Developing Afaan Oromoo Text based Chatbot for Enhancing Maize Productivity

By:

Segni Bedasa Dilbo

A Thesis Submitted to the Department of Computer Science and Engineering

School of Electrical Engineering and Computing

Presented in Partial Fulfillment for the Degree of Masters of Science in Computer Science and Engineering

Office of Graduate Studies

Adama Science and Technology University

January 2021

Adama, Ethiopia

Developing Afaan Oromoo Text based Chatbot for Enhancing Maize Productivity

By:

Segni Bedasa Dilbo

Name of Advisor: Mesfin Abebe(Ph.D.)

A Thesis Submitted to the Department of Computer Science and Engineering

School of Electrical Engineering and Computing

Presented in Partial Fulfillment for the Degree of Masters of Science in Computer Science and Engineering

Office of Graduate Studies

Adama Science and Technology University

January 2021

Adama, Ethiopia

Declaration.

I declare that the study has been on papered by myself and that no other submitted professional qualification for the research. I announce that the research paper succumbed is mine, with the exception of where the study involved in together-authored publications.

Name: Segni Bedasa Dilbo

Signature:____

The thesis presented for assessment with my authorization as a thesis advisor.

Name :- Mesfin Abebe (Ph.D.)

Signature :-____

Submission Date:__________________________________

Approval of Board of Examiners

We, the members of the board of examiners of the final open defense by *Segni Bedasa*, have read out and assessed his/her thesis entitled "*Developing Afaan Oromoo Text-based Chatbot for enhancing maize productivity*" and examined the candidate. This is, therefore, to certify that the thesis has been accepted in partial the fulfillment of the requirement of the Degree of Computer Science and Engineering.

(Supervisor/Advisor)	(Signature)	(Date)
(Chairperson)	(Signature)	(Date)
(Internal Examiner)	(Signature)	(Date)
(External Examiner)	(Signature)	(Date)

Acknowledgment

I would like to acknowledge God for his grace, strength and good health as I undertook this research. I would like to thank my esteemed Advisor Dr. Mesfin Abebe for his invaluable supervision, support , his friendly approach, readiness and willingness to advise on the research, his critical comments, commitment to guide, and support greatly improved this thesis work.

I would also like to acknowledge the members of Smart Software SIG of the CSE department. Including the SIG supervisor Dr. Mesifn Abebe for his continued commitment to guide and support this research from its inception to completion. And to all members who sat on each phase presentation panels for their inputs which greatly shaped and improved the work. I would like to thanks all my classmates who are working on their master's thesis for their timely help, idea, and support until the completion of my thesis.

My gratitude extends to the School of Electrical and Computing for the funding opportunity to undertake my studies at the Department of Computer Science and Engineering , Adama Science and Technology University . Additionally, I would like to express gratitude to the staff of ATA , Oromia Research center and Oromia agriculture bureu for providing Afaan Oromoo Text data for enhancing maize product.

Dedication

The dedication of my research study is to my family. An extraordinary sense of thankfulness to my Dady, Bedasa Dilbo who passed during my 1st year of graduate study , my dady always inspired and guide me to work hard on my study until he passed.

Table of Contents

List of Tables

List of Figures

List of Equation

List of Abbreviations

NLP	Natural language processing
FAO	Food and Agriculture Organization
SMS	Short Message Service
RQ	Research Question
KNN	K-nearest neighbor
AI	Artificial Intelligence
KCC	Kisan Call Center
DNN	Deep Neural Network
RNN	Recurrent Neural Network
Seq2Seq	Sequence to Sequence
AO	Afaan Oromoo
SIG	Special Interest Group
CSE	Computer Science and Engineering
ATA	Agricultural Transformation Agency
ML	Machine Learning

Abstract

Artificial neural networks are at the basis of the most state-of-the-art for a variety of activities and have enjoyed great success in a variety of tasks such as machine translation, Chatbot, image recognition, speech recognition, text summarization, and in another computation. Neural networks with recurrent model, have recently been applied to chatbot and begun to show successive results with different natural language llike English , Turkish language. How ever there is no any research conducted Afaan Oromoo text-based Chatbot. Developing Afaan Oromoo text-based Chatbot is very important, since our society have a lot problem that needs solution to solve. Since most of Ethiopia peoples are farmers there many problems that needs to solve especially in area of agricultutre.To live people must eat, to eat they must enhance their productivity , to enhance their productivity they need information how to enhance their product. In our country accessing information at right time is difficult. So, we proposed Afaan Oromoo Text-based Chatbot for enchancing maize productivity based neural network algorithm. The main goal of this research is to develop model which fit Afaan Oromoo Chatbot to solve the problem that the our society arise. We proposed two Deep learning algorithms Deep Neural Network and Recurent neural network which is based up on retrieval approach. Retrieval based Chatbot approach is approach which depend on the predefined data. We conducted human testing approach for our result, we got good performance in both model. The result from DNN model is **84.4**% testing accuracy value where as experimented result from RNN based Seq2Seq model was **80%** testing accuracy.

Generally,the experiment result show us developing Afaan Oromoo text-based Chatbot is good solution to help farmer for enhacing their product, because farmer can communicate with the bot at any time they need.

Keywords: Artificial neural network, Afaan Oromo, Retrieval based approach, Natural Language Processing, Chatbot, Seq2Seq , Deep neural network , Recurent neural network , farmer and Deep learning

CHAPTER ONE

1. Introduction

1.1. Background of study

Natural language processing (NLP) is a field of Computer Science, Artificial Intelligence, and Computational Linguistics concerned with the interactions between computers and human (natural) languages. As such, NLP is related to the area of human-computer interaction. Many challenges in NLP involve natural language understanding, that is, enabling computers to derive meaning from human or natural language input, and others involve natural language generation

A natural language refers to human languages (Amharic**, Afan Oromo**, Tigrigna, English, Arabic, Chinese, etc.), as opposed to programming languages such as C++, Java, Pascal, etc. A natural language is represented using texts in spoken or written forms. The goal of NLP is to accomplish human-like language processing for various tasks and applications such as machine translation, information retrieval, question-answering, etc.

Natural language processing with the field of Artificial Intelligence and Machine learning is used in many areas like Agriculture, health, education, smart cities, smart industries, and smart environments. The agriculture sector is the most known sector in Ethiopia. Most of the Ethiopian people were living in a rural area. In Ethiopia, about 83.9 % of the total population is lives in a rural area and agriculture is the main source of their livelihood[1]. Even if the people of the country we're living in a rural area there are a lot of problem-related to the Agriculture sector in our country Ethiopia. Those problems are a scarcity of food security, poverty, and unbalanced productivity are among the problem-related Agriculture Sector in Ethiopia. The reason for the above problem is the policy of the country, geographically area and lack of knowledge on technology on the farm. In my concept, I'm focusing on how technology can enhance agriculture productivity to solve the above problem in our country Ethiopia in case West Oromia region.

In the current, farmers get assistance with the help of government workers and field officers in the form of training. This way of supporting farmers will not solve the problem above explained. In the current according to Food and Agriculture Organization of the United Nations (FAO) represents one of the large specialized agencies of the United Nations striving to alleviate poverty and hunger by ensuring better agricultural development, food security and good nutrition for all. In Ethiopia, FAO will continue to work with the Government, international donors and other development partners with a view to eliminating hunger and malnutrition from Ethiopia. This organization also defined the basic variable that affects the agriculture productivity in Ethiopia. Those variables use fertilizer, use of pesticides, Manure, use of seeds and use natural resources like soil, water, and disease that affect crop productivity. Food and Agriculture Organization of the United Nations (FAO) discuss two points to solve the problem above explained. Their point of the solution is 1. Agricultural Policy and Regulatory Frameworks 2. Agricultural Information and Knowledge Management

My concern is how the improvement of technology will increase agriculture productivity? There are different technologies that related to agriculture productivity

Those are:

1. Digital Technologies: including the Internet, mobile technologies and devices, data analytics, artificial intelligence, digitally-delivered services, and apps—are changing agriculture and the food system
2. farm machinery automation allows fine-tuning of inputs and reduces demand for manual labor
3. remote satellite data improve the accuracy and reduce the cost of monitoring crop growth and quality of land or water

From the technologies that are very important in increasing agriculture productivity above discussed digital technologies is one of the important technologies that improve agriculture productivity in today's world.

A chatbot is one of the digital technologies which is very important to improve agriculture productivity. A chatbot is a piece of software businesses can use to communicate with potential customers. A chatbot can act as a virtual assistant for your business. They use a conversational, relatable style to interact with customers, allowing agriculture companies to improve customer service, productivity, and operational efficiency. Bots are incredibly useful in data-intensive industries like farming because they can help streamline and automate tasks that were previously only performable by strict counting and old-fashioned human analysis. I want to design Afan Oromo text-based chatbot for the farmer to increase maize production in case Oromia. Maize production is well known in Oromia even though the doesn't get good production because there are different variables that affect the productivity of the farmer.

1.2. The History of Chatbot

To put it simply, chatbots are computer applications powered by artificial intelligence to communicate with you. They are designed to mimic natural human conversation via an online chat interface, SMS, and sometimes, even voice chat. Despite being designed for human consumption, chatbots are also capable of communicating and gathering information from other chatbots. The idea of creating a machine that had humanlike thought processes has been around for centuries. Scientists, philosophers, and even sculptors were fascinated by the idea of a humanistic automaton capable of thought. The history of chatbots can be traced way back to 1950 when Alan Turing published his paper "Computing Machinery and Intelligence"[2].

Table 1: The History of Chatbot

No	*Chatbot Development*	*Year*
1.	Alan Turing published the first paper about the intelligent machine that can chat with a human	1950
2.	Joseph Weizenbaum was developed the first Chatbot (ELIZA)	1966
3.	*Kenneth Colby came out with PARRY, a chatbot that could simulate a person with paranoid schizophrenia*	*1972*
4.	*Artificial Linguistic Internet Computer Entity was developed by Richard Wallace*	*1995*
5.	*Peter Levitan was found Smarterchild*	*2001*
6.	Siri was founded by Apple	2010
7.	*Google Now was developed by google company for google voice search*	*2012*
8.	*Alexa which was developed by Amazon*	*2015*
9.	*Cortana Which was developed by Microsoft*	*2015*

1.3. The motivation of the Study

Under this section we explain why we motivated to conducted this study. There are main reason for our motivation to conduct this study. The problem around agriculture in oromiyaa espesically in lack of technology for enhance their productivity. The second reason for our motivation is that improving Afaan Oromoo for Technology. Ethiopian farming largely produces only enough food for the peasant holder and his family for consumption, leaving little to sell. This inadequate volume of production is ascribed to the tardy progress in the farming methods and scattered pieces of land holdings. Under this traditional sector, agriculture is practiced on public land and most of the produce is mainly for consumption[3][4]. The diverse climate of the country and the multiple utilization of crops have prompted the vast majority of agricultural holders to grow various temporary and permanent crops.Cereals are grown in almost all regions of Ethiopia with notable variation in the extent of areas planted and the volume of production obtained. This variation is seemingly caused by a shift in choice of crops by the holders and difference in weather conditions. Maize is one of the cereal crops which has high volume in planted area especially in Oromiyaa[5]. In oromiya regional maize productivity covers 22.90% with 34.40% yields productivity respectively. Even though maize is the large volume crop productivity in oromiyaa region, the wasted resource and gained product was not matched. From many resource small product. In our country concept, there is a mismanagement of resources like soil, water and human labor in agriculture fields. In the other almost all people of Ethiopia or 80% are farmers. So, we believe that changing the agricultures from traditional to using technology can enhance agriculture productivity. Afaan Oromo is one of the most widely used languages in Ethiopia, and also spoken in Kenya, Somalia, and Djibouti. Nowadays, textual information is highly increasing from time to time both in softcopy and hardcopy format in Afaan Oromoo, to change the level any this language we should adapt Afaan Oromoo with technology.

1.4. Statement of the Problem

Currently, the Agriculture sector will work on to enhance productivity. Even though they are doing to enhance productivity most of the way to increase productivity is manual processing. Especially Farmers will get support through field officers and agriculture sector training. This way of getting support will not efficiently and the capabilities of enhancing or increasing agriculture productivity are very less. The reason behind the way of getting agricultural support for farmers will not efficient are the following:

1. Geographically location of the farmers: in the rural area of our country Ethiopia the way of living is dispersed.
2. Human labor: In current for each kebele of the country only one(1) field officer for each agricultural field, so the number of field officers and farmers opposite proportion
3. Time wasting: Wasting time for training
4. Mismanaging of agricultural Input (seeds, fertilizers)

Currently, the world is all about information, most of it is in digital format. The World Wide Web contains billions of documents and it is growing at an exponential pace [5]. The amount of data we produce every day is truly mind-boggling. According to Forbe's report, there are 2.5 quintillion bytes of data created each day at our current pace, but that pace is only accelerating with the growth of the Internet of Things (IoT). Over the last two years alone 90 percent of the data in the world was generated. This is worth re-reading! While it's almost impossible to wrap your mind around these numbers, we gathered together some of our favorite stats to help illustrate some of the ways we create these massive amounts of data every single day. More than 3.7 billion humans use the internet (that's a growth rate of 7.5 percent over 2016). On average, Google now processes more than 40,000 searches EVERY second (3.5 billion searches per day)! While 77% of searches are conducted. Obviously, this explosion of information has caused a well-recognized information overload problem. For this matter, it is obvious that there is no time to get target response based on available information. Therefore, for this problem, there must be tools that provide timely access to, and digest of, various sources are necessary in order to alleviate the information overload people are facing. Chatbot is a technology for extracting target response from a given document in the form of text and audio by using different chatbot modeling approach. Afaan Oromo is one of the most widely used languages in Ethiopia, and also spoken in Kenya, Somalia, and Djibouti. Nowadays, textual information is highly increasing from time to time both in softcopy and hardcopy format. The textual information disseminated to the populations located at various countries is provided by various sources, such as; Internet, media, presses, books, journal articles, newspaper, etc. Even though Afaan Oromo is among the major languages in Africa with a lot of vocabulary, it lacks research work on the Area of Chatbot technology. There is no research paper that conducted on Afaan Oromoo text based Chatbot until this research work conducted. On the other side even if there is no Afan Oromo text-based chatbot technology that related to the agriculture field especially crop productivity there is some research that is conducted on Chatbot technologies. Most of the research that conducted on Chatbot technology on agricultural fields were based on ruled based which is not advance when compared with machine learning To improve the above problem Agronomy (Agricultural Technologies) the best solution.

There are a lot of Technology solutions to enhance agricultural crop productivity. For example,

1. Digital Technologies: including the Internet, mobile technologies and devices, data analytics, artificial intelligence, digitally-delivered services, and apps—are changing agriculture and the food system
2. farm machinery automation allows fine-tuning of inputs and reduces demand for manual labor
3. remote satellite data improve the accuracy and reduce the cost of monitoring crop growth and quality of land or water

From the above Technology solution to enhance productivity, I will try to explain how Digital Technology can enhance agricultural crop productivity. From digital Technology how chatbot technology can enhance crop productivity. I will propose Developing Afan Oromo Text-based Chatbot to enhancing maize productivity.

To solve the problem that exists in the current situation the following question raised.

RQ1. How to select the best modeling approach to develop a Chatbot application in the agriculture area?

RQ2. How to select better model which fit for proposed solution ?

RQ3. Is developing Afan Oromo chatbot is better access to information for a farmer? This question is answered developing the technology and comparing with the current way of getting and accessing information

1.5. Objective of the Study

1.5.1. *General Objective*

The general objective of the study is developing Afan Oromo text-based Chatbot for enhancing maize productivity

1.5.2. Specific Objectives

To achieve the general objective of the study, the following mentioned specific objective has to be addressed:

- To Review literature and related work a literature review on the Chatbot application area
- To Review literature and related work on Crop production in Ethiopia .
- To identify an appropriate algorithm for Afaan Oromo for Chatbot technology.
- Identifying which type of Chatbot is better to apply to the agriculture area.
- Develop Afan Oromo Corpus(dataset)
- Designing and developing Afaan Oromoo Chatbot

- Evaluate the response
- To state conclusion and recommendation based on the experimental results.

1.6. The methodology of the Study

To attain the above objectives and answer the research questions that explained above, the following methods and techniques followed.

