UNIVERSITY OF NAIROBI

DEPARTMENT OF COMPUTING AND INFORMATICS

FACULTY OF SCIENCE AND TECHNOLOGY

AI CHATBOT: IMPROVE EFFICIENCY IN HANDLING STUDENT QUERIES AT THE DEPARTMENT OF COMPUTING AND INFORMATICS, NAIROBI UNIVERSITY

BY

ONYALO WYCLIFFE AMOLLO

THIS PROJECT REPORT IS SUBMITTED IN PARTIAL FULFILLMENT OF THE REQUIREMENTS FOR THE AWARD OF THE DEGREE OF MASTER OF SCIENCE IN COMPUTATIONAL INTELLIGENCE OF THE UNIVERSITY OF NAIROBI

JUNE 2022

Declaration

This thesis report is my original work and has not been submitted for examination at any university for any award.

Signed…………… Date…28 July, 2022…

Wycliffe Onyalo

P52/37285/2020

This thesis report has been submitted for examination with my approval as the university supervisor.

Signed…………………. Date…29/07/2022...

Prof. Elisha T. Opiyo Omulo

Dedication

I dedicate this work to my mom, my lovely wife and my two little boys; Alvah and Aiden Onyalo

Acknowledgement

I thank the Almighty God for the gift of health and energy during this study. I sincerely thank my supervisor Prof. Elisha Opiyo for his patience and kind guidance throughout the course of this study. Lastly, I’m eternally grateful to my family and close friends for their encouragement and unconditional support throughout my study.

List of Figures

List of Tables

Contents

Abstract

Artificial Intelligent chatbots are increasingly replacing human chat service agents because of bot's ability to communicate with humans through natural language and artificial intelligence (AI) technologies. Studies have found that universities need to provide effective and efficient digital platforms, powered by AI, in order to support holistic virtual learning (Alharthi, Spichkova and Hamilton, 2019). The COVID-19 pandemic has disrupted face to face learning and most institutions have now adapted hybrid or pure online learning. In cases where new students are onboarded virtually, there has been an increase in student queries and the traditional channels of human support is becoming ineffective. This research aimed to develop a conversational AI Chabot that would improve efficiency in handling Student Queries at the Department of Computing and Informatics (DCI) at the University of Nairobi. Waterfall Software development methodology was used in this development study. The source of data was the content on the University of Nairobi website and DCI students. Content analysis and structured interviews were used to obtain the data. Natural Language Processing (NLP) and LSTM model were used to build the AI Chatbot (dubbed UniBot). BLUE evaluation method was used to assess the effectiveness of the UniBot in providing accurate responses. The research established a near perfect match in chatbot response with a BLUE score of 0.75. Quantitative approach was also adopted to evaluate the efficiency of the model by having 20 target students use the chatbot for a duration of 3 weeks and give their feedback through questionnaires. A mean score of 4.10 and standard deviation of 0.59 was obtained from the students' responses, meaning; the UniBot achieved its objective of improving efficiency in handling students' query. Particularly, students who interacted with the UniBot indicated that the bot was easy to use and could retrieve the needed information very fast. However, the chatbot was not able to answer questions on topics that it had not been exposed to and would leave the question unanswered. In such instances, it's recommended that the chatbot should provide relevant links or human contacts.

CHAPTER ONE: INTRODUCTION

1.1 Background

Artificial Intelligent chatbots are increasingly replacing human chat service agents given bot's ability to communicate with humans through natural language and artificial intelligence (AI) technologies. As observed by Adam, Wessel and Benlian (2021), cost-saving opportunities and time-saving opportunities have triggered the rise in the implementation of AI-based chatbots. Molnár and Szüts (2018) further state that the advancement in AI, increase in smartphones penetration and increased use of social messaging applications are the key factors driving global chatbot growth. In a study by IBM, Businesses handle 265 billion customer service calls per year resulting in annual expenditure of $1.3 trillion. Businesses can save this high cost by simply adopting intelligent chatbots. Chatbots can speed up response times and answer up to 80% of customers' queries. Just like any other industry, chatbots can also act as virtual assistants in the education sector. Molnár and Szüts (2018) observed that AI chatbots, if adopted by universities, can help create a big difference in the admissions process, support and faster access to relevant university information. This can improve *efficiency* in the overall student support by the university. According to Merriam Webster dictionary, *efficiency* is the capability of producing desired results with little or no waste of time.

1.2 Statement of the Problem

In a report by UNESCO (2020), approximately 1.6 billion students in one hundred and eighty-eight countries were directly impacted by the closure of institutions of learnings due to the containment measures that Governments around the world had put in place to help curb the spread of COVID-19. As a result, many universities quickly replaced face-to-face lectures with online learning. Schleicher (2021) however notes that universities should re-invent their learning environments by expanding online learning and digitalization which would then complement the relationships between the students and the institutions. Another study by Alharthi, Spichkova and Hamilton (2019), found out that universities need to provide effective and efficient digital platforms that supports holistic virtual learning.

In Kenya, the Government suspended physical learning across all learning institutions in the country in March 2020. This directive made many institutions of higher learning change their mode of learning to online. The University of Nairobi was among the institutions that adopted online classes for the students. The institution has a student population of 84,000 pursuing 540 academic programmes across 10 campuses. To manage students queries remotely, the institution has availed various online channels like email addresses and Telephone numbers where students can reach out for assistance on various issues. However, given the huge student population, handling multiple queries via traditional channels like emails or calls has proven difficult due the large numbers and they cannot all be served. Sandu (2020) points out that this manual approach, can hinder effective and timely student support. Furthermore, the National ICT Policy of 2019 require that services offered by the Government should be accessible anywhere and anytime to all citizens, even to their mobile devices.

1.3 Objectives

The overall objective of this study was to build a conversational AI Chabot that improves efficiency in handling Student Queries at the department of Computing and Informatics (DCI) in the University of Nairobi.

The **specific objectives** were as follows;

i. To undertake system analysis for a conversational AI Chabot that can support handling of students queries at the DCI, University of Nairobi
ii. To design a conversational AI Chabot for the University of Nairobi students
iii. To build a Recurrent Neural Network (RNN) Model for training the conversational AI Chabot
iv. To develop and deploy a conversational AI Chabot for answering student queries at the University of Nairobi
v. To evaluate the efficiency of the conversational AI Chabot in handling student queries at the University

1.4 Significance of the Study

This study has laid the foundation for the implementation of AI-based Chatbot to support student queries around Admission Process, Registration Process, Students Fees, Student Loans Status, Course Offered, Class Timetable, Examinations, Semester Start & End dates and Lecturers' contacts. The AI Chatbot can complement online learning at the University of Nairobi and other universities in Kenya. This study will benefit students at the department of Computing and Informatics by improving the response time to their queries. The chatbot will handle repetitive queries thus saving department administrators more time to focus on other niversity needs.

1.5 Scope of Study

The study focused on automating responses to frequent student queries at the department of Computing and Informatics at the University of Nairobi. The chatbot will only respond when engaged in short and correct English language. Lastly, the chatbot will answer queries that fall within the following topics; like Admission Process, Registration Process, Students Fees, Student Loans Status, Courses Offered, Class Timetable, Examinations, Semester Start & End dates and Lecturers' contacts.

