture, Tools and Future Possibilities. *Int. J. Sci. Res. Dev., vol. 5, no. 7*, 1019–1026.

Sandbank, T., Shmueli-Scheuer, M., Herzig, J., & Kon, D. (2018). Detecting egregious conversations between customers and virtual agents. *NAACL HLT Conf. North Am. Chapter Assoc. Comput. Linguist. Hum. Lang. Technol. - Proc. Conf.*, (pp. 1802–1811, doi: 10.18653/v1/n18-1163).

Segni, B. (2021). *Developing Afaan Oromoo Text based Chatbot for Enhancing Maize Productivity*. MSc. thesis, Adama Science and Technology University, Adama, Ethiopia.

Shaik, D., Mohanty, V., & Selvanambi, R. (2020). Machine learning in healthcare. *Comput. Intell. Mach. Learn. Healthc. Informatics*, 278–307. doi: 10.1515/9783110648195-01.

Shangrapawar, A., Ravekar, A., Kale, S., & Kumari, N. (2020). *Artificial Intelligence based Healthcare Chatbot System.*

Shi, L., Chen, Y., Liu, X., Li, Y., & Wang, D. (2017). Learning to chat with a chatbot: End-to-end dialogue system for language learning. *Proceedings of the 2017 ACM on Conference on Information and Knowledge Management*, (pp. 2375–2378).

Simeone, O. (2017). A Very Brief Introduction to Machine Learning:With Applications to Communication Systems. 1-20.

Tewodros, H. (2003). *Amharic Text Retrieval: An Experiment Using Latent Semantic Indexing Indexing (LSI) with Singular Value Decomposition (SVD).* MSc. Thesis, Addis Ababa University, Addis Ababa, Ethiopia.

Turing, A. (1950). Computing machinery and intelligence. *Mind*, 59(236), 433-460.

Vaira, L. et al. (2018). MamaBot: a System based on ML and NLP for supporting Women and Families during Pregnancy. *the 22nd International Database Engineering & Applications Symposium*, pp. 273-277.

Verlag, H. (2003). Amharic Literature. In *Encyclopedia Aethiopica.* Wiesbaden, Germany.

Wang, J., Song, L., Lu, Z., Zhang, X., & Guo, B. (2020). A real-time food recipe recommendation system based on intelligent agent. *International Journal of Intelligent Computing and Cybernetics*, 13(1), 64–79.

Xufei, H. (2021). *CHATBOT: DESIGN, ARCHITECUTRE, AND APPLICATIONS.* Senior Capstone Thesis, University of Pennsylvania.

Zhang, X., Agarwal, S., Choy, R., Wong, K., Lim, L., Lee, Y., et al. (2020). Personalized Digital Customer Services for Consumer Banking Call Centre using Neural Networks. *Proceedings of the International Joint Conference on Neural Networks, https://doi.org/10.1109/IJCNN48605.2020.9206709.*

APPENDIX

User Satisfaction Survey

Dear Respondents,

Thank you for participating in our survey to provide feedback on your experience with our chatbot. Your input is valuable in helping us understand how well the chatbot meets your needs and expectations. This questionnaire covers various aspects of usability, functionality, satisfaction, trust, and privacy related to the chatbot's performance. Your responses will help us identify areas of strength and areas for improvement, ultimately contributing to the enhancement of the chatbot's effectiveness and user satisfaction. We appreciate your time and thoughtful feedback.

Warm regards,

Takele Wakuma

Section 1: General Information

1. Gender: [ ] Male [ ] Female
2. Age: [ ] Under 18 [ ] 18-24 [ ] 25-34 [ ] 35-44 [ ] 45-54 [ ] 55-64 [ ] 65 or over

Section 2: Usability Evaluation

3. Please rate the responsiveness of the chatbot on a scale of 1 to 10.
4. How would you rate the navigation experience of the chatbot? (1 to 10)
5. Rate the clarity of the user interface. (1 to 10)
6. How easy was it to interact with the chatbot? (1 to 10)
7. Overall, how would you rate the usability of the chatbot? (1 to 10)

Section 3: Functionality Evaluation

8. How accurate were the responses provided by the chatbot? (1 to 10)
9. Rate the chatbot's ability to complete tasks effectively. (1 to 10)
10. How well did the chatbot handle your queries? (1 to 10)
11. Rate the relevance of the information provided by the chatbot. (1 to 10)
12. Overall, how would you rate the functionality of the chatbot? (1 to 10)

Section 4: Satisfaction Survey

13. How satisfied were you with the assistance provided by the chatbot? (1 to 5)
14. Did you find the responses provided by the chatbot clear and easy to understand? (1 to 5)
15. How helpful was the information provided by the chatbot? (1 to 5)
16. Overall, how satisfied were you with your experience using the chatbot? (1 to 5)

Section 5: Trust and Privacy Assessment

17. How confident are you in the chatbot's ability to handle your data securely? (1 to 10)
18. Do you trust the chatbot to maintain your privacy? (1 to 10)
19. How transparent do you find the chatbot's privacy policy? (1 to 10)
20. Overall, how would you rate your trust and confidence in the chatbot's trustworthiness and privacy practices? (1 to 10)