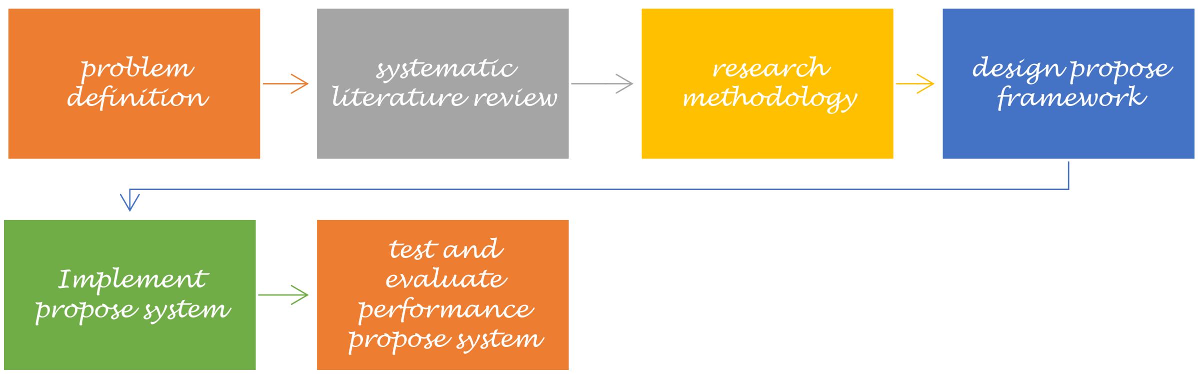


Figure 1: The flow of the research process

We mention the following steps as a research methodology

- First, Formulating by reviewing different literature review and survey paper within different angles like area perspective problem and technology perspective
- After problem are well defined, collecting data by different data collection techniques like document review and from domain expert guidelines
- Next, mention the research methodologies such as systematic literature review, data collection method, chatbot recognition method, propose framework design method and design and implementation tools used for the research work
- Designing Afan Oromo text-based chatbot for enhancing maize productivity
- Implementing the designing system by using a machine learning algorithm
- Finally, evaluating and testing the performance of a designing system

1.6.1. Modeling Approach

There are three different chatbot modeling approaches are there. Those are the Rule-based approach, Retrieval Approach, and Generative based Approach. In this thesis, I have a plan to use the Retrieval modeling approach. Retrieval-based dialogue systems, do not generate responses. For each context, they find the best response among a set of predefined responses (called candidate responses). For each candidate response, a matching score with the context is computed and then used in order to rank the candidate responses. The response with the highest score is then chosen. To model, this approach different techniques are employed. Those are

- Parsing
- Pattern Matching

Parsing**:** Parsing is a method that takes the text as an input and extracts meaningful information from it by converting that text into a set of more uncomplicated words (lexical parsing) that can be easily stored and manipulated.

Pattern Matching**:** Pattern matching is the most commonly used approach in chatbots to classify the user input as <pattern> and produce a suitable response stored in <template>. These <pattern> <template> pairs are handcrafted. Although pattern matching techniques are used in both early and modern catboats, the complexity of the algorithms used in them differs.

1.6.2. Development Tools

Anaconda: Anaconda is an open-source distribution for python and R. It is used for data science, machine learning, deep learning, etc. Anaconda helps in simplified package management and deployment. Anaconda comes with a wide variety of tools to easily collect data from various sources using various machine learning and AI algorithms.

Jupyter Notebook**:** "notebook" or "notebook documents" denote documents that contain both code and rich text elements, such as figures, links, equations, Because of the mix of code and text elements, these documents are the ideal place to bring together an analysis description, and its results, as well as, they can be executed perform the data analysis in real-time.

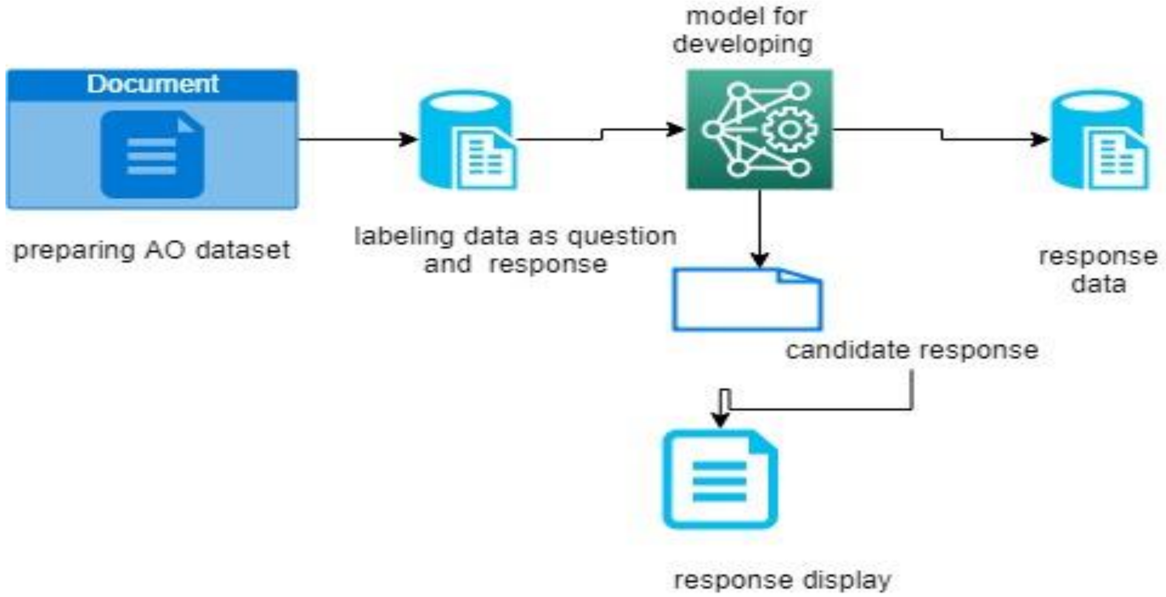


Figure 2: Flow of proposed System

User Input: - A user can give an input message to the system.

Pre-Processing: - consist of many steps like removal of stop works, tokenization, and conversion in lower case, etc.

Intent Classification: - The intent classification module identifies the intent of the user message.

Entity Recognition: - Entity Recognition module extracts structured bits of information from the message.

Response Generator: - The response generator is doing all the domain-specific calculations to process the user request. Response Candidates: - Response candidates are the set of responses which can be the output to the user input.

Response Selector: - The response selector scores the response candidates and selects a response that should work better for the user.

1.7. Scope and Limitations of the study

1.7.1. The scope of the study

The study examines the technical principles of work, specifications, and features of chatbot and also identifies how machine learning can predict conversational chatbot. The study explores the recent level of chatbot in agriculture to enhance productivity. The study depends upon close domain chatbot. Such bots are focused on a particular knowledge domain and might fail to respond to other questions. For example, a restaurant booking bot won't tell you how to enhance crop productivity. The study focuses on how to respond to enhance maize productivity depend on the question of the farmer. During the study, the evaluation of classification accuracy based on classifiers, the classifier machine learning algorithm with the highest accuracy is used as the best classification model. Develop Afan Oromo Text-based chatbot to enhance maize productivity.

1.7.2. The Limitations of the Study

Implementing Chatbot is more challenging and complex. Because of this fact evaluating Generative Chatbot and Open-domain Chatbot is beyond the scope of the study. The study is also geographically limited. It depends upon the Oromia region. Also, the study is not bilingual. It depends on Afan Oromo Text-based developing chatbot for enhancing maize productivity. On the other hand, it doesn't consider other crop products. Even though Chatbot is one of the most important research areas in the current, there are limitations for a chatbot for agriculture to review. The review of the study is based on available literature resources.

1.8. Application Of the results

The study extends the effort in solving how to enhance maize productivity problems. The study contributes it's to parts to enhance maize productivity. The significance of the study to the stakeholder:

- Provides information about maize productivity
- Specificity and localization were identified as keys to the information needs of farmers.
- Give precise answers to the disease-related questions
- Give precise answers for the fertility-related questions
- Give a precise answer for seed type related question
- Act as virtual assistants for the farmer

1.9. Organization oof the Thesis

The thesis organized into seven chapters.

Chapter one gives the overall introductory information of the argument. In this chapter, we try clear up what are the researchable problems, the research hypothesis used to answer relevance of the problem it is addressing the research problem, the purpose of general and specific objective to solve the research problems stated and the general research methodology carried out to accomplish research objectives are.

Chapter two is a literature review. This chapter describes the literature part and source of the research area and to evidence, the hypothesis of the research, different kinds of literature on the problem domain are reviewed. Additionally, this chapter identifies the gap of related paper by establishing a systematic review (systematic literature) approach.

Chapter three is about the research methodology. This chapter discusses more detail about how research methodologies are carried out and more descriptions and the logical reasoning of the morphology described in this chapter.

Chapter four is about the proposed solution of the Afaan Oromoo text-based Chatbot for enhancing maize productivity. Step by step description of how the proposed solution address the problems stated in the research problem using the Deep Neural Network and Recurent Neural Network developed and we discuss how the models are work in this chapter.

Chapter five is about the implementation and expperimetation of the DNN and RNN based Seq2Seq model for Afaan Oromoo Text-based Chatbot. Description of how the DNN and RNN based Seq2Seq model is developed and how it works is describe in this chapter.

Chapter six explain the discussion and result of the proposed solution compared with previous result findings drawn from this research. The effectiveness, of the proposed solution(model) of this conducted research is also described in this discussion and result chapter.

Chapter seven concludes the research and provide an insight into the recommendations and valuable suggestions from the author. The impacts of this research are also described in this concluding chapter.

CHAPTER TWO

2. Literature Review and Related Works

This chapter reviews relevant literature to further comprehend the concept and investigate the research problem. The review provides the chatbot application area, types of Chatbot, Algorithm which is better for a chatbot, defining the terminology which used in chatbot environments like intents and entities and this review provide the characteristics of a chatbot.

2.1. Chatbot

Definition 1 – English Wiktionary, “(Chat ro**bot**) Software that provides a text or verbal interaction with a person using native language”. Also called a "chatterbot," the chatbot is designed to emulate normal human responses. Chatbots can be very limited in scope, although they may be able to improve with use. The terms "chatbot" and "virtual assistant" are increasingly used synonymously; however, Chatbots preceded virtual assistants and generally provide suggestions or answers to questions about a specific topic or product. Chatbots may also be embedded within an app or Web page, whereas virtual assistants such as Siri and Cortana are stand-alone, ask-anything programs.

Definition 2 – Investopedia, “A chatbot is a computer program that simulates human conversation through voice commands or text chats or both”. Chatbot, short for chatterbot, is an Artificial Intelligence (AI) feature that can be embedded and used through any major messaging applications. There are a number of synonyms for a chatbot, including "talkbot," "bot," "IM bot," "interactive agent" or "artificial conversation entity."

The progressive advance of technology has seen an increase in businesses moving from traditional to digital platforms to transact with consumers. Convenience through technology is being carried out by businesses by implementing Artificial Intelligence (AI) techniques on their digital platforms. One AI technique that is growing in its application and use is Chatbots. Some examples of chatbot technology are virtual assistants like Amazon's Alexa and Google Assistant, and messaging apps, such as WeChat and Facebook messenger

Definition 3 – Wikipedia, “A chatbot is a piece of software that conducts a conversation via auditory or textual methods”. Chatbots are typically used in dialog systems for various practical purposes including customer service or information acquisition. Some Chatbots use sophisticated natural language processing systems, but many simpler ones scan for keywords within the input, then pull a reply with the most matching keywords, or the most similar wording pattern, from a database.

Definition 4 – Techopedia, "A chatbot is an artificial intelligence (AI) program that simulates interactive human conversation by using key pre-calculated user phrases and auditory or text-based signals". Chatbots are frequently used for basic customer service and marketing systems that frequent social networking hubs and instant messaging (IM) clients. They are also often included in operating systems as intelligent virtual assistants. A chatbot is also known as an artificial conversational entity (ACE), chat robot, talk bot, chatterbot or chatterbox.

From all the above definition we can summarize the following points

- Chatbot comes from the combination of two words those are Chat and robot. Chat means to talk in an informal or familiar manner, to take part in an online discussion in a chat room whereas A **robot** is a machine especially one programmable by a computer capable of carrying out a complex series of actions automatically.
- A conversation or message of Chatbot can be either in the form of audio or text format.
- Chatbot uses as a virtual assistant
- Chatbot Technology has increase business moving from traditional to digitalized one
- A chatbot is used in dialog systems for customer service or information acquisition

2.2. History of Chatbot Technology

In 1950, Alan Turing, a logician, cryptanalyst and pioneer of computer science, asks a question of what thinking means and whether machines can think in the article Computing Machinery and Intelligence[6]. At this Era, it's not easy to respond to the question that Machine can think instead of Alan replace the question 'Can Computer communicate in way insensible from a human.'

JOSEPH WEIZENBAUM'S ELIZA used simple pattern matching and a template-based response mechanism to emulate the conversational style of a non- directional psychotherapist which is called ELIZA[6]. Eliza was the first Natural Language Processing program that was created in the year of 1966. Since Eliza can only answer the question only defined before there is no Contextual event understanding.

PARRY was developed to simulate a paranoid patient[6][7][8]. Parry developed in the year of 1972 by Kenneth Colby. The program implemented a crude model of the behavior of a person with paranoid schizophrenia based on concepts, conceptualizations, and beliefs (judgments about conceptualizations: accept, reject, neutral).

A.L.I.C.E. (**Artificial Linguistic Internet Computer Entity**), is a natural language processing chatterbot—a program that engages in a conversation with a human by applying some heuristical pattern matching rules to the human's input. It's an NLP chatbot, which can engage in conversation with humans using some heuristical pattern matching rules. ALICE uses a simple stimulus-response technique to return the matching response to the user, similar to those used [9].

SmarterChild, SmarterChild a chatbot designed for use with various SMS platforms in early 2001[10]. SmarterChild was revolutionary in the fact that it was capable of providing the users chatting with news reports, weather updates, movie showings, and more. It was widely distributed across global instant messaging and SMS networks.

Siri, Siri was developed by Apple. Siri is a computer program that works as a virtual personal assistant and knowledge navigator. The feature uses a natural language user interface to answer questions, give suggestions, and perform actions by using a set of Web services[11]. It received praise for its voice recognition and contextual knowledge of user information, including calendar appointments, but was criticized for requiring stiff user commands and having a lack of flexibility

Google Assistant is a virtual assistant that can engage in two-way conversations. Initially, it appeared only on the Google Home smart speaker but is now available on Android devices as well as an iOS application. To create a bot on Google Assistant one has to use the 'Actions by Google' developer platform.

Amazon's Alexa, voice tech is now available to build chatbots[12][13]. Amazon is making its voice-control technology available to all, giving developers access to the same tools that power its digital assistant Alexa. Alexa was developed by Amazon Company in the year of 2014. Alexa is a virtual assistant AI developed by Amazon. It is capable of voice interaction, music playback, making to-do lists, setting alarms, streaming podcasts, playing audiobooks, and providing weather, traffic, sports, and other real-time information, such as news. Alexa can also control several smart devices using itself as a home automation system.

Cortana is a virtual assistant created by Microsoft for Windows[14]. Cortana can set reminders, recognize natural voices without the requirement for keyboard input, and answer questions using information and web results from the Bing search engine. Cortana is currently available in Multilingual.

Table 2: Summary of Chatbot history

No	Chatbot Development	Year
1.	*Alan Turing published the first paper about an intelligent machine that can chat with a human*	*1950*
2.	*Joseph Weizenbaum was developed the first Chatbot (ELIZA)*	*1966*
3.	*Kenneth Colby came out with PARRY, a chatbot that could simulate a person with paranoid schizophrenia*	*1972*
4.	*Artificial Linguistic Internet Computer Entity was developed by Richard Wallace*	*1995*
5.	*Peter Levitan was found Smarterchild*	*2001*
6.	*Siri was founded by Apple*	*2010*
7.	*Google Now was developed by google company for google voice search*	*2012*
8.	*Alexa which was developed by Amazon*	*2015*
9.	*Cortana Which was developed by Microsoft*	*2015*
10.	Cleverbot, Chatfuel, and Watson	*2017*
11.	Chatbot intelligent in voice and text form	*2017- currently*

2.3. Types of Chatbot

Chatbots come in two main flavors. The first is the standard rule-based bot that completes scripted actions based on keywords. The second is the AI-powered chatbot, which uses machine learning to converse more naturally.

2.3.1. Rule-based Chatbots

This is the primitive type of chatbot and is very specific regarding its versatility[15]. It answers questions and takes the necessary steps as per the codes punched into it. While working with them, one needs to keep in mind that the questions asked must be direct and simple. Only then the desired answer can be expected. Moreover, the answers are much general when compared to those from AI Chatbots. A bot development company can build rule-based Chatbots either with simple codes or complicated ones. It depends on the requirements stated by the company. But the bot won't do anything outside its code protocol. So, it might be difficult for a customer to find a solution for a new issue that has not been programmed. They are developed in Artificial Intelligence Markup Language. This type of bot has a high probability of failing the Turing test if it strictly abides by a particular model. There are certain advantages of rule-based Chatbots too like they incorporate direct-action buttons for many regular tasks such as sending an email, uploading an attachment, etc.

2.3.2. AI Chatbots

AI Chatbots in comparison with rule-based Chatbots provide a human-imitated reaction to the issues asked[16]. They give specific answers to each issue by learning natural language responses and regular updates in a machine learning engine. It gets better with more usage. An AI Chabot's basic advantage is the ability to be available all the time which accounts for 64% of its functionality. The next 55% of advantage accounts for instant responses. While the rest 55% is because of answering easy questions. Since the Artificial Intelligence companies are using machine learning to improve the functionality of Chatbots and are making them more intelligent.