1.6 Definition of Important Terms

a) **RNN** (Recurrent Neural Network) **–** This is a Neural Network where the input of the current step comes from the output of the previous step.

b) **TensorFlow** – This is an open-source software library for performing machine and deep learning.

c) **BLEU –** Bi-Lingual Evaluation Understudy is a score or metric used to evaluate a generated sentence to a reference sentence in NLP

d) **Efficiency -** An efficient system is one capable of producing desired results with little or no waste of time or materials.

e) **DCI –** Department of Computing and Informatics

f) **UniBot –** The developed AI chatbot for the DCI students at the University of Nairobi

1.7 Organization of the Project Report

Chapter one of this study gives a brief background of the study together with its objectives and significance. In the next chapter, chapter two, a detailed literature is reviewed. The chapter starts by outlining the history of chatbots, then looks at related studies in the field of AI chatbots before highlighting some of the gaps identified in the existing literature.

CHAPTER TWO: LITERATURE REVIEW

This chapter looks at the available literature materials in the field of AI and chatbot. It starts by highlighting the history of chatbots from Eliza in 1966 to Alexa in 2015. The chapter then details the research gap and finally outlines the architectural design that will be used to build the AI-chatbot for the University of Nairobi.

2.1 Introduction to Chatbots

Conversational AI chatbots have become popular over time. Currently, the chatbots are widely applied in ecommerce, telecommunication, online banking among others. AI chatbot technology has the potential to provide personalized service to different users, but unfortunately, its application in the education sector settings is still limited. (Yang and Evans, 2019).

Chatbot is a software application that communicates with human users through text or voice. To achieve this, chatbots use natural language and artificial intelligence (AI). In other words, chatbots are able to understands human languages through Natural Language Processing (NLP) and AI. In the education sector, chatbots can be trained to answer questions ranging from admission/registration process, payment issues, examination queries, accommodation issues, enrollment processes, class timetable and many more. Chatbots have many advantages. Hopkins and Silverman (2016) state that AI-chatbots can enable faster, convenient and cost-effective channels for supporting customers. Akkiraju et al. (2017) also note that, AI chatbots are beneficial in the sense that they offer cost and time saving opportunities, increases quality of services offered and improves customer experience. Frederick (2021) state that Africa has experienced a slower adoption of chatbots due to poorer understanding of how chatbots can add value to businesses. This is now changing though. Chatbots can be availed at various customer touchpoints including but not limited to websites and social media applications.

The main algorithms currently been used by modern AI chatbots are RNN, LSTM, NLP, Decision Trees, Naïve Bayes and Markov Chains. This study will however adopt RNN, LSTM and NLP because studies have shown that RNN and LSTM are the best algorithms for efficiently processing textual data. NLP on the other hand effectively enable the bot to understand human language, and therefore generate valid responses. Compared to other algorithms, RNN have been found to perform better sequence understanding plus its context. Vijayaraghavan et al. (2020).

2.2 History of Chatbots

Joseph Weizenbaum was the first person to develop a chatbot in 1966 at MIT AI lab, United States. The chatbot was named as ELIZA and it generated output by applying predefined set of rules on the keywords received as input. The second chatbot called Parry was developed in 1972 at Stanford University by Kenneth Colby; a psychiatrist. This Chatbot was intended to simulate a person with paranoid schizophrenia. The next major chatbot was developed in 1995 by Richard Wallace. He called the chatbot ALICE, short for Artificial Linguistic Internet Computer Entity. Though ALICE was awarded the Loebner prize 3 times, it never passed the Turing test; which is a test that is widely used to normally checks if a machine can think intelligently like humans. After ALICE, another major breakthrough happened from 2010 when several AI-based personal assistances were launched by big technology companies. First on the list was Siri by Apple in 2010, followed by Google Assistant for Android in 2012. In 2015, Microsoft followed suit and launched Cortana; a personal productivity assistant on Windows. Another breakthrough came when smart speakers were introduced which made it possible for voice conversation between bots and humans. Two main conversational smart speakers that dominate this space are Amazon’s Alexa and Google Home.

Messaging Platforms and AI Technologies have enabled the exponential growth of Smart Chatbots. Telegram become the first Messaging/Social media platform to allow developers integrate bots into the App in 2015. Facebook followed next and now all popular social media platforms including WhatsApp, Instagram, TikTok, Slack have provided developers with API for bot integration. Chatbots have gained popularity in the recent past and are currently being used to resolve customer queries, process customer orders, perform financial transactions etc. (Aleedy et el., 2019).

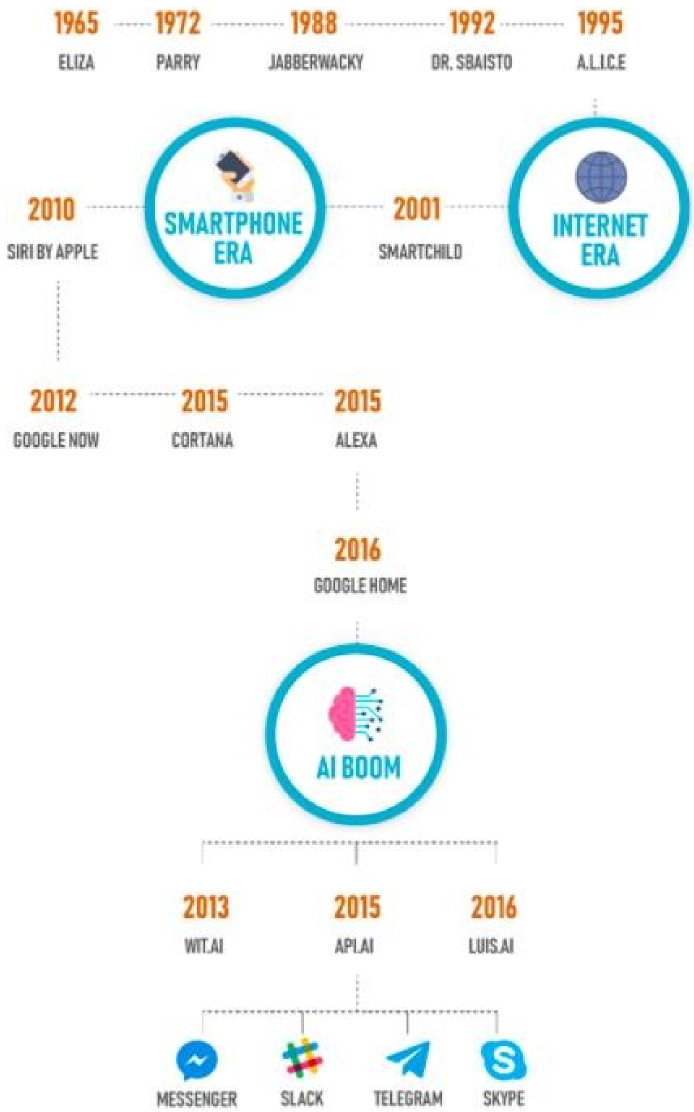

Fig. 2.0 Pictorial representation of the History of Chatbots

2.3 Chatbots Models

A chatbot is a software program that provides a real conversational experience to a human user. Raj (2021) notes that there are two major categories of intelligent chatbot models; Retrieval-based chatbots and Generative-Based chatbots. In **retrieval-based chatbots**, input patterns are predefined, and responses are embedded. Machine Learning or Neural Network models are then used to train the bots on the intents of the user and applicable responses. With this training, the chatbot will then be able to identify new intents and provide appropriate responses. One of the main advantages of retrieval-based chatbots is that they are not prone to grammatical errors since they do not generate any new sentences. The second category of chatbots is the **generative-based chatbots**, also known as self-learning bots. These chatbots are built using advanced NLP and Deep Learning techniques making them very efficient and with the ability to pick and identify user's intent by themselves. Raj (2021) further observed that though generative-based models are not easy to train due to the fact that the models would need time to learn sentence structure by themselves, but once trained they'll perform way better than retrieval models. According to Xu et al (2017), most organization use a hybrid version of both Generative and Retrieval methods which gives the chatbots strong ability to efficiently handle customer interactions in all situations.