2.4. Chatbot Modelling Approach or Priciples

2.4.1. Retrieval based model Approach

They are trained specifically on some questions and their possible answers. It is like a database of questions, and their answers are provided to the chatbot[16][17][18]. When a question is asked, the bot searches the most relevant answer from the lot and displays the one that is selected. It is different from other AI bots as it cannot generate a new answer to the question. But since they rely on machine learning, if the database has been arranged accurately, they can solve the queries commendably. It is up to the decision of developers and AI consulting services to set the rules of the query with ranging difficulty. The difficulty level is set using machine learning algorithms, and a suitable answer is selected. The advantage of this type of chatbot is that there is no issue regarding language and grammar as the answers are pre-set and don't clash with the syntax priority.

2.4.2. Generative model Approach

They are better than rule-based Chatbots since they can generate a new answer and the replies are different from the previous one[18]. The reason for their better performance is that they take words from the query and generate the answer accordingly. They have a shortcoming due to this property that they are more prone to errors than retrieval-based models as they require a spelling and grammar check before displaying the answer. For removing this issue, they need to be developed with better codes and guidance of expert machine learning companies in India. After full training, they can give the rule-based models a run for their money.

Proper planning and execution are necessary for the success of a chatbot. For a business to decide on which type of chatbot to be used for its processing, the Turing test results must be taken as a decisive result. The rule-based bots can be preferred for a business that deals with straight questions that don't require much intelligence and customized answers. Business dealing with delivery, tracking, and logistics offer the best application for rule-based Chatbots while AI Chatbots are ideal for customer issues, where pre-set answers cannot serve the purpose of satisfying the customer.

2.5. Knowledge-based Chatbot

Knowledge-based Chatbots are classified based on the knowledge they access from the underlying data sources or the amount of data they are trained on. The two main data sources are open-domain and closed-domain[19][20].

2.5.1. Open Domain Chatbot

Open-domain data sources answer depends on general topics and respond appropriately. Examples of open-domain are Allen AI Science and Quiz Bowl. In an open domain (harder) setting the user can take the conversation anywhere[19][20]There isn't necessarily have a well-defined goal or intention. Conversations on social media sites like Twitter and Reddit are typically open domain – they can go into all kinds of directions. The infinite number of topics and the fact that a certain amount of world knowledge is required to create reasonable responses makes this a hard problem.

2.5.2. Close Domain Chatbot

Closed-domain data sources focus on a particular knowledge domain. All information required for answering the question is provided in the dataset itself such as Daily Mail. In a closed domain (easier) setting the space of possible inputs and outputs is somewhat limited because the system is trying to achieve a very specific goal[19][20]. Technical Customer Support or Shopping Assistants are examples of closed domain problems. These systems don't need to be able to talk about politics, they just need to fulfill their specific task as efficiently as possible. Sure, users can still take the conversation anywhere they want, but the system isn't required to handle all these cases – and the users don't expect it to.

2.6. Chatbot Development Algorithm

The chatbot Algorithm can be divided into three types.

i. Pattern matches
ii. Machine Algorithms
iii. Artificial Neural network

2.6.1. Pattern Matchers

Pattern matching is the most effective and commonly used concept in Chatbots. Pattern matching is used to categorize a text input given by the user to generate an appropriate response for the user. AIML stands for "Artificial Intelligence Markup Language[21][22][23][24]". It can be considered a typical framework for pattern matching. An easy pattern matching example:

```
:aiml version = "1.0.1" encoding = "UTF-8"?>
 <category>
   <pattern> WHO IS ABRAHAM LINCOLN </pattern>
   <template>Abraham Lincoln was the US President during American civil war.</template>
 </category>

 <category>
   <pattern>DO YOU KNOW WHO * IS</pattern>
   <template>
     <srai>WHO IS <star/></srai>
   </template>
 </category>
:/aiml>
```

Figure 3: Pattern Matchers

2.6.2. Machine Algorithms

In chatbot technology for each kind of input given by the user, a distinct pattern is stored in the database[25][26]. These patterns help in providing users a suitable response. The use of algorithms is to reduce the classifiers and to create a large combination of patterns. These patterns generally follow a hierarchy that creates an effective and manageable structure. These algorithms follow the basic algorithm called Multinational Naive Bayes. For example, assume a set of words are given which belong to a specific class. With each new input of a word, each word is counted for its occurrence. And each class is assigned a score. The highest score in the class is the closest to be linked with the input word. Few sample Input sentence classification: Input - “Hello good morning” term - “hello” (no matches) Term - “good” (class: greeting) term: “morning” (class: greeting) classification: greeting (score=2) with the help of such an equation, word matches are noted for each class. Classification scores identify the class with the most term matches. But it also has certain limitations. The score explains that which intents to most likely match the sentence. There is no certainty that it is perfect.

2.6.3. Artificial Neural Networks

Artificial Neural Network is used to calculate the output from the input given by the user with the help of weighted connections. These weighted connections are generated from recurrent iterations during the processing or training of the data. At each step of training or processing, the data the weights are modified. This modification in weights results in an output with utter accuracy.

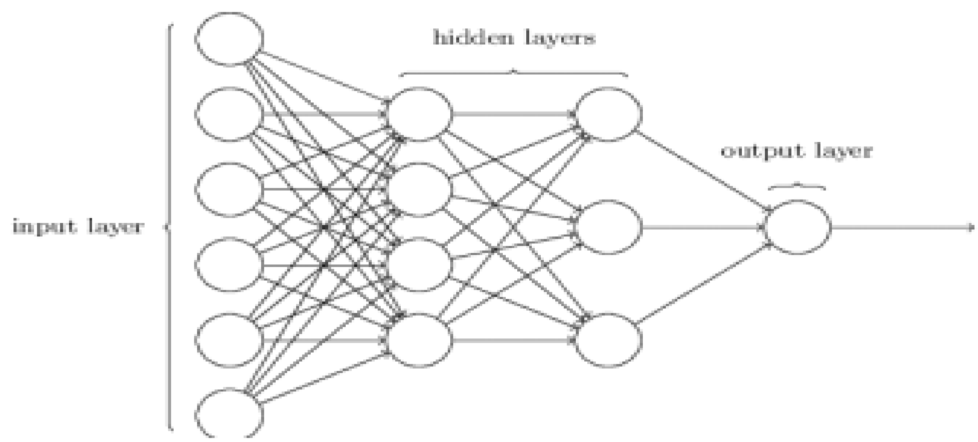


Figure 4: Artificial Neural Networork

In Artificial Neural Network, each sentence given as input is divided into separate words. Each word is then used as an input for the neural network[27]. There can be many different kinds of neural network and pattern matches but as the complexity increases the output of the chatbot becomes more accurate.

NLU (Natural Language Understanding): It consists of three concepts: Natural language understanding (NLU) is a branch of natural language processing (NLP), which helps computers understand and interpret human language[28].

• **Entities**: Entity can be considered as the main ideology of your chatbot. It can be a payment system chatbot or customer support chatbot or a resume bot as proposed in this paper.

- **Intents:** Intents are basically the outputs or the responses that a chatbot gives on encountering an input by the user. In short, these are the actions that a chatbot performs when a user gives a text input.
- **Context:** In the NLU algorithm whenever a sentence is processed or scanned, the system does not have any information about the past conversations of the user. It means if a user asks a user, and the chatbot has given a response to that question, but, the chatbot will not have a record of the question that has been just asked. So for distinguishing the parts of the conversation, the states of the conversation are recorded. A state can be a flag like "school studied" or parameter such as "educational qualification".

NLP (Natural Language Processing) In Natural Language Processing (NLP) chatbot uses a collection of steps that can be a collection of questions to transform the user's input text into relevant data and then give an appropriate answer or response. These are the steps involved in Natural Language Processing:

- **Sentiment Analysis**: This analysis keeps track of the conversations to check if the user is having a good quality experience.
- **Tokenization**: The NLP separates the sentence into different words or tokens. NLP classifies and divides a string of data into small parts or tokens. They are differently used for application purposes.
- **Normalization:** In this step, the chatbot looks for misspelled words or any kind of typing mistakes in the text input by the user.
- **Dependency Parsing**: In this step, the chatbot finds out the subject, verb, and object of the sentence given by the user. It looks for nouns and related or dependent phrases in the user's text input to understand what the user is trying to convey.

Similar to all the web applications present today, Chatbots also need to have a database. The database is the collection of information and facts that are used to generate a suitable and most appropriate response to the user. The data related to all the activities are stored in the database. So, NLP converts common human language to relevant information. The information can be a collection of patterns and relevant texts that can be applied to get a proper response.

2.7. Taxonomy of Chatbots Application

The recent interest in Chatbots can be attributed to two key developments. Firstly, messaging service growth has spread rapidly over the past few years. It incorporates features such as payments, ordering, and booking, which would require a separate application or website. Secondly, advanced AI techniques in combination with machine learning and deep learning techniques have made considerable progress to improve the quality of understanding and decision making on cheap processing power. It can handle the vast amount of data and process it to get results that exceed human performance. Chatbot applications can be grouped into four different categories, namely service, commercial, entertainment, and advisory chatbot. Service Chatbots are designed to provide facilities to customers. For example, logistics firm to respond to questions about deliveries and provide copies of dispatch documents through instant messaging channel rather than emails or phone calls. Commercial Chatbots are designed to streamline purchases for customers. For example, a pizza company can take delivery orders or notify promotions via messaging interface. Entertainment Chatbots are designed to keep customers engaged with sports, favorite bands, movies or other events. It offers the option of placing bets, detail on upcoming events and ticket deals. Advisory Chatbots are designed to provide suggestions, give recommendations on service, offer maintenance or repair goods. This type of chatbot can contact people, offer support and advice tips when it is needed.

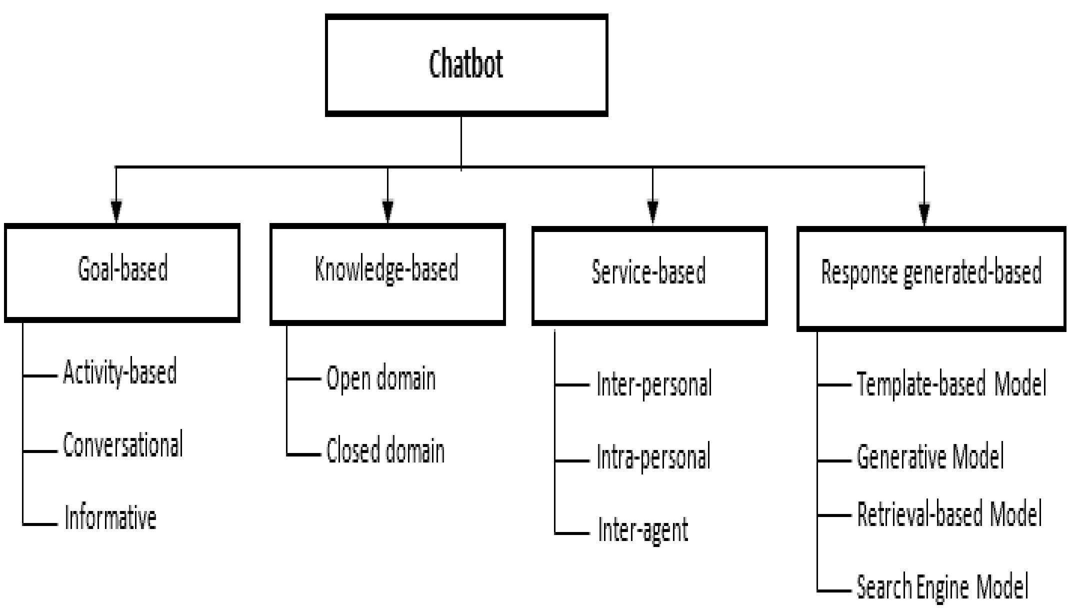


Figure 5: Taxonomy of Chatbot Application

2.8. Feature Extraction Of Chatbot

There are different feature extraction in developing Chatbot. From that Bag of Words and Word Embeding are the most known feature extraction in developing chatbot.

2.8.1. Bag of Words

Bag of Words (BOW): We make the list of unique words in the text corpus called vocabulary. Then we can represent each sentence or document as a vector with each word represented as 1 for present and 0 for absent from the vocabulary. Another representation can be count the number of times each word appears in a document. The most popular approach is using the **Term Frequency-Inverse Document Frequency (TF-IDF)** technique.

- **Term Frequency (TF)**: is a scoring of the frequency of the word in the current document. Since every document is different in length, it is possible that a term would appear much more times in long documents than shorter ones. The term frequency is often divided by the document length to normalize.

$$TF(t) = \frac{Number\ of\ times\ term\ t\ appears\ in\ a\ document}{Total\ number\ of\ terms\ in\ the\ document}$$

Equation 1: Term Frequency Equation

- **Inverse Document Frequency (IDF)**: is a scoring of how rare the word is across documents. IDF is a measure of how rare a term is. Rarer the term, more is the IDF score.

$$IDF(t) = log_e(\frac{Total\ number\ of\ documents}{Number\ of\ documents\ with\ term\ t\ in\ it})$$

Equation 2: Inverse Document Frequency equation

Thus,

$$TF - IDFscore = TF * IDF$$

2.8.2. Word Embedding

It is a representation of text where words that have the same meaning have a similar representation. In other words it represents words in a coordinate system where related words, based on a corpus of relationships, are placed closer together. Word Embedding is a type of word representation that allows words with similar meaning to be understood by machine learning algorithms. Technically speaking, it is a mapping of words into vectors of real numbers using the neural network, probabilistic model, or dimension reduction on word co-occurrence matrix. It is language modeling and feature learning technique. Word embedding is a way to perform mapping using a neural network.

WordVec: Word2vec represents words in vector space representation. Words are represented in the form of vectors and placement is done in such a way that similar meaning words appear together and dissimilar words are located far away. This is also termed as a semantic relationship. Neural networks do not understand text instead they understand only numbers. Word Embedding provides a way to convert text to a numeric vector.These are all words that showed up quite frequently in our conversation data file. While they are common in the realm of social media, they aren't in a lot of traditional datasets. Normally, my first instinct when approaching any NLP task is to simply use pre-trained vectors, as they are trained on large corpuses for a large number of iterations. However, given that we have so many words and acronyms that aren't in typical pre-trained word vector lists, generating our own word vectors Is critical to making sure that the words get represented properly.To generate word vectors, we use the classic approach of a Word2Vec model. The basic idea is that the model creates word vectors by looking at the context with which words appear in sentences. Words with similar contexts will be placed close together in the vector space

2.9. Maize productivity in Ethiopia

Food security in Ethiopia, and elsewhere in Africa, is a majorsocio-political issue. Its economic wellbeing is also dependent on the success of its agriculture[29]. Ethiopia has long suffered from food shortages and economic underdevelopment even though it is endowed with a wide range of crop and agroecological diversity[30]. Maize, teff (Eragrostis tef), sorghum,wheat, and barley among cereals and enset (Ensete ventricosum) (Bfalse banana^) among Broots and tubers^ provide the main calorie requirements in the Ethiopian diet. Crop productivity and production remained low and variable in the 90s for the most part but there have been clear signs of change over the past decade. Maize is ethiopia's most important cereal crop both in terms of level of production and area coverage. Within the country, maize is the largest cereal commodity in terms of total production and yield and second in terms of acreage next to tef. Maize has expanded rapidly and transformed production systems in Africa as a popular and widely cultivated food crop since its introduction to the continent around 1500 A.D.

2.10. Challenges of Chatbot

Challenges that companies face during chatbot development and ways to effectively tackle them. There are a lot of challenges which are associated withchatbots[31].

A. **Context in Chatbots:** The key to the evolution of any chatbot is it's integration with context and meaningful responses, as conversation without any context would be vague. It becomes challenging for companies to build, develop and maintain the memory of bots that offers personalized responses.That's when AI technologies like Machine Learning or NLP-Natural Language Processing come into the picture and overcome the challenge of understanding the depth of conversation; up-to an extent. NLP understands the databases and data sets when bots are structured, in predefined sequential order and then converts it into a language that users understand.However, humans don't interact in a defined order, as a result intelligent slot filling, which stores the preferences of the regular users is the alternative to maintain the memory of a bot effectively. This insures that your virtual agents are not interacting in the same old predefined order but in a more personalized fashion.

B. **Limited User Attention :** Users have limited time span for their queries and expect lightning-fast replies. It's quite challenging for firms to develop chatbots, that holds user's attention till the end. Conversational UI, here plays an important role in exhibiting human like conversations and better customer experiences. It initiates interactions to be more social than being technological in nature. The conversations as a result, should be natural, creative and emotional in order for your chatbot to be successful. In some cases, however a machine wouldn't always render the same empathy that a human could and this is when a human replacement should take care of the users request.

C. **Chatbot Testing:** Chatbot testing is another main issue where most of the complexity lies. Chatbots are continuously evolving due to its upgradation in natural language models. Thus, it becomes vital to test and run chatbot to check it's accuracy. Testing a chatbot will depend on what type of method you want to experiment.