2.4 Components of a Chatbot

Keerthana and Fathima (2021), an intelligent chatbot is composed of three key components, that is; User interface or Channel, NLP Engine and Database/external data as shown in the figure 2 below.

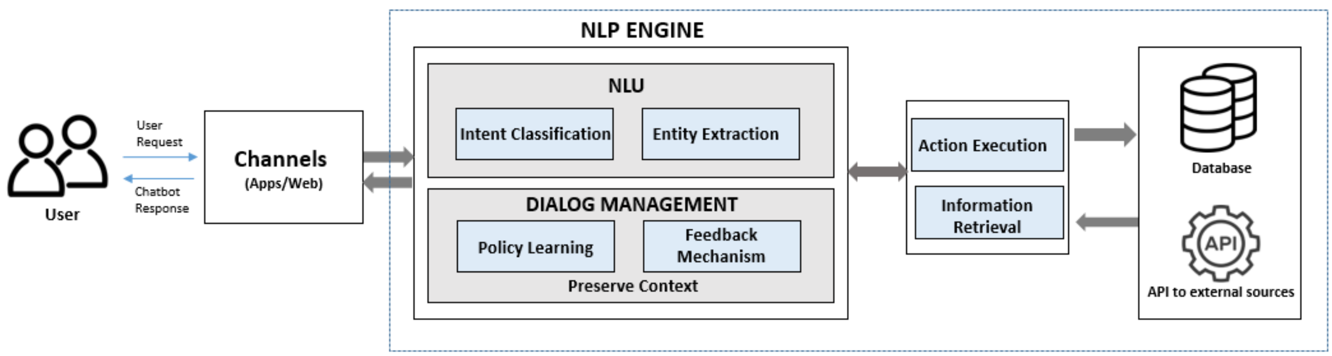

Fig. 2.1: Key components of an AI chatbot

The first component is the UI (User Interface) or Channel which enables a user to interact with the chatbot. A user types or speaks to the chatbot via these channels and gets respond from the chatbot through the same channel. Social Media Platforms and Websites as some of the chatbot access channels.

The second and critical component is the **NLP Engine**. This is the 'brain' of the chatbot. It comprises of two components; the Natural Language Understanding (NLU) and the Dialog Management tool. NLU performs Intent Classification where the intent of the user is identified and meaning extracted. Based on the meaning, the user input is matched to one of the intents that the chatbot supports. NLU also supports Entity Extraction where the key information is extracted from the user query. The Dialog Management (DM) is responsible for determining the context of the conversation. It supports the flow of the dialogue between the user and the chatbot. DM achieves its purpose through Policy Learning and the Feedback Management. Policy Learning entails assessing and choosing the next action for the conversation based on the response that has the highest confidence level. The feedback mechanism is primarily for periodically taking feedback from the users to assess conversational efficiency thus allowing the bot to learn and upgrade itself for future conversations.

The final component of the a chatbot is the Database or API solutions. The APIs enables the chatbots to retrieve responses from external sources like websites, CRM, BI systems and many more (Xu et al., 2017)

2.5 Natural Language Processing

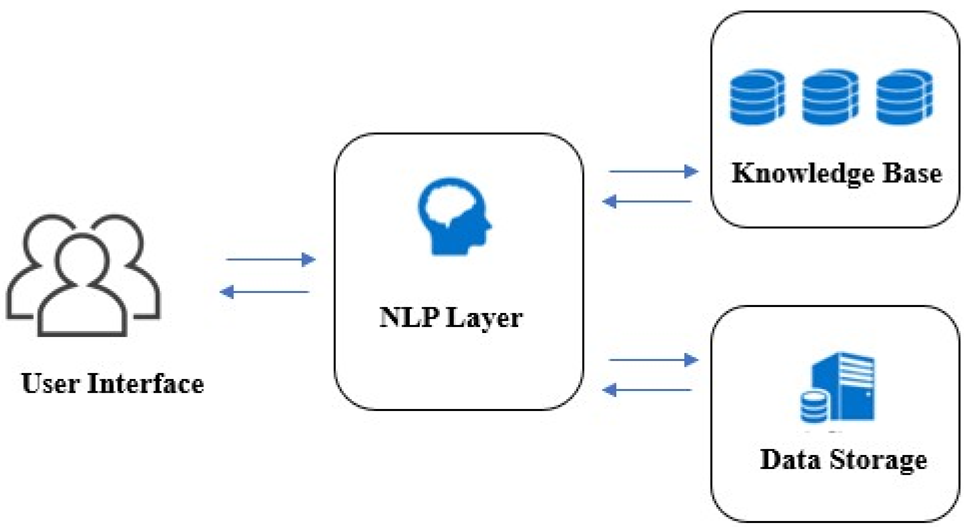

Fig. 2.2: NLP Principles

NLP is a subfield of computer science that analyzes human language in both text and speech. According to Hirschberg and Manning (2015), NLP enables human to machine and machine to human communication. The study further notes that NLP and Text mining are extensively applied in AI chatbots to determine an appropriate answer to customers' query. This has majorly helped reduce over reliance on human agents to support customers. Artificial Intelligence and Natural Language Processing are the critical technologies driving the wide spread adoption of customer service chatbots. The chatbot applications are mainly important when customers need support outside working hours (Xu et al, 2017). Based on Turing test, the conversation between a user and an ideal chatbot should be very natural to the extent that the human user wouldn't be aware that they conversing with a machine. Through machine learning (ML) and big pool of conversational data, a chatbot's algorithm can learn the intricacies of human language. NLP has a wide range of functionalities like summarization of text, vectorization of words, n-gram, Part of Speech tagging and sentiment polarity analysis which helps the chatbot to understand textual data like grammar, sentiment and intent. NLP is thus very important to a chatbot's algorithm because it enables the chatbot to not only understand textual data but also to interpret it and present relevant response.

2.6 Deep Learning Models (Recurrent Neural Network)

Deep learning models have the ability to directly perform classification related tasks on sound, images and textual data. The models are trained with large sets of unlabeled data and neutral networks like RNN will learn and extract features directly from the data set. For sentence representations and text classification, the most appropriate neutral network models are RNN models and specifically, Long Short-Term Memory (LSTM) model due to their ability to keep sentence context (Aalipour et al., 2018). RNN and LSTMs were therefore adopted in this study because studies have shown that they are the best algorithms to process textual data efficiently (Vijayaraghavan et al.,2020). Figure 3 shows the architecture of RNN.

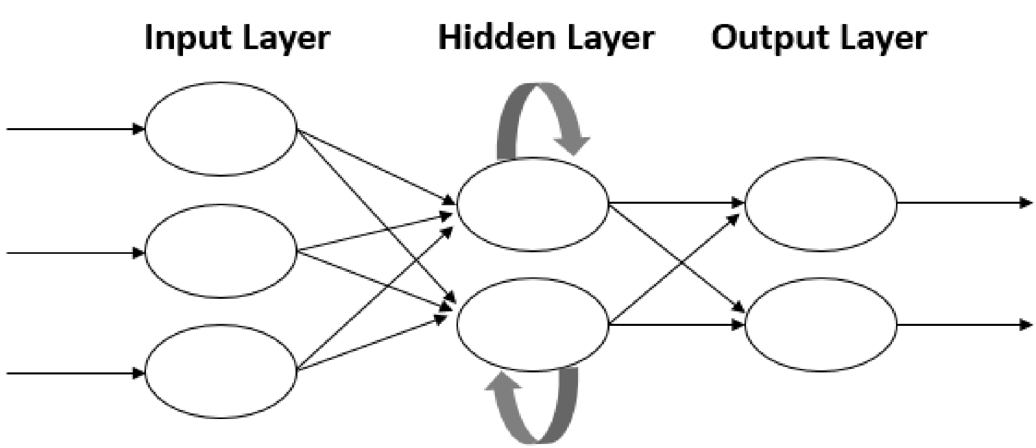

Fig. 2.3: The Architecture of RNN Model

The normal feed-forward neural networks will map one input to one output while Recurrent Neural Network is advantageous because it has the ability to map one input to many outputs, many inputs to many outputs and many inputs to one output. Many to many mappings is used in translation while many to one mapping is used in voice classification. Out study adopted many to many mappings.

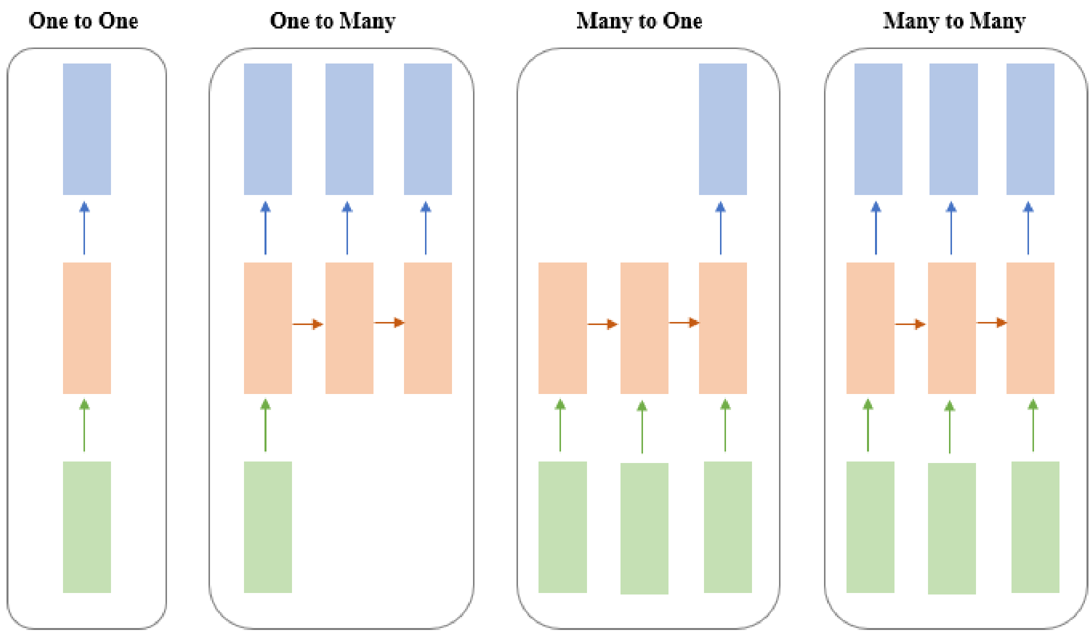

Fig. 2.4: RNN Input to Output mapping

Unlike a normal feed-forward neural networks, information in RNN cycles through a loop. That is, RNN considers both the current the input and the previous input to inform the output. RNN achieves this from an internal memory which helps it to produce an output and loop that output back into the neural network. RNN's ability to consider present input as well as past input helps it to determine what will come next. This memory feature has made RNN outperform other algorithms in textual analysis.

2.7 Chatbot Model with RNN (LSTM)

As observed by Aalipour et al., 2018, the ability of LTSM models to store previous input have made them the preferred models for processing textual data. Richard (2017) noted that RNNs can be used for language modeling by training them to learn the probability distribution from an input of sequence of words. Aleedy et al., 2019 later stated that the LSTM models can generate better translated sentences that it would be difficult to differentiate between the machine vs human responses.

LSTM has been used in developing the chatbot model because it extends the memory of RNN by learning from past crucial experiences that have a long-time lag in between. LSTM units or network are normally used as building blocks for RNN layers thus enabling later memory of inputs over time. LSTM stores information in a memory as a gated cell and it can read, write or delete that information based on importance it assigns to that information. The algorithm learns over time what information is important and what is not. To achieve this, LTSM uses an input gate, forget gate and output gate. Input gate is for letting in new information, forget gate for deleting unimportant information and output gate for letting the new information impact the output at a given timestamp. Sigmoid function is used as the gating function for the 3 gates because it outputs values in the range of 0 or 1. The function can block or allow information to flow through the 3 gates.

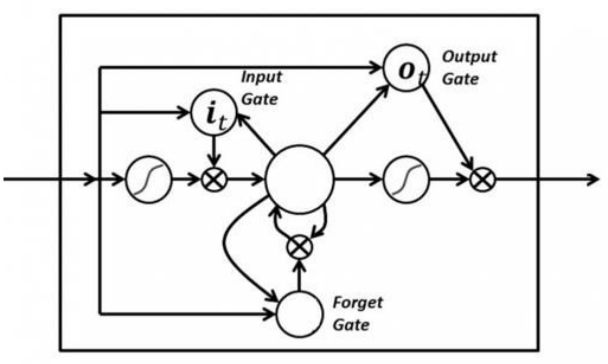

Fig. 2.5: Basic Recurrent Neural Network Architecture with three gates (Input, Forget & Output)

2.8 Waterfall System Development Methodology

Major system development models currently in use are Waterfall, Agile, Lean, Iterative, Prototyping, Spiral and V-model. Waterfall system development methodology was used to develop the university AI chatbot. This approach was adopted because of its orderly sequence of development steps that ensured that the development had proper controls and was completed on time. Also, with waterfall methodology, each phase of the development was processed and completed one at a time and this assisted the researcher with better planning and completion of the project within the stipulated time. The key disadvantage of this methodology though is that it gave little room for use of iteration hence some problems were not discovered until system testing.