- First method involves automated testing of chatbots. There are many automation testing platforms like Zypnos, TestyourBot, Bot Testing, Dimon etc. These platforms allow detailed reports of the results and coding of test scripts, which could be run for all the test cases.

- The other method involves testing of conversational logic i.e. manual testing executed by a closed group of testers. They act as users and check the bot for all the unexpected slots possible. This method can be time consuming and partially accurate. However, it has its own benefits that outrage automation, by checking the logic against human conversations

D. **Viability of Data:** There is no point of having lots of data, intelligent slot filling or technologically advanced chatbot. It is vital for a chatbot to not only be enriched with meaningful data, but also to be equipped to deliver the brand identity to your target audience. In order to check the viability of your virtual agents you should consider asking yourself the following:

  - Are your virtual agents delivering to the right audience?
  - Does it offer business goals uniquely?
  - How is it different from other chatbots?

It may definitely seem to be a great idea to implement chatbot in your digital strategy, but creating a one that meets the expectations of your organisation and users, is a big challenge. An effective and well planned strategy is important for you to consider before presenting the chatbot to your audience. If done well, chatbots can become the contact point for your business and can increase the overall productivity by meeting the customer's on-demand expectations.

2.11. Afaan Oromo Language

2.11.1.Background

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. Oromoo people are also inhabited in neighboring countries like Kenya and Somalia. Afaan Oromoo 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[32]. It is used by Oromo people, who are the largest ethnic group in Ethiopia, which amounts to 34.5% of the total population[33]. 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[32]. 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. Now a day, it is an official language of the Oromia state (which is the largest Regional State among the current Federal States in Ethiopia).

2.11.2. Writing system of Afaan Oromo language

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[34][35], 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".

2.12. Related Work

To meet the objective of the research, a systematic literature review (related works) is a very important concept to be issued. Many were research is conducted with related agricultural crop production related to technology and other human labor life. From the research that closely related to the study:

Agricultural productivity can be increased by using two ways. The first method is through improvement in technology given some level of input and the other option of improving productivity is to enhance the output per household labor ratio of rural household farmers, given fixed level of inputs and technology. This study was mainly concerned about the second option of increasing productivity i.e. output per labor input and output per cultivated area of land[1]. Even if the two factors are very important to increase crop productivity the researcher concerned with only improving productivity is to enhance the output per household labor ratio of rural household farmers. On the other hand, technology is also the main to improve crop productivity. Production levels of the main agricultural crops, distribution of cultivated area by farm size and the importance of seasonal differentiation[36]. According to the researcher to improve the agriculture crop production farm size and seasonal differentiation is the main factor. Even the two factors researcher explained are the tackle of improving productivity proving these problem is not enough to improve productivity. Technology enhancement is very important in agriculture to improve crop production. To develop a chat-bot that will provide assistance and guidance to farmers using NLP[37]. This research was done by Indian professionals it considers the KNN algorithm to improve crop productivity. They use a conversational, relatable style to interact with customers, allowing agriculture companies to improve customer service, productivity, and operational efficiency[38]. This article explained the benefit of chatbot on agricultural crop productivity. The researcher does not explain how to apply a machine-learning algorithm to build a chatbot for improving agricultural crop productivity. AI can change the agriculture landscape, the application of drone-based image processing techniques, precision farming landscape, the future of agriculture and the challenges ahead[39]. This journal within the researcher tries to address how to improve agriculture productivity within the help of artificial Intelligence. Developed a knowledge base for potato farming using the KCC dataset[40]. The researchers basically depend on the knowledge base for information gain to improve the productivity of potatoes. FarmChat which combined conversational and language technologies to naturally converse with farmers in answering their farming-related queries[41]

Table 3: Summary of Realted Work

Title	***Author***	***Objective***	***Gap mapping***
The determinants of agricultural productivity and rural household income in Ethiopia	*T. Urgessa*	*Mainly discus how productivity can be increased output per labor input*	*Recommends technology to be more effective get product*
Crop production in Ethiopia: Regional patterns and trends	*A.Seyoum Taffesse, P. Dorosh, and S. A.Gemessa*	*Distribution of cultivated area by farm size and the importance of seasonal differentiation*	*Again their recommendation is also technology can enhance productivity*
Smart Chatbot for Agriculture	*P. Y. D. K, R. Hemalatha, and G. Niveditha*	*To develop a chat-bot which will provide assistance based on KNN algorithm*	*They other machine learning algorithm to get more accurate response*
Using Chatbots to Answer Farmer Queries in India	*W. Vota*	*Depend on knowledge base to information gain to improve productivity of potatoes*	*With knowledge the target beneficiary can't access as the want*

CHAPTER 3

3. Research Methodology

In this chapter I have give an overview of the models used to create a chatbot and the chatbot implementation. A user can sent a message to the chatbot to which the chatbot formulates a reply using the question-answer dataset. The user and chatbot can sent multiple messages and replies, creating a chat conversation. In this studies I have used machine learning models, For both designing and created a chatbot implementation, In this study Afaan Oromoo maize productivity enhancement question-answer datasets are used as the basis for the chatbot.

3.1. Data Collection

The data used in this study comes from ATA (Agriculture Tranformation Agency), Which is used as Call Center for Agriculture productivity and Oromia agriculture bureau.Data from Oromia agriculture bureau is training package which create awareness for farmer how to enhance their maize productivity. The flow of data from ATA Farmers can call to the the Call center ask agriculture related questions to call center and the call center response by voices depend on the users. The framework in which the questions are asked is a chat setting. Therefore the conversation between user and expert is not limited to one question and one answer, but can also contain an entire chat session. The data is stored in three JSON1 files. The first one contains the first question a user asked, the second one contains all messages in the rest of the chat session and the third one contains information about who typed which message.

3.1.1. Building Dataset

The Objectives of this sudy are developing Afaan Oromoo Chatbot for enhancing maize productivity. So, it needs to build a new Afaan Oromoo Conversation for enhancing maize productivity. The new dataset are needed because of there is no annoatade for this purpose. To develop the new dataset for developing Afaan Oromoo chatbot for enhancing maize productivity the following steps are needed.

1. Selecting the related document from ATA and Oromia Agricultural bureau
2. Data preprocessing
3. Annotation dataset

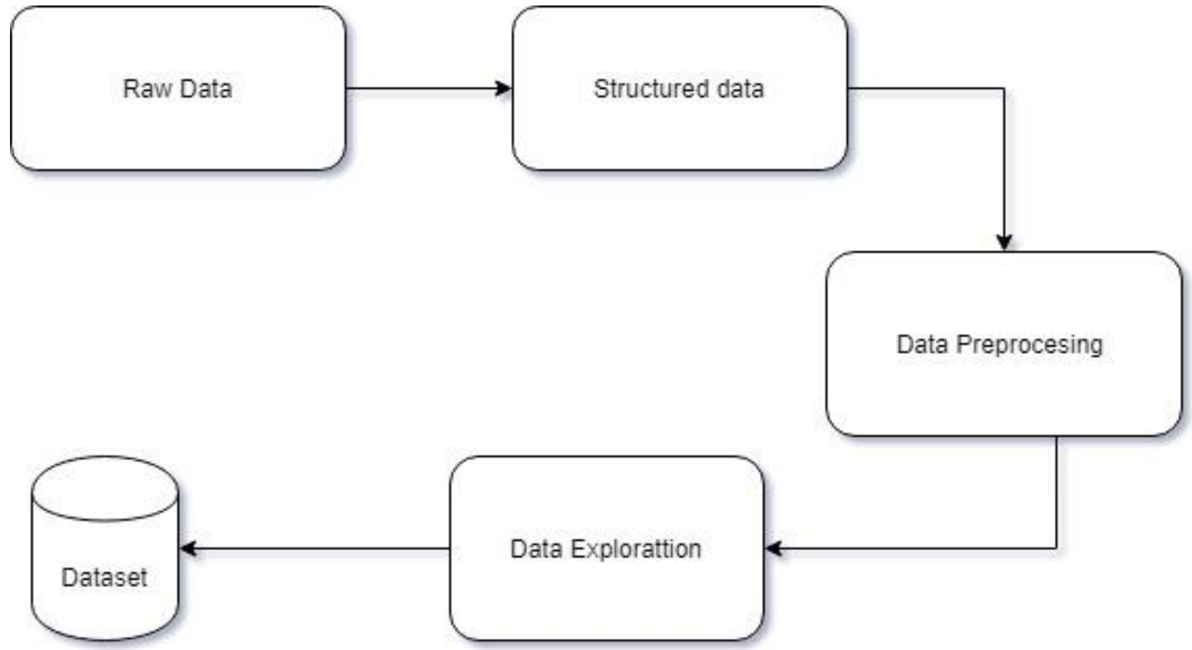


Figure 6: flow of dataset building

In the figure 7: , the raw data in the case of this study, the data from ATA and Oromia agricultural bureau. Data is not structured,so it needs to structure. That means the second step to build dataset in this study. Structured the data in this study means preparing the in the form of conversation. That means in the form of question and response. The single question have different response. Multiple response for single question the concepts of this study.

3.1.2. Data Preprocessing

After selected the raw data , data preprocessing is the next important point in building dataset. Data preprocessing needs the following steps to be processed.

1. Formatting the data to make it suitable for ML (structured format)

Example of labeling data in compatable with ML

```
   {"tag": "Omishaaf",
         "patterns": ["Omishaa boqqoloo dabaluuf maal gochuun qaba?",
"Akkamitti omisha boqqoloo dabaluu dandaha?", "Galatommaa","guyyaa gaarii"],
         "responses": ["gosa lafa qonnaa kessani beeku", "Naannoo qillensa
kessaa jiraatani beeku", "homa rakkoo hin qabu", "nagaan ollaa"],
         "context_set": ""
        },
```

2. Cleaning the data to remove incomplete variables

Example for cleaning unrelevant data.

Ilbisoota sadarkaa biqilaatti boqqoolloo huban keessaa raammoon murtuun,’agadaa qorqor’ , geerriin adda dureedha.

3. Sampling the data further to reduce running times for algorithms and memory requirements.

Afaan oromoo text preprocessing method is used to clean up the question and response based on Afaan Oromoo language and this used bbasic text mining techniques. Data preprocessing techniques used for making the dataset ready for feature extraction method.

Data Cleaning : Data cleaning is a process used to determine inaccurate, incomplete or unreasonable data and then improve the quality through correcting of detected errors and omissions. Generally data cleaning reduces errors and improves the data quality. Correcting errors in data and eliminating bad records can be a time consuming and tedious process but it cannot be ignored.

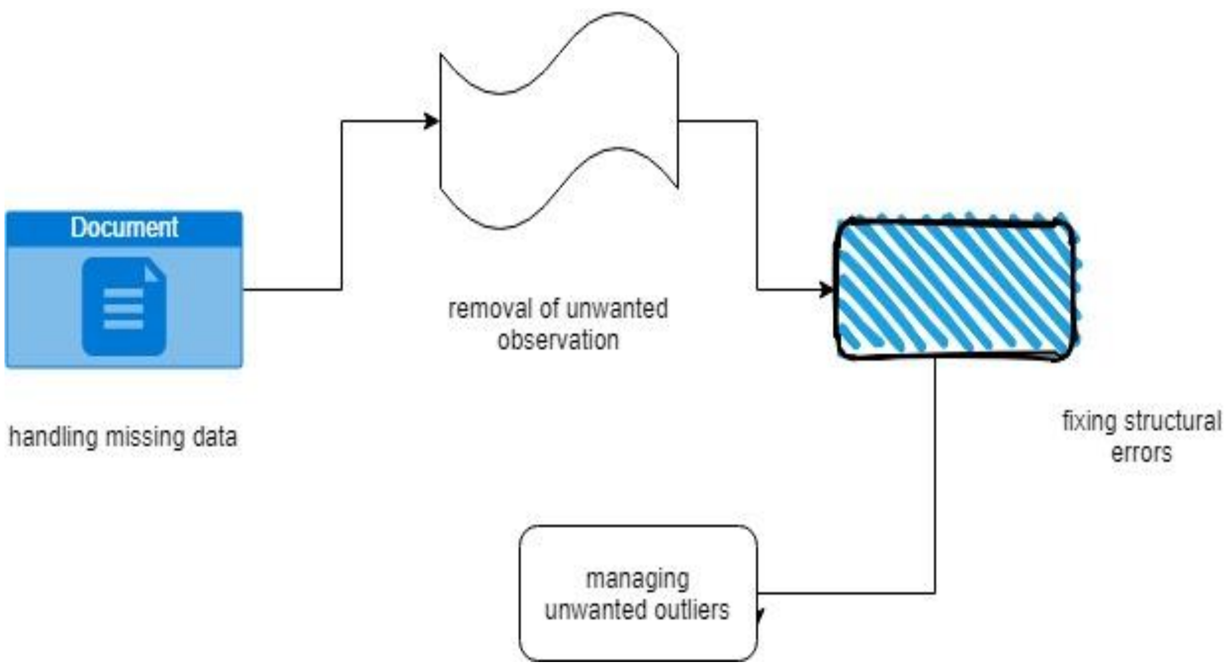


Figure 7: data preprocce flow

3.2. Chatbot Features Extraction

These methods are based on the state of art text mining feature extraction, techniquues applied for reducing redundant features and dimensionality. This techniques are used for extracting the best response for the question from the user. There are different feature extraction methods that used for modelling Afaan Oromoo Chatbot.

Term Frequency-Inverse Document Frequency (TF-IDF): TF-IDF is a statistic that reflects how important

a certain word is to a text in a collection of texts[42]. The TF-IDF statistic increases when a word appears more often in a text, but this is offset by the amount of times the word appears in all texts. If *t* is the term frequency in the input, *d* is the number of text documents the term appears in and *D* is the total

$$\text{TF-IDF} = t \log D/d$$

number of text documents, then With this definition, a term that appears in every document and is thus oflittle value for separating the documents, will have an almost zero IDF (log *Dd*) value, leading to a small TF-IDF value. However, a term that only appears in one document will have a very high IDF and thus also a higher TF-IDF. TF-IDF algorithm had adjusted IDF-weights of each word, so it could be used to maximize the probability of retrieving a correct answer for every question. The IDF-weights were adjusted by applying gradient descent to bring questions and answers closer together.

Word Embedding: Word Embedding is a type of word representation that allows words with similar meaning to be understood by machine learning algorithms[43][42]. Technically speaking, it is a mapping of words into vectors of real numbers using the neural network, probabilistic model, or dimension reduction on word co-occurrence matrix. It is language modeling and feature learning technique. Word embedding is a way to perform mapping using a neural network.There are various word embedding models available such as word2vec (Google), Glove (Stanford) and fastest (Facebook).In this study Wes have used only Word2vec models among varius model.

Word2vec: Word2vec is the technique/model to produce word embedding for better word representation. It captures a large number of precise syntactic and semantic word relationship. It is a shallow two-layered neural network. Before going further, please see the difference between shallow and deep neural network: The shallow neural network consists of the only a hidden layer between input and output whereas deep neural network contains multiple hidden layers between input and output. Input is subjected to nodes whereas the hidden layer, as well as the output layer, contains neurons. The word2vec tool takes a text corpus as input and produces the word vectors as output. It first constructs a vocabulary from the training text data and then learns vector representation of words. The resulting word vector file can be used as features in many natural language processing and machine learning applications. Word embedding is the numerical representation of a text. Since all Machine Learning and Deep Learning algorithms are incapable of reading the plaintext, we require word embedding. Mapping of a word to a vector is done using dictionary and this vector is represented in one-hot encoded form which means.This technique can be applied to chatbots. If you are making a chatbot that needs to answer a question, it needs to understand the meaning of the question. We can extract keywords from the question and calculate the average vector of all keywords to determine the meaning (topic) as described above.