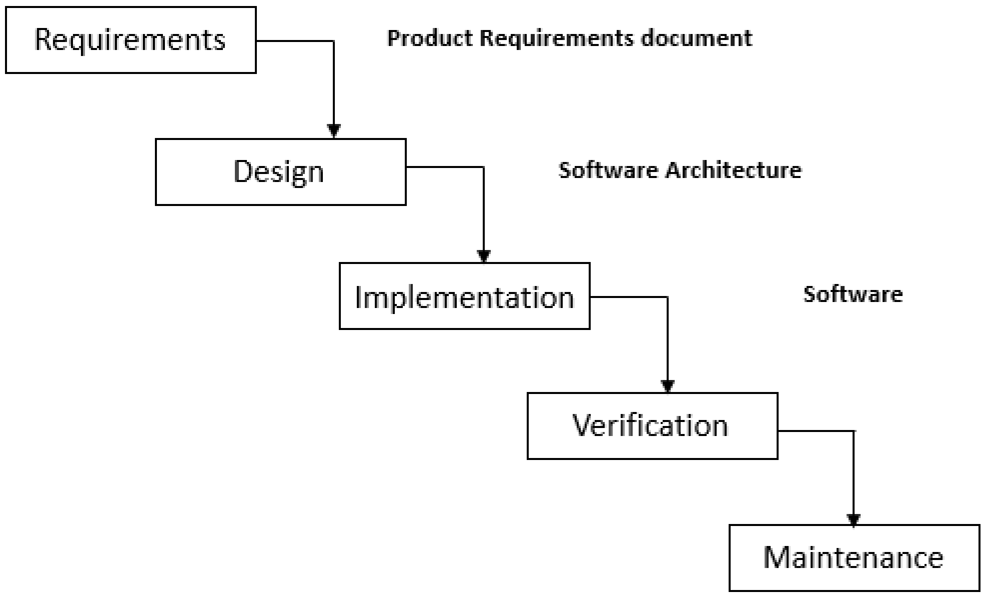

Fig. 2.6: Waterfall System Development Model

2.9 Related Works

Kahiga (2020) states that performance, effort and security of chatbot are the key determinants for the adoption of chatbots in Kenya. His study focuses on technology companies, and specifically Zuku, that has already integrated their chatbot on Telegram social media. The study however fails to mention how local companies can build their own chatbot either from the exiting bot frameworks or from ground-up. Tarus, Gichoya and Muumbo (2015) detail 10 recommendations to successful implement e-learning in the Kenyan universities. The paper is however silent on the infusion of AI to build a robust e-learning platform for learners. Nyongesa, Omieno and Otanga (2020) identified nine distinct factors that influence the adoption of AI chatbot in the telecoms industry. Some of the factors include Perceived benefits, Top management support, Organizational readiness, Customer pressure and Need for automation among others. The study however did not expound on how AI chatbots could be adopted in the educational sector. Piccolo, Roberts, Iosif and Alani (2018) recommended future research on the evaluation of technical effort that can be adopted to build a functional, robust and intelligent chatbot. In a research by Manyu Dhyani and RajivKumar (2020), that was aimed to improve the accuracy of a chatbot and make conversation between a human and a bot very close to real world conversations, they established that building a chatbot using a bidirectional RNN and attention mechanism with TensorFlow resulted in an efficient and accurate chatbot with an outcome leaning rate, Bleu score and Average time per 1000 steps at 0.0001, 30.16 and 4.5 respectively.

2.10 Research Gap

Whilst a lot of studies have been conducted in the field of conversational AI chatbot (Tarus et al., 2015., Piccolo, 2018., Kahiga, 2019., Nyongesa et al, 2020), no study has gone to the level of prototyping a conversational AI chatbots in universities in Kenya. This research proposal will therefore attempt to address the concept of AI-powered chatbots and how universities in Kenya can develop and deploy chatbots to assist in addressing students queries.

2.11 High Level Architecture of the developed chatbot

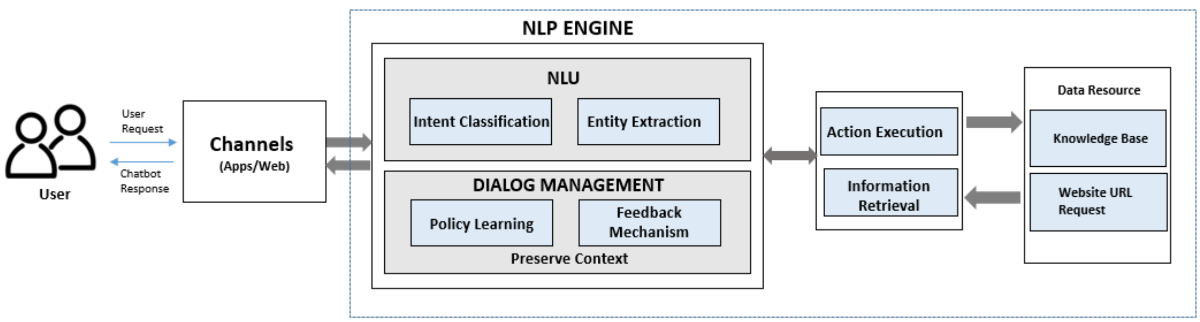

Fig. 2.7: Architecture of a Conversational AI Chatbot

The key components of the chatbot's architecture will be User interface, NLP Engine for text manipulation & Data Source for storing the knowledge base of the frequently asked queries by the students at Computing and Informatics Department, Nairobi University. RNN algorithm will be used to train the model.

2.12 Chapter Summary

This chapter has outlined the history of chatbots from ELIZA in 1966 to Amazon's Alexa in 2015. The chapter also covers research related works plus existing gaps, in the field of AI chatbots in education sector. The chapter closes by highlighting the high-level architecture of developed chatbot system.

CHAPTER THREE: METHODOLOGY

3.1 Introduction

System development methodology is a framework used to structure, plan and control the process of developing an information system (Kyeremeh and Kwadwo, 2019). Hoffer et al (1999) also defines System methodology as a method for describing the activities that is involved in defining, building and implementing a system. Waterfall system development methodology was used to develop the university AI chatbot because each phase of the waterfall methodology is properly planned and time boxed and this enabled the researcher to plan better and thus completed the chatbot development within the stipulated time.

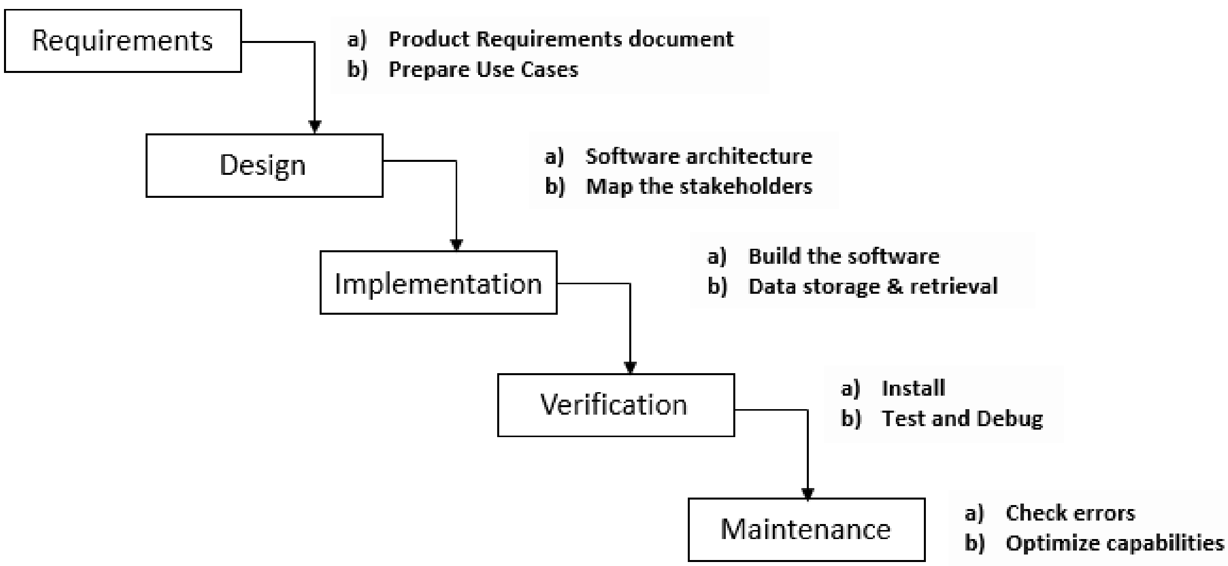

Fig. 3.0: Waterfall Methodology

3.2 Requirement Analysis

Requirements analysis is the process of defining user expectations for a new system that is to be developed or an existing one that is to be modified (Ahituv, Neumann and Zviran, 2002). This process involves identifying and documenting the needs of different stakeholders. Requirements analysis produces a system requirement specification (SRS) document that captures the user requirements. High standard requirements are actionable, measurable, testable, traceable, identifies business opportunity and should be well defined to support system design, Rastogi (2015). For the University AI chatbot, the requirement gathering involved structured interviews with the MSc. students DCI who were identified as the key stakeholders and end-users. Content Analysis of the official university website was also adopted. The researcher interviewed 30 students to understand some of the main issues that normally drive enquires at the department. The researcher also visited the University of Nairobi website to obtained published various Frequently Asked Questions (FAQs). Content analysis was used to extract and analyze the information obtained from the official university website: https://www.uonbi.ac.ke/frequently-asked-questions-faqs and https://academics.uonbi.ac.ke/frequently-asked-questions. In documenting the user requirements, the study adopted three techniques; Use Cases, Data Flow Diagrams (DFD) and Flowcharts. Use Cases helped design the Chatbot from the end user's perspective while DFD were used to graphically represent how the data flows in the chatbot system to enable it perform specified functionalities. Flowcharts show sequential flow and control logic of a related set of activities. Flowcharts are useful for technical and non-technical members. Flowcharts were useful for this study because it gave better visualization and simplification of the user requirements for the AI Chabot.

3.3 System Design

Systems design is the process of defining and developing the various components of a system that will satisfy user requirements. (Kyeremeh, 2019 and Alireza, 2014). According to Kyeremeh and Kwadwo (2019), an architectural design is a high-level plan that helps to describe the structure and behaviour of the system. Ahituv et al. (2002) further notes that an architectural design is the structure and relationship between various modules of system development process. The user requirement document was the key input for the chatbot design and it included an architectural design, Sequence Diagram and Flowchart Diagram for the various components that were identified during the requirement gathering.

3.4 System Development

According to Hoffer et al (2015), system implementation is the process of carrying out the development tasks required to convert the design into a working system. This study used Python 3.6 version, Keras Deep Learning Framework, Jupyter Notebook with Anaconda, JSON for file handling and NLTK libraries to implement the university AI chatbot. The first step of the system implementation involved building the dataset by defining different intents, patterns and responses for the chatbot based on the data obtained during data collection. The second step entailed data preprocessing where the data was cleaned, tokenized and vectored ready to be trained. The processed data was then split into 2 sets; training and test datasets. The third step involved creating the LTSM layers, training the model and saving the model. Once the model was trained and saved, it was tested for accuracy level. Finally, the trained model was deployed, and its results analyzed

3.5 System Testing and Deployment

The study employed three types of testing the AI chatbot. These were, Data-set testing, System Testing and User Acceptance Testing (UAT). Under Data-Set Testing, the chatbot data (Knowledge base), was separated into training and testing set and each set was then randomly tested to confirm that the results were similar. Under System Test, BLEU measurement was used to evaluate the chatbot's ability to respond to students queries accurately and efficiently. User Acceptance Tests was carried buy the target students after the Chatbot was deployed in production. According to Rastogi (2015), the User Acceptance Testing (UAT or Beta-Testing) is the final testing and it's normally done by the intended users in the real world. (Rastogi, 2015).

3.6 System Evaluation

Software evaluation is an assessment that seeks to determine if a software or system is the best possible fit for the needs of a given client (Hoffer et al, 2015). The chatbot was evaluated based on the study's objective of Efficiency in responding to students' queries. Chatbot Efficiency measures the accuracy and speed of the answer to the user's query. According to Cesas el., (2020), efficiency for chatbots is related to their performance towards achieving their goal which is to determine whether or not, the task the chatbot is made for is achieved.

This study adopted automated measurements and human judgment to test the University Chatbot. For the human judgment, the chatbot efficiency was measured by the accuracy and speed at which it responded to students' queries. 20 DCI students were granted access to the deployed Chatbot and asked to interact with it for a period of 3 weeks. They were then given questionnaire to evaluate the Efficiency of the chatbots. The following metrics were adopted for the efficiency metrics; Quickness, Information Retrieval, Relevancy, Interaction, Satisfaction, User Recommendation.

For the automated measurements, the study employed the BLEU score measurement. BLEU was developed for translation but can also be used for evaluating text generated for different NLP tasks. The score ranges between 0.0 to 1.0 with 0 being a perfect mismatch and 1 being a perfect match (Kishore Papineni, et al., 2002). *Corpus_bleu ()* function from Python's NLTK library was used to calculate the BLEU score for the generated sentences.

3.7 Summary of Tools and Methods Adopted

Python

Python is an interpreted high-level and general-purpose language that enables programmers write clear, logical code for small and large-scale projects. Most deep learning models like RNN are well built on Python

Natural Language Processing

Artificial Intelligence (AI) and Natural Language Processing (NLP) are the critical technologies driving the wide spread adoption of customer service chatbots.

Stemming

Stemming algorithm is used to find the stem (root) of each concept (CT) for further analysis. Once stop words are identified and removed, each inflected word is reduced back to its stem. We have used a Stemming technique that abbreviates word by removing affixes and suffixes.

Bag of Words

At the core of it, Machine Learning and Neural Networks algorithms only take numerical input. BOW helped the researcher to convert string input into numerical representation that was then fed into the models algorithm. BoW uses binary 0 and 1; where 1 means that a particular word exists in the sentence while 0 means that the word is not present.

3.8 Ethical Considerations

There were no safety threats of this project to the respondents and to the users. Respondents consent were sought before conducting the interviews and their participation was purely voluntary.

3.9 Chapter Summary

This chapter has covered the methodology that the study followed in developing the University AI chatbot. Waterfall model was adopted for the end to end delivery of the system. System Requirements were gathered through unstructured interviews and the content/document analysis of the university website materials. Python and Keras tools were used to develop the chatbot. Chatbot evaluation was done through BLEU score and human evaluation

CHAPTER 4: SYSTEM ANALYSIS AND DESIGN

4.1 Introduction

This chapter outlines requirements analysis and system design approach that the researcher employed in developing the university chatbot.

4.2 Requirement Analysis

Following the in-depth interviews with the DCI MSc students and Content Analysis of the University website, students' key areas of enquires at the university were identified and categorized into Admission Process, Registration Process, Students Fees, Student Loans Status, Courses Offered, Class Timetable, Examinations, Semester Start & End dates and Lecturers' contacts. Students also expressed that a chatbot would only be good if the user interface is friendly, if it's easily accessible and provides correct information faster.

As result, the researcher grouped the requirements into Functional and Non-functional requirements for ease in system design.

Functional Requirements	**Non-Functional Requirements**
1. User interacts with the UniBot via web UI 2. User asks any Questions on the following Topics: Admission Process, Registration Process, Students Fees, Student Loans Status, Courses Offered, Class Timetable, Examinations, Semester Start & End dates and Lecturers' contacts 3. User responds fast and with correct information	1. Immediate response to the user 2. Storing of intent and patterns in a json file 3. Apply LSTM layers to train the Model

Table 4.1: Functional and Non-functional Requirements

4.3 System Analysis and Design

4.3.1 Use Case Diagrams

The students and the university administer were identified as the key actors in the system. The student task involved typing in their query through the chatbot interface. While the administrator role entailed; adding, viewing, updating and deleting the FAQs. The administrator would also trigger re-training of the model once FAQs were updated.

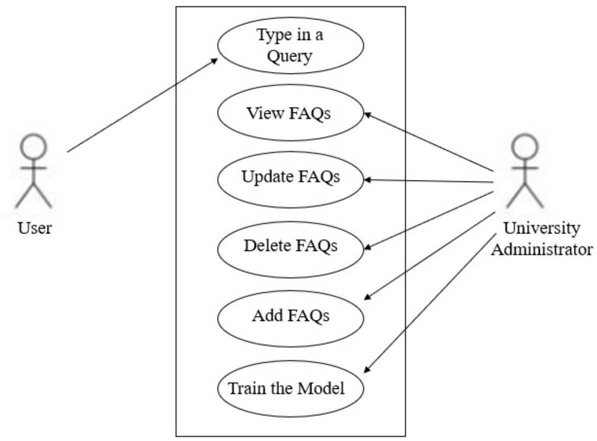

Fig. 4.1: Use Case Diagram of User and Admin roles

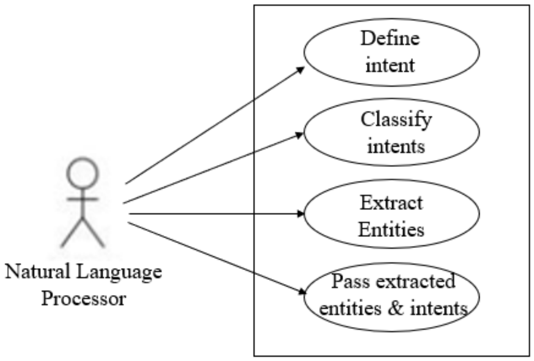

Fig. 4.2: Use Case Diagram of NLP roles

4.3.2 Data Flow Diagram (DFD

The below logical DFD was used to represent the flow of data for the chatbot in a graphical manner. 1000 FAQs formed part of the chatbot Knowledge Base (KB). Whenever the user asks a queries the chatbot, the process query task would be triggered to run and fetch keyword from the user input which would then be used to return relevant answer to the user input.