3.3. Deep Learning Algorithm

The objective of this study was developing Afaan Oromoo text based Chatbot for enchancing maize productivity in Oromia.Deep learning is a method of data analysis that automates analytical model building. A machine learning (ML) engine, based on neural networks, looks at a pattern (say, a text message) and maps it to a concept such as the semantics, or the intent of the customer sending the message[44]. It's a branch of artificial intelligence based on the idea that systems can learn from data, identify patterns and make decisions with minimal human intervention. Although there is a vast range of different approaches and techniques encompassed within the notion of machine learning, underlying most of them is statistics. This enables algorithms to make predictions based on extracting patterns from incomplete and often noisy input data. Within machine learning, artificial neural networks have gained a prominent position and were initially inspired by the networks of connected neurons found in human and animal brains. Essentially, when it comes to training an artificial neural net, the best way to do it is to have the system make a guess, receive feedback, and guess again, continually shifting the probabilities that will get it to the right answer. As brain-inspired systems designed to replicate the way that humans learn, neural networks modify their code to find the link between input and output in situations where this relationship is complex or unclear. This decade, artificial neural networks have benefited from the arrival of deep learning, which increases the 'depth' of different layers in the network to extract different features until the network can recognize what it is looking for. The objectives this study is to develop Afaan Oromoo Chatbot for enhancing maize productivity in Oromia region. Among the machine learning algorithm We select Deep Neural Network. There are different Deeps neural network among we select

Deeps neural network for Chatbot :DNN-based methods to handle request-response matching in natural language processing[45][46][32]. DNN have diffrerent models and algorithm to handle request-response matching in natural language processing. Deep neural network find the intents and match the question with the correspondence responses and identify the class of the question..Intent is important concept in developing the chatbot technology. An **intent** is the intention of the user interacting with a chatbot or the intention behind each message that the chatbot receives from a particular user. Intents may vary from Chatbot to another deopende on the domain or knowledge area. The reason why we need to identify intents when we devevlop Chatbot technology was:

- In order to answer questions
- Search from domain knowledge base
- Perform various other tasks to continue conversations with the user

Recurrent neural network (RNN) for Chatbot: This method can take as input a variable length sequence x = (x1, ..., xn) and produce a sequence of hidden states h = (h1, ...,hn), by using recurrence[47][48]. A Recurrent Neural Network (RNN) is designed to preserve the previous neuron state. This allows the neural network to train the question and response dataset and produce the desired output based on the question from the user. In chatbot systems recurrent neural networks (RNNs) are used because these networks can learn vector representations from words. In the case of this study Seq2Seq network which is good model building NLP in RNN is proposed. To use this network both question and answer are split by words, and each word is embedded in a vector. These vectors are fed into the same RNN word-by-word.

3.4. Evaluation

Evaluation of this study has been conducted on phases of the development of developing Afaan Oromoo Text based chatbot model for enhancing maize productivity.

3.4.2. Model Performance Evaluation Metrics

Model evaluation is required to quantify the model performance of developed Afaan Oromoo text-based Chatbot model. In this study we used Accuracy for evaluate the model .

Accuracy: Accuracy of a chatbot can be defined as the percentage of utterances that had the correct intent returned. It calculate the percentage of the response that has been correctly match the intents which is defined our dataset trained by our models.

CHAPTER FOUR

4. Proposed Afaan Oromoo Text-based Chatbot

This Chapter discusses the proposed solution of developing Afaan Oromoo Text-based Chatbot for enhancing miaze productivity using Deep learnig techiniques which is suitable solve problem under the study.This study creates conversational dataset which consists question and response. The dataset collect from Oromia Agricultural buraue and federal agricultural transformation agency(ATA).

4.1. The Proposed Chatbot Architecture

The proposed architecture used develop Afaan Oromoo Text-based Chatbot for enhancing maize productivity , depend on the input from the user (question) the bot was select the best response for the user. The proposed solution was based on the architecture shown in figure 8. It takes Afaan Oromoo dataset as input and then the preprocessed takes places based on nature of the language which removing punctuations, tokenize and another basic necessary preorocess. After the preprocess was takes places then Feature extraction was takes places to extract feature using Word embedding.

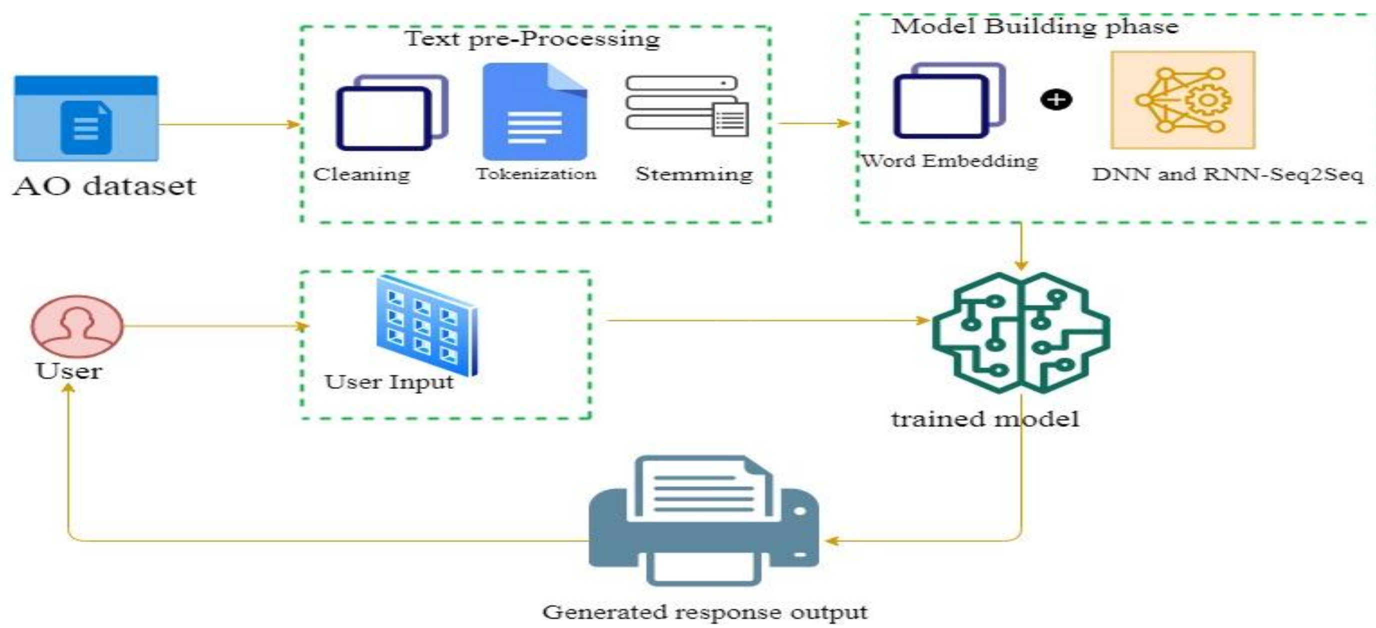


Figure 8: Proposed Afaan Oromoo Text-based Chatbot framework

4.2. Proposed Afaan Oromoo Text Preprocessing

This is subcomponent performs Preprocessing of Afaan Oromoo dialog dataset for training and testing evaluation model. This Preprocessing is performed based on the Afaan Oromoo language and basic text preprocess techniques such as removing(cleaning) puntuations and special characters , normalization and tokenization.

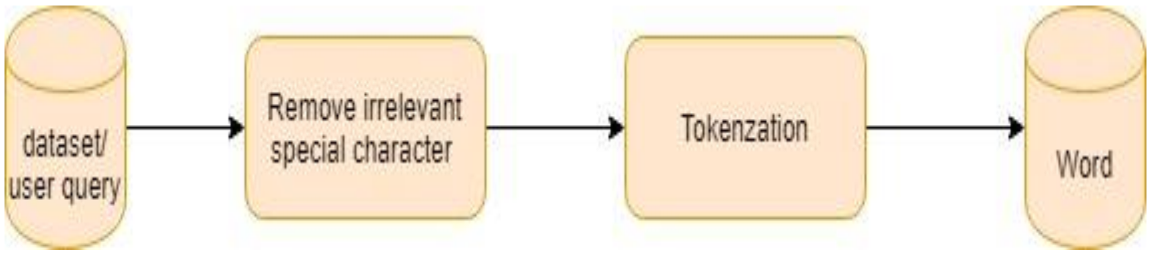


Figure 9: Text Preprocessing architecture

4.2.1. Removing(Cleaning) Irrelevant Character and Special Character

The dataset was created using the data from Ethiopian Agricultural Agency and Oromia Agriculture Bureau which contains a special character and puntuattion. This removing (cleaning) tasks removes all irrelevant character as wel as special character that the text contains.

Pseudocode: Removing Irrelevant Characters

```
Input: text in a dataset
Output: clean text
Begin:
1. Read the text in the dataset;
2. While (! end of the text in a dataset):
        If the text contains special_char [,’! @#$%^&*] then
        Remove special_char
        If the text contains symbol [<>‹›« »=:-`~_/] then
        Replace symbol and add space
        If a text contains Afaan_oromoo_Punc='[, • ] then
        Remove Afaan_oromoo_Punc
        If text contain English_word_num = [a-z A-Z] [0-9] then
        Remove English_word_num
        If text contain number = [0-9] then
        Remove number
        If a text contains extra white space then
        Trim the text
3. Return clean_text;
End:
```

4.2.2. Tokenization

After removing of irrelevant and Special character tokenization method was follow , which splits the question and response individual words or tokens by using the spaces between words or punctuation marks. This is important because of the meaning of text ganarally depends on the relations of the words in that text and help the feature extraction methods to get appropriate feature form the dataset.

Pseudocode: tokkenization

```
Initialize feature vector bgfeature = [0,0,...,0]
for token in text.tokenize() do
   if token in dict then
      tokenidx = getIndex(dict, token)
      bgfeature[tokenidx]++
   else
    continue
   end if
end for
return bgfeature
```

4.3. Proposed Feature Extractions

The proposed feature extractions components of the detection system performs the extractions of important feature of the dataset. These methods take input of preprossed and tokenized dataset words and performs. Then extracted features are used for training the models and predict the class of the intents and give the best response for the question based the intents class.

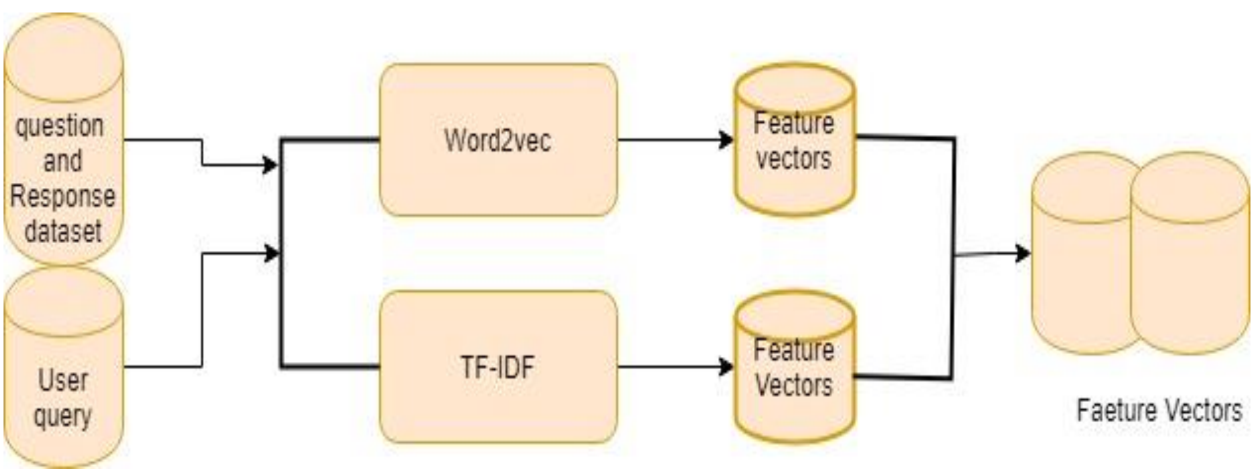


Figure 10: Proposed feature extraction

Word2vec and TF-IDF feature extraction methods are well known in the text minning approach. Each methods gives feature vector for train machine learning.

4.3.2. TF-IDF Feature Extraction

The proposed TF-IDF feature extraction is used to get the word frequency in the question and response dataset by applying the TF and measuring the importance of the word in dataset by applying IDF. The result of this feature extraction is to get the frequency and the importance of the word in the dataset.

4.3.3. Word2vec feature extraction

Under this study , the proposed word2vec method performs word to vector modeling of data. This are the question and response collected from different research center. This data are question and response are used for building the model. These model is used to extract the feature from the text in the dataset by calculating the average of all vectors using the model.

4.4. Deep Learning Model for Developing Afaan Oromoo text-based Chatbot

4.4.2. Deep neural network for developing Afaan Oromoo text-based Chatbot

As we discussed in chapter three and four Deep neural network is one of the proposed algorithm among the two proposed solution for this study. Deep neural network has different models to built Chatbot technology. This model contains different four layers including Softmax as activation fuction. DNNs are made and trained to give accurate results for the specific purpose they were made for. Basically, information is first fed to the input neurons. This information will then flow through multiple layers of neurons called hidden layers. Neurons in these hidden layers will send a high or low value to all the neurons in the next layer depending on the degree of importance (weight) of the link between the first mentioned neurons and the ones in the previous layer. Finally comes out of the output neurons, values, from which we will pick the highest to represent the result given by our neural network.

Pseudocode: DNN algorithm for training model

```
Input : dataset
  output : training result
  Inputlayer load dataset for training
  FOR each hiddenLayers
    FOR each hiddenLayer's neurons
      Set weightedSum to 0
      FOR each neuron's links
        Multiply link's weight with associated previousLayer's neuron's value
        Add result to weightedSum
      END FOR
      Call activation function
      Set neuron's value to result
    END FOR
  END FOR
  // OUTPUT LAYER
  FOR each outputLayer's neurons
    Set weightedSum to 0
    FOR each neuron's links
      Multiply link's weight with associated previousLayer's neuron's value
      Add result to weightedSum
    END FOR
    Call activation fuction
    Set neuron's value to result
  END FOR
END PROCEDURE
```

4.4.3. Recurrent Neural Network developing Afaan Oromoo Text-based chatbot

Recurrent Neural Network (RNN) is composed of input layer, multiple hidden layers and output layer[49]. In input layer, input is feed as vector representation. Then, input vector is multiplied by some weight and some biases are added. Then, the output from input layer is passed to next hidden layer where each consecutive hidden layer is composed of numerous RNN cells. After getting output from input layer, the cells in hidden layer multiplies the generated output from input layer by their own cell weights and biases. Next, in each of the hidden layer cells, some global activation function is applied to generate output from hidden layer. Then, output from each hidden layer cells is passed to successive hidden layer. Similar to previous hidden layer cells, some weight, biases and activation function is applied to the input of current hidden layer cell. This procedure propagates though all consequent hidden layers. Finally, output generated from the final hidden layer is passed to output layer and the output layer applies some function (e.g. Softmax) to generate final output. For RNN, the output vector from Final output layer is then again fed into the input layer as an input vector. Hence, the sequence information is stored in the memory and utilized. As we discussed in chapter three and four recurrent neural network is one of the proposed algorithm among the two proposed solution for this study. Recurent neural network has different models to built Chatbot technology. From different models of Recurnet neural network We select seq2Seq to model which contains different neural network layers. Encoder and decoder layers are the components of Seq2Seq model for our proposed study. Sequence To Sequence model become the Go-To model for Dialogue Systems. It consists of two RNNs (Recurrent Neural Network) , an Encoder and a Decoder. The encoder takes a sequence(sentence) as input and processes one symbol(word) at each time step. Its objective is to convert a sequence of symbols into a fixed size feature vector that encodes only the important information in the sequence while losing the unnecessary information. You can visualize data flow in the encoder along the time axis, as the flow of local information from one end of the sequence to another.

4.4.4. Adam Optimizer

The proposed solution uses Adam optimization algorithm. Why because Adam prdict the first (mean) and second (variance) to regulate the coefficients corrections which is suitable optimization algorithm for the sequential types of data format (e.g.DNN, RNN) and suitable optimizer used in Tensorflow and Keras framework. Adam optimizer is very tolerant of initialing the learning ratewhich is known as alpha and more training parameters as.

4.4.5. Activation Function

We use activation functions, to formulate the result of a neuron forward. The result gained by the neurons of the subsequent layer to the neuron is associated with the fit for the final layer builtin. The activation function used in the proposed model is the softmax activation function. The Softmax function is types of activation function used in neural computing. It is used to compute probability distribution from a vector of real numbers. The Softmax function produces an output which is a range of values between 0 and 1, with the sum of the probabilities been equal to 1. The Softmax function is used in multi-class models where it returns probabilities of each class, with the target class having the highest probability.

CHAPTER FIVE

5. Implementation and Expertimentation

This chapter presents the implementation of the proposed solution for Developing Afan Oromo Text-based Chatbot for enhancing Maize productivity by using the methodology discussed in chapter three and implementing proposed solution in chapter four. This study follows based on the experimental approach by implementing deep neural algorthim and natural language processing feature extraction.

5.1. Implementattion Environmet

Under this section several development tools and packages are used to implement the proposed solution for the developing of Afan Oromo Text-based Chatbot for enhancing maize productivity. This study uses python programming language for implementing and experiment each proposed solution starting from data preprocessing to model development by using deep neural network algorithm. **Table 4.** shows the list of tools and python packages with their vesion and description which is used in this study.