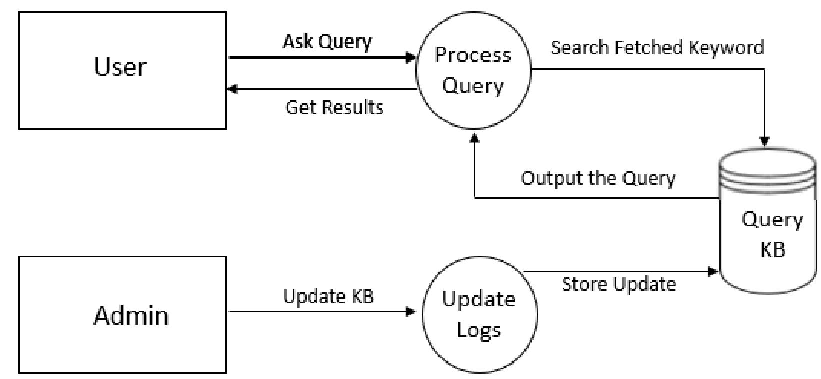

Fig. 4.3: Data Flow Diagram of the Chatbot System

4.3.3 Architectural Design

The University AI Chatbot architecture captured the 3 major components of the bot i.e. The access channels (User Interface), the NLP Engine and the Data Source (Knowledge Base). Figure 4.4 shows the architectural design, Figure 4.5 the Sequence Diagram and Figure 4.6 Flowchart for the University AI chatbot.

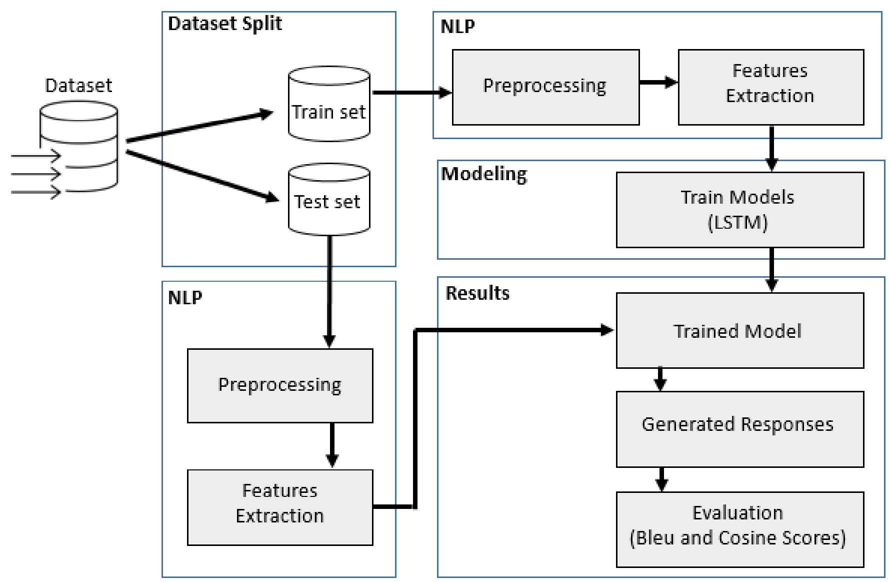

Fig. 4.4: Architectural design of the UniBot

4.3.4 Sequence Diagram

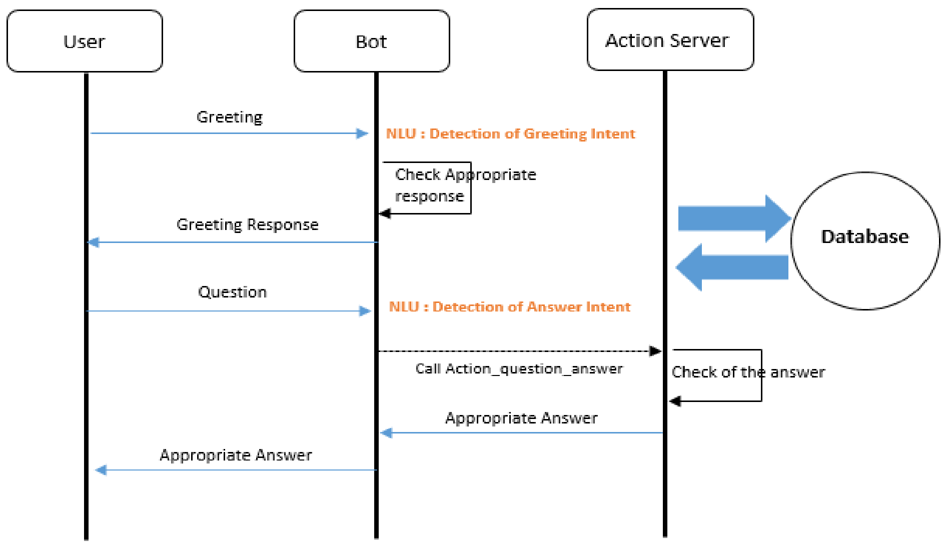

Fig. 4.5: Sequence Diagram for the UniBot

4.3.5 Flowchart Diagram

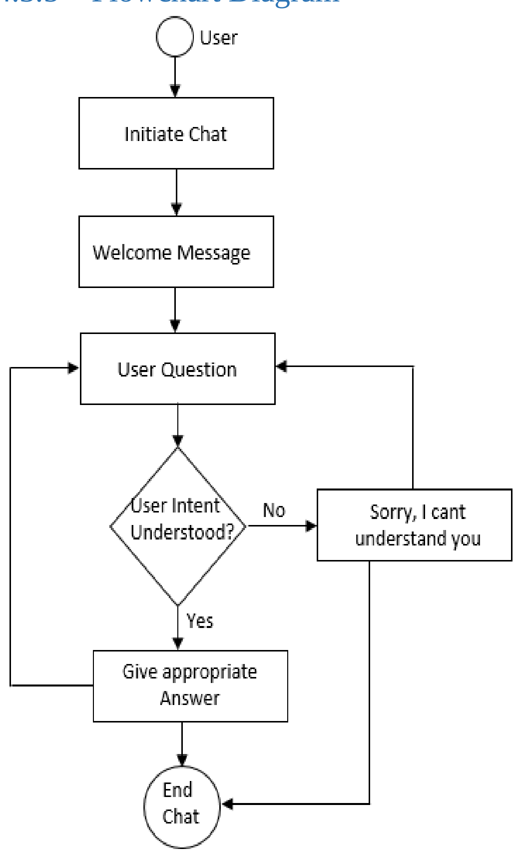

Fig. 4.6 Flowchart Diagram for the UniBot

4.4 Design Algorithm

To build the chatbot algorithm, the knowledge base data was preprocessed through cleaning, tokenizing and vectorizing the dataset. LSTM layers were then created, and the model trained and saved. All user inputs were equally preprocessed then passed through the model for response generation.