Table 4: Implementation Environment

Tools	*Description*
Anaconda navigator	*Allows us to launch development applications and easily manage conda packages, environments, and channels without the need to use command-line commands*
Jupyter notebooks	*An open-source web application that allows us to create and share documents that contain live code, equations, visualizations, and narrative text. Uses include data cleaning and transformation, numerical simulation, statistical modeling, data visualization, and machine learning.*
Spyder	*Spyder, the Scientific Python Development Environment, is a free integrated development environment (IDE) that is included with Anaconda. It includes editing, interactive testing, debugging, and introspection features.*
Python	*Easy to learn, powerful programming language to develop a machine learning application.*
Scikit-learn	*A set of python modules for machine learning and data mining. This study uses it for feature extraction and training and testing model. The name of the package is called sklearn.*

Tensorflow	*TensorFlow is a Python library for fast numerical computing created and released by Google.It is a foundation library that can be used to create Deep Learning models directly or by using wrapper libraries that simplify the process built on top of TensorFlow.*
Keras	***Keras*** *is an open-source library that provides a Python interface for artificial neural networks. Keras acts as an interface for the TensorFlow library.*
pandas	*High-performance, easy-to-use data structures, and data analysis tools. This study uses it for data reading, manipulation, writing and handling the data frame.*
NumPy	*Array processing for number, strings, and objects. This study uses it for handling converting the text to numeric data for features and training and testing the model.*
Gensim	*Python library for topic modeling document indexing and similarity retrieval with large corpora. This study uses it for the constricting word2vec model.*
nltk	*Build python programs to work with human language data. This study uses it for tokenization.*
matplotlib	*Publication quality figures in python. This study uses it for data and results visualization.*

5.2. Deployment Environmet

The tools which are discussed above in section 5.1 has been deployed on a personal computer computer equipped with, a processor Intel(R) Core(TM) i5-4210U CPU @ 1.70GHz 2.40GHz, Gigabyte of physical memory , 464 Gigabytes hard disk storage capacities. The operating system is window 8.1 pro 64 bits.

5.3. Data Set Description

In order to achieve the objectives , we need to build the dataset that are collected from different sources. To build the dataset three organization are selected Oromia Agricultural and Natural Resource bureu , Bako Agriculture research center and Ethiopain Agriculture Transformation Agency.

5.4. Preprocessing Implementation

To implement Afan Oromo preprocessing method the study used python programming language and modules. In order to perform the whole implementation of the methods, we perform the following common activities , which are importing the important libraries packages and load the dataset file disk using the code in figure 5.1. Afan Oromo preprocessing method uses nltk modules.

```python
import nltk
from nltk.stem.lancaster import LancasterStemmer
stemmer = LancasterStemmer()

# things we need for Tensorflow
import numpy as np
import tflearn
import tensorflow as tf
import random

# import our chat-bot intents file
import json
with open('C:/Users/Sanyichoo/chat.json') as json_data:
    intents = json.load(json_data)
```

Figure 11: Preprocessing Implementation

5.4.2. Implementation of Cleaning Irrelevant character

In order to clean the question and response of the dataset, we implement the method that used clean and replace the puantauatin mark , special character , symbol and emojis. The code for cleaning the question and response of the dataset.

```python
#rtemove unnnecessary,, split into words, no hyphens
#list of words
def sentence_to_wordlist(raw):
    clean = re.sub("[^a-zA-Z]"," ", raw)
    words = clean.split()
    return words
```

5.4.3. Implementation of Dataset tokenization

To tokenize the word of the text in the dataset we used a python module nltk. The python split function and nltk method are used for word tokenize. The result of this funtions is words, which is input for feature extraction methods .

```python
words = []
classes = []
documents = []
ignore_words = ['?']
# loop through each sentence in our intents patterns
for intent in chat['intents']:
    for pattern in intent['patterns']:
        # tokenize each word in the sentence
        w = nltk.word_tokenize(pattern)
        # add to our words list
        words.extend(w)
        # add to documents in our corpus
        documents.append((w, intent['tag']))
        # add to our classes list
        if intent['tag'] not in classes:
            classes.append(intent['tag'])

# stem and lower each word and remove duplicates
words = [stemmer.stem(w.lower()) for w in words if w not in ignore_words]
words = sorted(list(set(words)))
```

Figure 12: Implementation of Dataset tokenization sample code

5.5. Feature Extraction Imolementation

To train model we need to extract feature s from the dataset. We used python Scikit-learn module for TD-IDF and python Gensim for Word2vec. To implement the feature extraction the the dataset must be clean and tokenize. Each method of feature extraction are implemented with vector transformation methods, because the machine learning models operates in vectors instead of words.

5.5.2. Implementation of TF-IDF

To implement the TF-IDF feature extraction the study used the method TfidfVectorizer class of sci-kit learns packages. This method converts a question and response of the dataset to a matrix of TF-IDF feature vectors. This matrix of TF-IDF contains the word frequency and their importance in the dataset.

5.5.3. Implemetation of Word2Vec

To implement **word2Vec,** this study uses a python Gensim module which is used to implement different embedding methods. Feature modeling with gensim word2Vec is straightforward. First import and instantiate word2Vec class with necessary parameter and builds the vocabulary, and training the Word2Vec model using the question and response dataset.

```
from __future__ import absolute_import, division, print_function
import gensim
from gensim.models import Word2Vec
```

```
import codecs
import glob
glob.glob('/pycharm/*')
import logging
import multiprocessing
import os
import pprint
import re
```

```
import nltk
import gensim.models.word2vec as w2v
import sklearn.manifold
import numpy as np
import matplotlib.pyplot as plt
import pandas as pd
import seaborn as sn
```

The training resulted in a feature vector is also known as the embeddings. Embeddings are features that describe the target word. Then the result word2vce model used to extract features by computing similarity for a word in the dataset and used it as a feature to train machine learning models.

5.6. Deep Neural Network Implementation

As we discussed in chapter three and four Deep neural network is one of the proposed algorithm among the two proposed solution for this study. Deep neural network has different models to built Chatbot technology. This model contains different four layers including Softmax as activation fuction . Sample Implementation code for this model

```
# create train and test lists
train_x = list(training[:,0])
train_y = list(training[:,1])

# reset underlying graph data
tf.reset_default_graph()
# Build neural network
net = tflearn.input_data(shape=[None, len(train_x[0])])
net = tflearn.fully_connected(net, 8)
net = tflearn.fully_connected(net, 8)
net = tflearn.fully_connected(net, len(train_y[0]), activation='softmax')
net = tflearn.regression(net)
```

Figure 13: Deep neural Netwoork implementation sample code

5.7. Recurent Neural Network Implementation for Afaan Oromoo Text-based Chatbot(RNN)

As we discussed in chapter three and four recurrent neural network is one of the proposed algorithm among the two proposed solution for this study. Recurent neural network has different models to built Chatbot technology. From different models of Recurnet neural network We select seq2Seq to model which contains different neural network layers. Encoder and decoder layers are the components of Seq2Seq model for our proposed study. To implement this model we need Softmax activation fuction. Sample Implementation code for this model

```
In [15]: encoder_inputs = tf.keras.layers.Input(shape=( None , ))
         encoder_embedding = tf.keras.layers.Embedding( num_input_tokens, 256 , mask_zero=True ) (encoder_inputs)
         encoder_outputs , state_h , state_c = tf.keras.layers.LSTM( 256 , return_state=True , recurrent_dropout=0.2 , dropout=0.2 )(
         encoder_states = [ state_h , state_c ]

         decoder_inputs = tf.keras.layers.Input(shape=( None ,  ))
         decoder_embedding = tf.keras.layers.Embedding( num_output_tokens, 256 , mask_zero=True) (decoder_inputs)
         decoder_lstm = tf.keras.layers.LSTM( 256 , return_state=True , return_sequences=True , recurrent_dropout=0.2 , dropout=0.2)
         decoder_outputs , _ , _ = decoder_lstm ( decoder_embedding , initial_state=encoder_states )
         decoder_dense = tf.keras.layers.Dense( num_output_tokens , activation=tf.keras.activations.softmax )
         output = decoder_dense ( decoder_outputs )

         model = tf.keras.models.Model([encoder_inputs, decoder_inputs], output )
         model.compile(optimizer=tf.keras.optimizers.Adam(), loss='categorical_crossentropy')
```

Figure 14: Implementation of Recurent Neural network sample code

5.8. Model Testing and Evaluation

The model was tested through automatic testing by accuracy metrics and loss metrics depend up on the batch size, training epochs and data size. Both models are tested through booth metrics . In addition to this the proposed models are tested by humans.

CHAPTER SIX

6. Result and Discussions

In this chapter, the result of the proposed solution for Developing Afan Oromo Text based Chatbot for Enhancing Maize Productivity using Deep Neural Network algorithm and Recurent neural network are discussed. Here also the result of labelling data depend on the intent classification. At the end we discussed the experiment of the model built in each processes.

6.1. Dataset Labeling Result

As we discussed in chapter three dataset that we used for this study is taken from three difffrerent organization like Oromia Agriculture buruae , Oromia research center and Ethipian agriculture transformation agency(ATA). All the data are the text script. The data are labeled in the form of question and response based the intents. When we labeled the data we used intent, tag, pattern and response. We used tag as the class of the intent, pattern as the question and response the output for the pattern discussed in the dataset. As sentences level 500 sentences was labeled and as word level 2800 Afaan Oromoo words was labeled for this study. In general we have 100 pattern or question for 400 different responses. Each pattern has different responses depend on the intents the message from the user.

Table 5: Dataset labelling result

Type	Amount
Tag	25
Pattern	100
Response	400
Sentence	500
words	2800

6.2. Models Evaluation Result

In this section, we explain some experimental results on the bases of the training set through adjusting training data(%), batch size, activation function and training epochs. We observed that the experimental result of the proposed method in a sufficiently good to get response for the pattern(question) that the user asked the bot. For this study we proposed two different deep learning which is Deep neural network and Recurent Neural Network(Seq2seq model) both models are used different neural network layers.

6.2.1. ***Deep*** neural Network Training Results

In this section we evaluate the performance of Deep neural network to develop Afaan Oromoo Text-based Chatbot for enhancing maize productivity. When we evaluated this model we used training data , batch size, softmax activation fuction and training epochs. In case of this model the model is evaluated in terms of training accuracy and training loss of the model. On other hand this model was evaluated how exactly get the calss of the intents. This model has different neural network layers and has 1000 training epochs. As the number of training epoch increases the accuracy of our model also increases and the training loss of the model become decreased. In this case we used four neural network layers. The layers are one input, two hidden layers and one output layers. The sample result of training epochs, training loss and training accuracy based on the training epochs and the results.

Table 6: Deep neural Network Training Results

Neural network layers	*Epochs*	*Training accuracy*	*Training loss*
Four neural networks	*1-100*	0.2188	2.1008
	101 - 200	0.6250	1.2222
	201 - 300	0.7500	0.8341
	301 - 500	0.8125	0.7168
	501 – 650	0.8750	0.5231
	651 – 800	0.880	0.4574
	801 – 900	0.89	0.4336
	901 – 999	0.90	0.3733
	1000	0.923	0.2948

6.2.1.1.Intent class evaluation Result

In this case we evaluate our model how to get the class of the intent or how much the model can identify the class of the response based on pattern (question) the from the user. The model identify the claas where the response are depend on the pattern given to the bot. The sample code and sample result are expressed below

```
def classify(sentence):
    # generate probabilities from the model
    results = model.predict([bow(sentence, words)])[0]
    # filter out predictions below a threshold
    results = [[i, r] for i, r in enumerate(results) if r > ERROR_THRESHOLD]
    # sort by strength of probability
    results.sort(key=lambda x: x[1], reverse=True)
    return_list = []
    for r in results:
        return_list.append((classes[r[0]], r[1]))
    # return tuple of intent and probability
    return return_list

def response(sentence, userID='123', show_details=False):
    results = classify(sentence)
    # if we have a classification then find the matching intent tag
    if results:
        # loop as long as there are matches to process
        while results:
            for i in intents['intents']:
                # find a tag matching the first results
                if i['tag'] == results[0][0]:
                    # a random response from the intent
                    return print(random.choice(i['responses']))
```

```
# ===
data = pickle.load(open("training_data", "rb"))
words = data['words']
classes = data['classes']
train_x = data['train_x']
train_y = data['train_y']

import json
with open('C:/Users/Sanyichoo/chat.json') as json_data:
    intents = json.load(json_data)

# load saved model
net = tflearn.input_data(shape=[None, len(train_x[0])])
net = tflearn.fully_connected(net, 8)
net = tflearn.fully_connected(net, 8)
net = tflearn.fully_connected(net, len(train_y[0]), activation='softmax')
net = tflearn.regression(net)
model = tflearn.DNN(net, tensorboard_dir='tflearn_logs')
model.load('./model.tflearn')

print (classify('Akkam jirtu?'))
print (classify('Omishaa boqqoloo dabaluuf maal gochuun qaba?'))
print (classify('akkamin lafa lafa qonnaa qophessaa?'))
print (classify('Sanyii gosaa meeqatu jiraa?'))
print (classify('naannoo qilleensa baramaa kamtu boqolloof gaariidha?'))
```

Figure 15: Intent classification sample code

The result from the above sample is

```
[('nageenya', 0.8947484)]
[('Omishaaf', 0.92373466)]
[('lafa', 0.85000485)]
[('Sanyii ', 0.7976578)]
[('Qilleensa', 0.7634855)]
```

Figure 16: Intent classification sample result

6.2.2. Testing of DNN Model For Afaan Text-based Chatbot

Under this section the testing of the testing of the Chatbot performed by human. In most of the research that conducted on chatbot the accurate testing is through human.

Table 7: Testing result of DNN model

No	Expert	Question/pattern	Target response	Missed Response
1.	Person1	6	5	1
2.	Person2	5	5	0
3.	Person3	6	4	2
4.	People4	7	6	1
5.	People5	4	4	0
6.	People6	5	4	1
7.	People7	6	5	1
8.	People8	6	5	1
Total		45	38	7

As the result from Table 7: shows the experiment conducted 8 expert for testing the result. Out of 45 question the model answer 38/45 exact(target) respose, the model misses 7/45 question. From the experiment we conclude that the testing experiment of the model scored 0.844.or 84.4% accuracy, that is good result result. The experiment also shows the model misses some target question/pattern. As Table 7. Shows there are 7/45 question which is missed the target response , that is 0.156 or 15.6% loss scored value are there for DNN model for this study.

```
User: Omishaa boqqoloo dabaluuf maal gochuun qaba
ChatBot: nagaan ollaa
User: Akkamitti omisha boqqoloo dabaluu dandaha
ChatBot: gosa lafa qonnaa kessani beeku
User: akkamin lafa lafa qonnaa qophessaa
ChatBot: Lafti kessaan jabaata yoo tahe yeroo sadii yooakn afuur qotuu qabdu
User: yeroo meeqa qulleessa
ChatBot: qotiisa marsaa dhumaa sirritti qotuu qabdu
User: "Sanyii gosaa meeqatu jiraa
ChatBot: haala qilleensa irraatti hundaahun bakka saditti qoodama.
User: Gosni sanyii akkamitti murtaaha
ChatBot: Lamanuu barbaachisadha yeroo gargaaratti tahu qaba
User: Gosa isaa kamtu omishaaf gaariidha
ChatBot: Lamanuu barbaachisadha yeroo gargaaratti tahu qaba
User: Xaa'oo akkaatan itti fayyadamnuu haala rooba arganna irraatti hundahaa
ChatBot: Xaa'oo akkaatan itti fayyadamnuu haala rooba arganna irraatti hundahaa
User: Aanaalee Kibbaa Oromiyaatti haali itti fayyadama baruuf
ChatBot: Lafti qonnaa kessaan Oromiyaa lixaa yoo tahe heekteraatti yuuriyaa kuntaala lama fi walakaa fi xaa'oo mixxiin 1 fayyad
ama
User: Aanaalee Kibbaa Oromiyaatti haali itti fayyadama baruuf
ChatBot: Omishtummaan kessaan kibbaa Oromiyaa yoo tahe xaa'oo yuuriyaa kuntaala lama fi walakaa fi xaa'oo mixxiin kuntaala tokk
o
User: Dabalin xaa'oon haalan qilleensan murtaahu dandaha
ChatBot: Eyyeen haali qilleensa omishi kessaan kessaa jiruu murtessaadha
User: quit
```

Figure 17: Testing of Deep neural network sample from user

6.2.3. Recurent Neural Network Training Results

In this section we evaluate the performance of Recurent neural network model to Afaan Oromoo Text-based Chatbot for enhancing maize productivity. When we evaluated this model we used training data , batch size, softmax activation fuction and training epochs. In case of this model the model is evaluated in terms of training accuracy and training loss of the model. This model has different neural network layers and has 100 training epochs. As the number of training epoch increases the accuracy of our model also increases and the training loss of the model become decreased. In this case we used four neural network layers. The layers are one input, two hidden layers and one output layers. The sample result of training epochs, training loss and training accuracy based on the training epochs and the results. As table 8. Results shows as the at the beginning of training the model, the training accuracy of our model is very less and our model scored high loss at the beginning of training the model. At the end of the training results automatically changed. Training accuracy scored high values and training loss scored small values. Generally, when epoch number increases training accuracy increases while training loss decrease as our experiment results shows.

Table 8: RNN based Seq2Seq results

Neural network layers	Epochs	Training accuracy	Training loss
Four neural networks	1-10	0.1738	1.2465
	11 - 20	0.1175	1.1155
	21 - 30	0.1363	0.9616
	31 - 40	0.2351	0.8071
	41 - 50	0.3492	0.6815
	51 - 60	0.4566	0.5908
	61 - 70	0.5707	0.5022
	71 - 80	0.6712	0.4181
	81-90	0.7649	0.2948
	91 - 99	0.8024	0.2997
	100	0.8143	0.2934

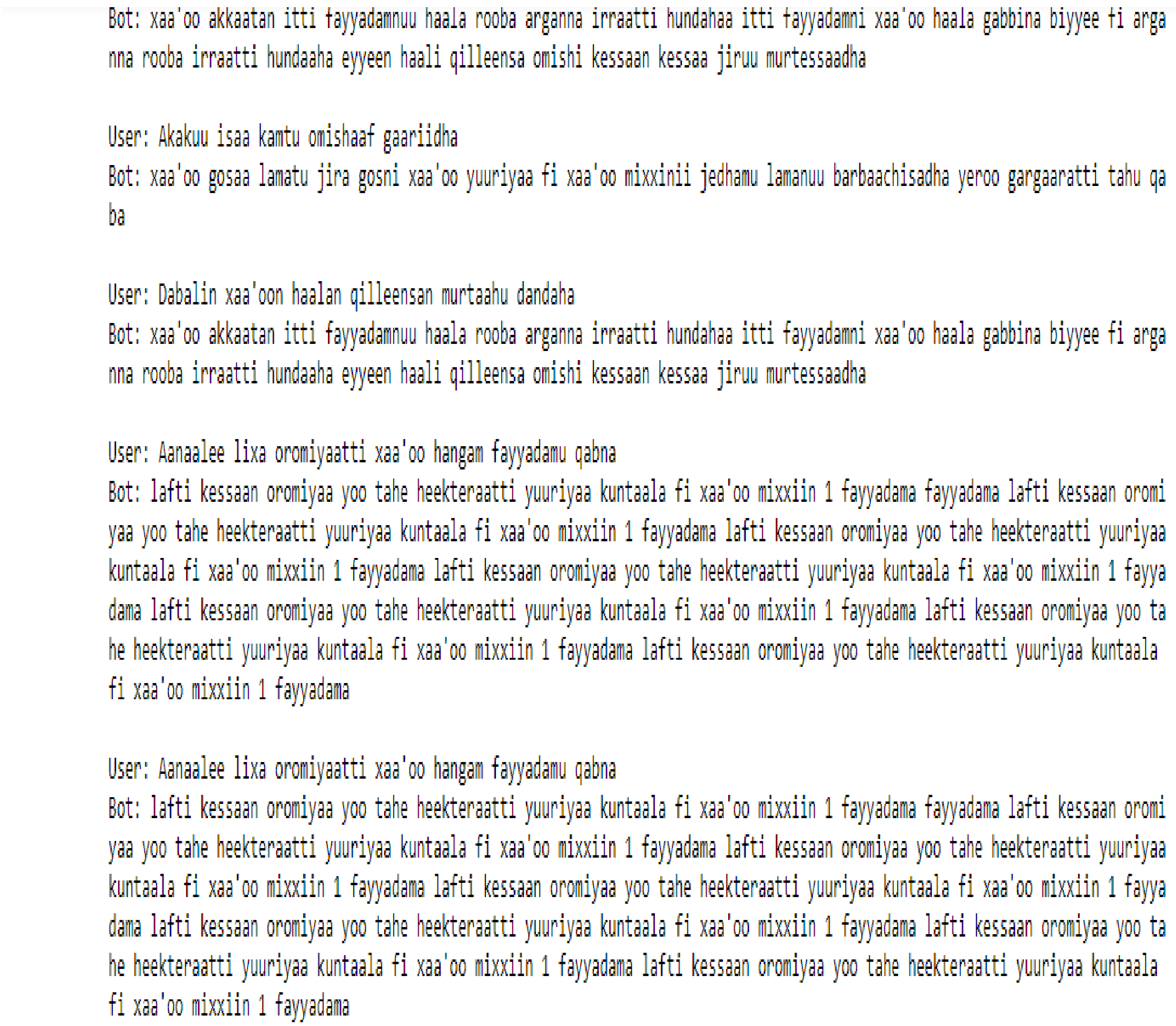
6.2.4. Tesing of Recurent neural network model for Afaan Oromoo text-based Chatbot

Under this section the testing of the testing of the Chatbot performed by human. In most of the research that conducted on chatbot the accurate testing is through human.

Table 9: Testing Result of RNN(Seq2Seq) model

No	Expert	Question/pattern	Target response	Missed Response
1.	Person1	6	4	2
2.	Person2	5	5	0
3.	Person3	6	4	2
4.	People4	7	5	2
5.	People5	4	4	0
6.	People6	5	4	1
7.	People7	6	5	1
8.	People8	6	5	1
Total		45	36	9

As the result from Table 9: shows the experiment conducted 8 expert for testing the result. Out of 45 question the model answer 36/45 exact(target) respose, the model misses 9/45 question. From the experiment we conclude that the testing experiment of the model scored 0.80.or 80% accuracy. The experiment also shows the model misses some target question/pattern. As Table 9. Shows there are 9/45 question which is missed the target response , that is 0.20 or 20% loss scored value are there for RNN(Seq2Seq) model for this study.



Bot: xaa'oo akkaatan itti fayyadamnuu haala rooba arganna irraatti hundahaa itti fayyadamni xaa'oo haala gabbina biyyee fi arga
nna rooba irraatti hundaaha eyyeen haali qilleensa omishi kessaan kessaa jiruu murtessaadha

User: Akakuu isaa kamtu omishaaf gaariidha
Bot: xaa'oo gosaa lamatu jira gosni xaa'oo yuuriyaa fi xaa'oo mixxinii jedhamu lamanuu barbaachisadha yeroo gargaaratti tahu qa
ba

User: Dabalin xaa'oon haalan qilleensan murtaahu dandaha
Bot: xaa'oo akkaatan itti fayyadamnuu haala rooba arganna irraatti hundahaa itti fayyadamni xaa'oo haala gabbina biyyee fi arga
nna rooba irraatti hundaaha eyyeen haali qilleensa omishi kessaan kessaa jiruu murtessaadha

User: Aanaalee lixa oromiyaatti xaa'oo hangam fayyadamu qabna
Bot: lafti kessaan oromiyaa yoo tahe heekteraatti yuuriyaa kuntaala fi xaa'oo mixxiin 1 fayyadama fayyadama lafti kessaan oromi
yaa yoo tahe heekteraatti yuuriyaa kuntaala fi xaa'oo mixxiin 1 fayyadama lafti kessaan oromiyaa yoo tahe heekteraatti yuuriyaa
kuntaala fi xaa'oo mixxiin 1 fayyadama lafti kessaan oromiyaa yoo tahe heekteraatti yuuriyaa kuntaala fi xaa'oo mixxiin 1 fayya
dama lafti kessaan oromiyaa yoo tahe heekteraatti yuuriyaa kuntaala fi xaa'oo mixxiin 1 fayyadama lafti kessaan oromiyaa yoo ta
he heekteraatti yuuriyaa kuntaala fi xaa'oo mixxiin 1 fayyadama lafti kessaan oromiyaa yoo tahe heekteraatti yuuriyaa kuntaala
fi xaa'oo mixxiin 1 fayyadama

User: Aanaalee lixa oromiyaatti xaa'oo hangam fayyadamu qabna
Bot: lafti kessaan oromiyaa yoo tahe heekteraatti yuuriyaa kuntaala fi xaa'oo mixxiin 1 fayyadama fayyadama lafti kessaan oromi
yaa yoo tahe heekteraatti yuuriyaa kuntaala fi xaa'oo mixxiin 1 fayyadama lafti kessaan oromiyaa yoo tahe heekteraatti yuuriyaa
kuntaala fi xaa'oo mixxiin 1 fayyadama lafti kessaan oromiyaa yoo tahe heekteraatti yuuriyaa kuntaala fi xaa'oo mixxiin 1 fayya
dama lafti kessaan oromiyaa yoo tahe heekteraatti yuuriyaa kuntaala fi xaa'oo mixxiin 1 fayyadama lafti kessaan oromiyaa yoo ta
he heekteraatti yuuriyaa kuntaala fi xaa'oo mixxiin 1 fayyadama lafti kessaan oromiyaa yoo tahe heekteraatti yuuriyaa kuntaala
fi xaa'oo mixxiin 1 fayyadama

Figure 18: Sample Testing Result for RNN Model

6.3. Discussion

Under this section we discuss the result of the proposed solution. Since developing Afaan Oromoo text-based Chatbot was the first time in our case the result of our proposed solution was good enough. In our study we Select Retriieval based approach to develop Afaan Oromoo Text-based chatbot for enhancing maize productivity. The reason that we select retrieval approach is that our study is depend up on closed domain Chatbot approach. Which means it depend on only specific knowdge area, Since our study is only about enhancing maize productivity the selected approach is good in order access the information that is important for enhancing maize productivity in our study.

As we select retrieval based Chatbot approach for modeling, we select two Deep learning models which is good for this approach. In diffrenet Deep learning algorithm we select Deep neural network and recurrent Neural Network which is good to fit our selected approach.Both models takes sequential data for training as well as for testing our models. Both models has four network layers which is Input layer, hidden layer and output layers. In both cases we used two hidden layers. A hidden layer in an artificial neural network is a layer in between input layers and output layers, where artificial neurons take in a set of weighted inputs and produce an output through an activation function. It is a typical part of nearly any neural network in which engineers simulate the types of activity that go on in the human brain. a hidden layer is located between the input and output of the algorithm, in which the function applies weights to the inputs and directs them through an activation function as the output. In short, the hidden layers perform nonlinear transformations of the inputs entered into the network. Hidden layers vary depending on the function of the neural network, and similarly, the layers may vary depending on their associated weights.

The experimented result from both model , Deep neural network and recurrent neural network shows good results. In case of Deep neural network (DNN) let as take sample of input(pattern/question), predicted response and target response. The trainining accuracy of the DNN model is 92.3% accuracy, which good result for our study. To train this we used 1000 epoches for training the model. Since the validation testing of the study conducted by human or user input value, table 10. Shows that how the bot was response for the user pattern or question.

Table 10:Predicted Result from model

Input/pattern/question	*akkamitti omisha boqolloo dabaluu dandaha*
Target response	*gosa lafa qonnaa kessani beeku*
Predicted response	*gosa lafa qonnaa kessani beeku*
Input/pattern/question	*sanyi gosa meeqa jira*
Target response	*Gosni sanyii gosa afurtu jira*
Predicted response	*Gosni xaa'oo Yuuriyaa fi xaa'oo Mixxinii jedhamu*
Input/pattern/question	*akkamin lafa qonnaa qophessaa*
Target response	*Lafti kessaan jabaata yoo tahe yeroo sadii yooakn a fuur qotuu qabdu*
Predicted response	*Lafti kessaan jabaata yoo tahe yeroo sadii yooakn a fuur qotuu qabdu*
Input/pattern/question	*Gadi fageenyi biyyee hanga*
Target response	*hamma dheerina quuba kessaani isaa xiqqaa gadi faga achu qaba*
Predicted response	*hamma dheerina quuba kessaani isaa xiqqaa gadi faga achu qaba*
Input/pattern/question	*Aanaalee lixa oromiyaatti xaa'oo hangam fayyadamu q abna*
Target response	*Lafti qonnaa kessaan Oromiyaa lixaa yoo tahe heekte raatti yuuriyaa kuntaala 3 fi xaa'oo mixxiin 1 fayy adama*
Predicted response	*Lafti qonnaa kessaan Oromiyaa lixaa yoo tahe heekte raatti yuuriyaa kuntaala 3 fi xaa'oo mixxiin 1 fayy adama*
Input/pattern/question	*Omishaa boqqoloo dabaluuf maal gochuun qaba*
Target response	*Naannoo qillensa kessaa jiraatani beeku*
Predicted response	*homa rakkoo hin qabu*

This study experimented through two different deep learning algorithms DNN and RNN based Seq2Seq model. In DNN algorithms we got higher accuracy in both training and testing of the model. Since the testing the both models conducted manual by taking input from the user and predicting the responses of the bot, developing Afaan Oromoo Text-based Chatbot based on DNN model testing results are **84.4%** accuracy based on the question from the user and the targeted responses. Since Afaan Oromoo Text-based Chatbot based on DNN algorithms is not developed until this studies the experimented result is good enough. The training accuracy of the algorithms is much more than testing accuracy of the models, since the training accuuarcy result is automatic. The intent class prediction is also performed in this case. The class of the intents are predicted by taking question/pattern and predicting the clas/tag of the question where that question/pattern are existed.

As we discussed above we proposed RNN algorithm based Seq2Seq models for this studies. Since we used the same dataset for both models the result of experimented based on RNN based on Seq2Seq model is also depend on the training accuracy and testing accuracy , the difference was only in Seq2Seq model used Encoder and Decoder layers for Input and Output. In our case both models used testing by human, by taking Input (question from the user) and predicting responses from the bot depend on the predefined conversation training dataset. The training result of RNN based Seq2Seq is scored **81**% training accuracy for this model. We used 100 epochs for training our model. Since the validation testing of the model is accepting input from the user we create the following cases for the validation of the model. For RNN based Seq2Seq model we take sample of 45 question for 8 experts. In this case the bot accurately responds 36 question out 45 question/pattern. The predicting accuracy of RNN based Seq2Seq model is **80**%.

Table 11: models result summary

Models	*Training accuracy*	*Training loss*	*Testing accuracy*
DNN	*0.923*	*0.2948*	*0.844*
RNN(Seq2Seq)	*0.81*	0.2860	*0.80*

As table 11. Shows the experiment of DNN is scoring better tesing accuracy than RNN model in our study.So, in our study we recommend Deep Neural network model which the testing result is higher than recurrent Neural Network as our experiment result show.When we compared the way of accessing information , our study is better for accessing information for farmer for inhancing their productivity over the existing system. The existing system was boring for accessing information since there is a lot of tackle as we arise in our study in the problem defining section. But, our study simplified the way of accessing information for enhancing their product for the farmer. Because our study addressed the problem that defined in our study.

CHAPTER SEVEN

7. Conclusion, Recommendation and Future work

7.1. Conclusion

In this studies , we addressed our results based on two deep learning algorithm DNN and RNN based on Seq2Seq model. In this experimental study, we proposed two deep learning model which is good for the determined problem. Both model are depend on the pattern(question) and response.Study was also based retrieval based Chatbot , which is depend on predefined conversational dataset.

Since the studies is based domain based or knowledge area Chatbot the study was carried out with small size of dataset. Experiments on developing Afaan Oromoo text-based Chatbot for enhancing maize productivity tasks show that our system achieves significant results on small amount of the training corpus collected from Oromia agricultural buereu , Oromia research institute center and Agricultural transformation agency. The proposed method achieved better performance for developing Afaan Oromoo text-based Chatbot for Retrieval based Chatbot.

Under this section we summarizes the findings and contributions made on this study. Firstly, we proposed a deep learning approach for developing Afaan Oromoo text-based Chatbot for enhancing maize productivity by using DNN and RNN based on Seq2Seq model approach. Secondly, the study introduce the Afaan Oromoo language to deep learning techniques. Thirdly, we contribute dataset set for users, researchers and developers of NLP tasks. Finally, this research scored a good results for both models based expert tesing and training accuracy results. The summary of the above finding.

- Firstly, we propose a deep learning approach for developing Afaan Oromoo text-based Chatbot for enhancing maize productivity by using DNN and RNN based on Seq2Seq model approach.
- Secondly, the study introduce the Afaan Oromoo language to deep learning techniques.
- Thirdly, we contribute dataset set for users, researchers and developers of NLP tasks.
- Finally, this research scored a good results for both models based expert tesing and training accuracy results

7.2. Recommendation and Future works

The proposed model are the capability of responding the exact target responses based on the real time learning by changing environments because they have better performance and solves runtime problems. We strongly recommends those who want to use and reference this research with in the summary of the following points.

Dataset: Based on the dataset , the studies conducted with small data size , so we recommend more large scale training for dataset and testing dataset.

Domain of the studies: The studies depend on the specific domain area that is how developing Afaan Oromoo text-based for enhancing maize productivity , We recommend other domain area.

Model approach: The model approach for proposed system is retrieval approach that we select only DNN and RNN based on Seq2Seq Models , there are other deep learning approach for developing the same system, so we recommend other Deep learning approach like LSTM , BiLSTM , BERT, CNN and so on.

Running Time: We use CPU for this studies for training our dataset , we recommend GPU for training the models and valiadating models.

Finaly, our studies depend Retrieval chatbot which only on predefined data which is not conceder the context of the question rather than which is predefined before. As a future work, we formulate the other version of Chatbot which Generative based Chatbot.This type of Chatbot generate the response based the context of the question. We also try to increase the size of our dataset to get good enough accuracy scoring. We also try to incorporate more than two models compare their results.

SPECIAL ACKNOWLEDGEMENT

This research project is funded by Adama Science and Technology University under the grant number

ASTU/SM-R/096/19

Adama, Ethiopia

REFERENCES

[1] T. Urgessa, “The determinants of agricultural productivity and rural household income in Ethiopia,” *Ethiop. J. Econ.*, vol. 24, no. 2, pp. 63–91, 2015.

[2] A. Deshpande, A. Shahane, D. Gadre, M. Deshpande, and P. M. Joshi, “a Survey of Various Chatbot Implementation Techniques,” *Int. J. Comput. Eng. Appl.*, vol. XI, no. ISSN 2321-3469, p. 7, 2017.

[3] B. Issd, “Updated Version of Analysis of Seed System in Oromia : Opportunities for Improvement Oromia National Regional State , Bureau of Agriculture and Natural Resources,” no. January, 2020.

[4] U. T. Bekabil, “Review of Challenges and Prospects of Agricultural Production and Productivity in Ethiopia,” vol. 4, no. 18, pp. 70–78, 2014.

[5] W. Mwangi, *Farmers ’ Maize Seed Systems In Western Oromia , Farmers ’ Maize Seed Systems in Western Oromia* , no. June. 2001.

[6] M. T. ZEMČÍK, “A Brief History of Chatbots,” *DEStech Trans. Comput. Sci. Eng.*, no. aicae, 2019.

[7] B. A. Shawar and E. Atwell, “ALICE chatbot: Trials and outputs,” *Comput. y Sist.*, vol. 19, no. 4, pp. 625–632, 2015.

[8] M. Jo, P. Fialho, and R. Ribeiro, “Chatbots ’ Greetings to Human-Computer Communication,” no. October, pp. 24–25, 2018.

[9] G. Molnar and Z. Szuts, “The Role of Chatbots in Formal Education,” *SISY 2018 - IEEE 16th Int. Symp. Intell. Syst. Informatics, Proc.*, no. September 2018, pp. 197–201, 2018.

[10] V. Consulting, “Chatbots Chatbots 101 by V-Soft Consulting,” p. 18, 2017.

[11] A. Vichare, A. Gyani, Y. Shrikhande, and N. Rathod, “A chatbot system demonstrating Intelligent Behaviour using NLP,” vol. 4, no. 10, pp. 4–6, 2015.

[13] J. Seering, M. Luria, G. Kaufman, and J. Hammer, “Beyond dyadic interactions: Considering chatbots as community members,” *Conf. Hum. Factors Comput. Syst. - Proc.*, 2019.

[14] J. Grudin and R. Jacques, “Chatbots, humbots, and the quest for artificial general intelligence,” *Conf. Hum. Factors Comput. Syst. - Proc.*, 2019.

[15] M. S. Thesis, I. Kulatska, M. Theune, D. K. J. Heylen, and J. B. Van Waterschoot, “ArgueBot: Enabling debates through a hybrid retrieval-generation-based chatbot.”

[16] Hubtype, “Rule-Based vs AI Chatbots,” *Hubtype*, pp. 1–6, 2019.

[17] Y. Song, C. Te Li, J. Y. Nie, M. Zhang, D. Zhao, and R. Yan, “An ensemble of retrieval-based and generation-based human-computer conversation systems,” *IJCAI Int. Jt. Conf. Artif. Intell.*, vol. 2018-July, pp. 4382–4388, 2018.

[18] B. Di Martino, C. Mirarchi, S. Ciuffreda, and A. Pavan, “Complex, Intelligent, and Software Intensive Systems,” *Complex, Intelligent, Softw. Intensive Syst. CISIS 2019*, vol. 993, no. January, pp. 837–846, 2020.

[19] E. Adamopoulou and L. Moussiades, “Machine Learning with Applications Chatbots :

History , technology , and applications," *Mach. Learn. with Appl.*, vol. 2, no. October, p. 100006, 2020.

[20] E. A. B, *An Overview of Chatbot Technology*. Springer International Publishing, 2020.

[21] F. A. Mikic-Fonte, M. Llamas-Nistal, and M. Caeiro-Rodriguez, "Using a Chatterbot as a FAQ Assistant in a Course about Computers Architecture," *Proc. - Front. Educ. Conf. FIE*, vol. 2018-Octob, 2019.

[22] C. Tack, "Artificial intelligence and machine learning | applications in musculoskeletal physiotherapy," *Musculoskelet. Sci. Pract.*, vol. 39, pp. 164–169, 2019.

[23] B. Abu Shawar and E. Atwell, "A chatbot system as a tool to animate a corpus," *ICAME J. Int. Comput. Arch. Mod. Mediev. English J.*, vol. 29, pp. 5–24, 2005.

[24] G. Ochoa, "a Rtificial I Ntelligence (Csc9Ye)," vol. 4, no. 4, pp. 1–10, 2016.

[25] A. Stöckl, "Classification of Chatbot Inputs," no. July, 2017.

[26] E. Pratt, "So what does this mean for technical communicators and for User Assistance? Artificial Intelligence and Chatbots in Technical Communication – A Primer Artificial Intelligence and Chatbots in Technical Communication – A Primer," *Iiblog*, p. 2, 2017.

[27] M. Boyanov, I. Koychev, P. Nakov, A. Moschitti, and G. Da San Martino, "Building chatbots from forum data: Model selection using question answering metrics," *Int. Conf. Recent Adv. Nat. Lang. Process. RANLP*, vol. 2017-Septe, pp. 121–129, 2017.

[28] L. O. G. In and S. I. G. N. Up, "Bot s I ntelligent Chatbot s : A G u i d e to B u il d ing an d U s ing I ntelligent Agent s," pp. 1–29, 2019.

[29] A. R. October, "the Agriculture Knowledge , Learning , Documentation and Policy Project Annual Report October 2015 – September 2016 the Agriculture Knowledge , Learning , Documentation," no. October 2015, 2016.

[30] U. T. Bekabil, "Review of challenges and prospects of agricultural production and productivity in Ethiopia.," *J. Nat. Sci. Res.*, vol. 4, no. 18, pp. 70–77, 2014.

[31] A. M. Rahman, A. Al Mamun, and A. Islam, "Programming challenges of chatbot: Current and future prospective," *5th IEEE Reg. 10 Humanit. Technol. Conf. 2017, R10-HTC 2017*, vol. 2018-Janua, no. March 2018, pp. 75–78, 2018.

[32] W. Tesema, D. Tesfaye, and T. Kibebew, "Towards the Sense Disambiguation of Afan Oromo Words Using Hybrid Approach (Unsupervised Machine Learning and Rule Based)," pp. 61–77.

[34] M. I. Aguilar *et al.*, "Oromo Studies," *J. Oromo Stud. III Number 1/2 Winter/Summer 1996 ISBN 1070-2202*, vol. III, pp. 1881–1900, 1996.

[35] W. Tegegne, "The Development of Written Afan Oromo and the Appropriateness of Qubee , Latin Script , for Afan Oromo Writing," *Hist. Res. Lett.*, vol. 28, no. 1976, pp. 8–14, 2016.

[36] A. Seyoum Taffesse, P. Dorosh, and S. A. Gemessa, "Crop production in Ethiopia: Regional patterns and trends," *Food Agric. Ethiop. Prog. Policy Challenges*, vol. 9780812208, pp. 53–83, 2013.

[37] P. Y. D. K, R. Hemalatha, and G. Niveditha, "Smart Chatbot for Agriculture," vol. 9, no. 5, pp. 22203–22205, 2019.

[38] A. Us, "Chatbot for Farmers and Agriculture," pp. 1–10, 2019.

[39] H. Murase, "Artificial intelligence in agriculture," *Comput. Electron. Agric.*, vol. 29, no. 1–2, pp. 1–2, 2000.

[40] W. Vota, "ICTworks FarmChat : Using Chatbots to Answer Farmer Queries in India," *ICTworks*, pp. 1–5, 2019.

[41] M. Jain, P. Kumar, I. Bhansali, Q. V. Liao, K. Truong, and S. Patel, "FarmChat: A Conversational Agent to Answer Farmer Queries," *Proc. ACM Interactive, Mobile, Wearable Ubiquitous Technol.*, vol. 2, no. 4, pp. 1–22, 2018.

[43] F. Peters, "Design and implementation of a chatbot in the context of customer support," 2018.

[44] L. C. Hosting, "Machine Learning or Linguistic Rules : Two Approaches to Building a Chatbot," pp. 10–13, 2019.

[45] A. Bartl and G. Spanakis, "A retrieval-based dialogue system utilizing utterance and context embeddings," *Proc. - 16th IEEE Int. Conf. Mach. Learn. Appl. ICMLA 2017*, vol. 2017-Decem, pp. 1120–1125, 2017.

[46] O. Vinyals and Q. Le, "A Neural Conversational Model," vol. 37, 2015.

[47] A. Tiha, "Intelligent Chatbot using Deep Learning," no. April, pp. 1–22, 2018.

[48] D. Morales, H. Nguyen, and T. Chin, "A Neural Chatbot with Personality," *Comput. Sci. Dep. Stanford Univ.*, 2017.

[49] M. Mahmoud and R. Kumar, "A Review on Chatbot Design and Implementation Techniques," *Int. J. Eng. Technol.*, vol. 7, no. April, p. 2791, 2020.

Annex A: Cost Break down

I expect the entire Budget source will be covered by Adama Science and Technology University

Table 1.1 Business plan

No	ITEM	Quantity /Unit	Unit Price	Total
1	Pen (Lexi)	1packet	10*20.00	200.00
2	DVD-RW	4pcs	4*50.00	200.00
3	Flash Disk 16GB	2Pcs	2*400.00	800.00
4	Mobile Card	---	3500.00	3500.00
5	Note Book	1packets	1*200.00	200.00
6	Photo copy(printing)	9*100	2*900.00	1800.00
7	Internet	---	2000.00	2000.00
10	Transport Approximately	45days*200(cost per trip)	45*200.00	9000.00
11	White Paper	1packet	1*300.00	300.00
12	Researcher support	7	7*1000.00	7000.00
			Total (In Ethiopian BIRR)	25000.00

ANNEX B. Sample Code

B.1. Sample source code defining package

```
import numpy as np
import tensorflow as tf
from tensorflow.keras import layers , activations , models , preprocessing , utils
import pandas as pd

print( tf.VERSION )
```

B.2. Sample source code for preprocessing Pattern, response and tag.

```
decoder_target_data = list()
for token_seq in tokenized_output_lines:
    decoder_target_data.append( token_seq[ 1 : ] )

padded_output_lines = preprocessing.sequence.pad_sequences( decoder_target_data , maxlen=max_output_length, padding='post' )
onehot_output_lines = utils.to_categorical( padded_output_lines , num_output_tokens )
decoder_target_data = np.array( onehot_output_lines )
print( 'Decoder target data shape -> {}'.format( decoder_target_data.shape ))
```

B.3. Sample source code for Deep Neural Netwoork

```
# Build Deep neural network
net = tflearn.input_data(shape=[None, len(train_x[0])])
net = tflearn.fully_connected(net, 8)
net = tflearn.fully_connected(net, 8)
net = tflearn.fully_connected(net, len(train_y[0]), activation='softmax')
net = tflearn.regression(net)
model = tflearn.DNN(net, tensorboard_dir='tflearn_logs')
model.fit(train_x, train_y, n_epoch=1000, batch_size=8, show_metric=True)
model.save('model.tflearn')
```

B.4. Sample code for DNN interface

```
    while True:
        print(Fore.LIGHTBLUE_EX + "User: " + Style.RESET_ALL, end="")
        inp = input()
        if inp.lower() == "quit":
            break
        result = model.predict(keras.preprocessing.sequence.pad_sequences(tokenizer.texts_to_sequences([inp]),
                                             truncating='post', maxlen=max_len))
        tag = lbl_encoder.inverse_transform([np.argmax(result)])
        for i in data['intents']:
            if i['tag'] == tag:
                print(Fore.GREEN + "ChatBot:" + Style.RESET_ALL , np.random.choice(i['responses']))
        # print(Fore.GREEN + "ChatBot:" + Style.RESET_ALL,random.choice(responses))
print(Fore.YELLOW + "Start messaging with the bot (type quit to stop)!" + Style.RESET_ALL)
chat()
```

B.5. Sample code for preparing for encoder

```
input_lines = list()
for line in lines.input:
    input_lines.append( line )

tokenizer = preprocessing.text.Tokenizer()
tokenizer.fit_on_texts( input_lines )
tokenized_input_lines = tokenizer.texts_to_sequences( input_lines )

length_list = list()
for token_seq in tokenized_input_lines:
    length_list.append( len( token_seq ))
max_input_length = np.array( length_list ).max()
print( 'Input max length is {}'.format( max_input_length ))

padded_input_lines = preprocessing.sequence.pad_sequences( tokenized_input_lines , maxlen=max_input_length , padding='post' )
encoder_input_data = np.array( padded_input_lines )
print( 'Encoder input data shape -> {}'.format( encoder_input_data.shape ))

input_word_dict = tokenizer.word_index
num_input_tokens = len( input_word_dict )+1
print( 'Number of Input tokens = {}'.format( num_input_tokens))
```

B.6. Sample code for preparing for decoder

```
output_lines = list()
for line in lines.output:
    output_lines.append( '<START> ' + line + ' <END>' )

tokenizer = preprocessing.text.Tokenizer()
tokenizer.fit_on_texts( output_lines )
tokenized_output_lines = tokenizer.texts_to_sequences( output_lines )

length_list = list()
for token_seq in tokenized_output_lines:
    length_list.append( len( token_seq ))
max_output_length = np.array( length_list ).max()
print( 'Output max length is {}'.format( max_output_length ))

padded_output_lines = preprocessing.sequence.pad_sequences( tokenized_output_lines , maxlen=max_output_length, padding='post' )
decoder_input_data = np.array( padded_output_lines )
print( 'Decoder input data shape -> {}'.format( decoder_input_data.shape ))

output_word_dict = tokenizer.word_index
num_output_tokens = len( output_word_dict )+1
print( 'Number of Output tokens = {}'.format( num_output_tokens))
```

B.7. Sample code for target decoder

```
decoder_target_data = list()
for token_seq in tokenized_output_lines:
    decoder_target_data.append( token_seq[ 1 : ] )

padded_output_lines = preprocessing.sequence.pad_sequences( decoder_target_data , maxlen=max_output_length, padding='post' )
onehot_output_lines = utils.to_categorical( padded_output_lines , num_output_tokens )
decoder_target_data = np.array( onehot_output_lines )
print( 'Decoder target data shape -> {}'.format( decoder_target_data.shape ))
```

B.8. Sample code for defining model for RNN

```
encoder_inputs = tf.keras.layers.Input(shape=( None , ))
encoder_embedding = tf.keras.layers.Embedding( num_input_tokens, 256 , mask_zero=True ) (encoder_inputs)
encoder_outputs , state_h , state_c = tf.keras.layers.LSTM( 256 , return_state=True , recurrent_dropout=0.2 , dropout=0.2 )( enc
encoder_states = [ state_h , state_c ]

decoder_inputs = tf.keras.layers.Input(shape=( None ,  ))
decoder_embedding = tf.keras.layers.Embedding( num_output_tokens, 256 , mask_zero=True) (decoder_inputs)
decoder_lstm = tf.keras.layers.LSTM( 256 , return_state=True , return_sequences=True , recurrent_dropout=0.2 , dropout=0.2)
decoder_outputs , _ , _ = decoder_lstm ( decoder_embedding , initial_state=encoder_states )
decoder_dense = tf.keras.layers.Dense( num_output_tokens , activation=tf.keras.activations.softmax )
output = decoder_dense ( decoder_outputs )

model = tf.keras.models.Model([encoder_inputs, decoder_inputs], output )
model.compile(optimizer=tf.keras.optimizers.Adam(), loss='categorical_crossentropy', metrics=['accuracy'])

model.summary()
```

B.9. Sample code for training model for RNN

```
model.fit([encoder_input_data , decoder_input_data], decoder_target_data, batch_size=128, epochs=100)
model.save( 'model.h5' )
```

B.10. Sample code for RNN interface

```
enc_model , dec_model = make_inference_models()

enc_model.save( 'enc_model.h5' )
dec_model.save( 'dec_model.h5' )
model.save( 'model.h5' )

for epoch in range( encoder_input_data.shape[0] ):
    states_values = enc_model.predict( str_to_tokens( input( 'User: ' ) ) )
    empty_target_seq = np.zeros( ( 1 , 1 ) )
    empty_target_seq[0, 0] = output_word_dict['start']
    stop_condition = False
    decoded_translation = ''
    while not stop_condition :
        dec_outputs , h , c = dec_model.predict([ empty_target_seq ] + states_values )
        sampled_word_index = np.argmax( dec_outputs[0, -1, :] )
        sampled_word = None
        for word , index in output_word_dict.items() :
            if sampled_word_index == index :
                decoded_translation += ' {}'.format( word )
                sampled_word = word

        if sampled_word == 'end' or len(decoded_translation.split()) > max_output_length:
            stop_condition = True

        empty_target_seq = np.zeros( ( 1 , 1 ) )
        empty_target_seq[ 0 , 0 ] = sampled_word_index
        states_values = [ h , c ]

    print( "Bot:" +decoded_translation.replace(' end', '') )
    print()
```