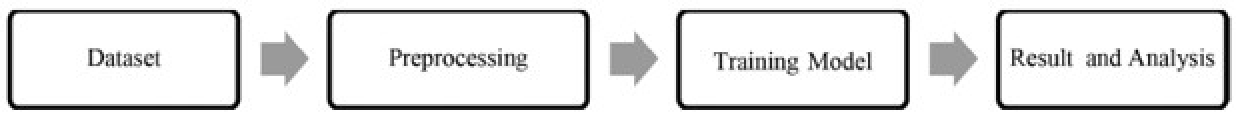

Fig. 4.7 Process Design

4.5 User Interface Design

The channel for interaction with the Chabot was a web interface.

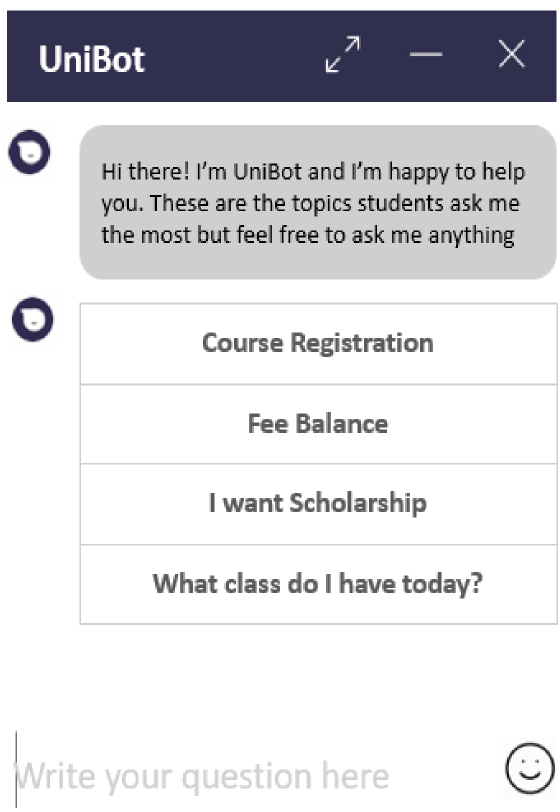

Fig. 4.8 User Interface

4.6 Data Storage

The data was stored in a JSON file. The intents were grouped into various tags. Anticipated user per topic inputs were categorized under 'patterns' and related responses given under the responses category. Figure 4.9 below shows a snippet of the data storage.

```
{"intents": [
        {"tag": "greeting",
         "patterns": ["Hi there", "How are you", "Is anyone there?","Hey","Hola", "Hello", "Good day","Good morning", "Good evening", "Good Afternoon"],
         "responses": ["Hello, thanks for asking", "Good to see you again", "Hi there, how can I help?"],
         "context": [""]
        },
        {"tag": "goodbye",
         "patterns": ["Bye", "See you later", "Goodbye", "Nice chatting to you, bye", "Till next time"],
         "responses": ["See you!", "Have a nice day", "Bye! Come back again soon.","Kwaheri, come back again soon"],
         "context": [""]
        },
        {"tag": "thanks",
         "patterns": ["Thanks", "Thank you", "That's helpful", "Awesome, thanks", "Thanks for helping me"],
         "responses": ["Happy to help!", "Any time!", "My pleasure"],
         "context": [""]
        },
        {"tag": "noanswer",
         "patterns": [],
         "responses": ["Sorry, can't understand you", "Please give me more info", "Not sure I understand"],
         "context": [""]
        },
```

Fig. 4.9 FAQ data in a JSON file

4.7 Design Considerations

The following design considerations were adopted; Friendly User Interface for students, Prior Training of Model to match user questions with relevant answers, Supporting only English dialogue.

CHAPTER 5: SYSTEM IMPLEMENTATION, TESTING & EVALUATION

5.1 Introduction

This chapter describes the results that were obtained post the implementation, testing and evaluation as des cribbed in the methodology section.

5.2 Hardware Implementation

This model was trained on Windows PC and will run on any device with internet access.

5.3 Software Implementation

5.3.1 Knowledge Base

Frequently asked Questions from the Department of Computing and Informatics was obtained from the University website and uploaded as a dictionary in a JSON file which become the chatbot's knowledge base. The JSON file was stored in a variable called *data* as shown below.

```
with open("uon_faqs.json") as file:
    data = json.load(file)
```

Fig. 5.1 Storing the json data in *data* variable

5.3.2 Data Extraction

All the patterns (questions) and their corresponding class/tag were extracted and each pattern turned into a list of words using *nltk.word_tokenizer* and then added into the a list called docs_x. Its corresponding tag was put into the docs_y list.

```
words = [] # blank list
labels = []
docs_x = []
docs_y = []

for intent in data ["intents"]:
    for pattern in intent ["patterns"]:
        wrds = nltk.word_tokenize (pattern) #stemming : Takes
        words.extend(wrds)
        docs_x.append (wrds)
        docs_y.append (intent ["tag"])

        if intent ["tag"] not in labels:
            labels.append(intent["tag"])
```

Fig. 5.2: Extracting data from and storing it as x and y list.

5.3.3 Data Preprocessing

Word Stemming process was used to minimize the model's vocabulary. The strings were then converted into bag of words (numbers). Lastly, the training data and output was converted to numpy arrays

```
training = []
output = []

out_empty = [0 for _ in range (len(labels))]

#For the tags
for x, doc in enumerate (docs_x):
    bag = []

    wrds = [stemmer.stem (w) for w in doc]

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

    output_row = out_empty[:]
    output_row [labels.index(docs_y[x])] = 1

    training.append (bag)
    output.append (output_row)

#Takes input, change into an array so we can fit into our model
training = numpy.array (training)
output = numpy.array(output)
```

Fig. 5.3 Data Preprocessing

5.3.4 Model Development

After preprocessing the data, the model was then created and trained. Standard feed-forward neutral network with 2 hidden layers was used. The goal of the Neutral Network was to look at a bag of words and determine the probability of the tag they belonged to.

```
tensorflow.reset_default_graph()

net = tflearn.input_data(shape=[None, len(training[0])])
net = tflearn.fully_connected(net, 8)
net = tflearn.fully_connected(net, 8)
net = tflearn.fully_connected(net, len(output[0]), activation="softmax")
net = tflearn.regression(net)

model = tflearn.DNN(net)
```

Fig 5.4: The architecture of LSTM Neural Network model adopted

5.3.5 Training and Saving the Model

When the model had been set up, it was then trained on the knowledge base data. This was done by fitting the data to the model. The epochs set was to 1000. i.e. The model saw the same information 1000 times while training. The model upon training completion was saved to the file **model.tflearn**

```
model.fit(training, output, n_epoch=1000, batch_size=8, show_metric=True)
model.save("model.tflearn")
```

Fig 5.5 Training and Saving the Model

5.3.6 Making Predictions

```
def bag_of_words(s, words):
    bag = [0 for _ in range(len(words))]

    s_words = nltk.word_tokenize(s)
    s_words = [stemmer.stem(word.lower()) for word in s_words]

    for se in s_words:
        for i, w in enumerate(words):
            if w == se:
                bag[i] = 1

    return numpy.array(bag)


def chat():
    print("Start talking with the bot (type quit to stop)!")
    while True:
        inp = input("You: ")
        if inp.lower() == "quit":
            break

        results = model.predict([bag_of_words(inp, words)])
        results_index = numpy.argmax(results)
        tag = labels[results_index]

        for tg in data["intents"]:
            if tg['tag'] == tag:
                responses = tg['responses']

        print(random.choice(responses))

chat()
```

Fig. 5.6 Making Prediction from User's input

Bag_of_words function was used to transform user input (string) to a bag of words using the list of the created words while lets_chat () function was used to get the model prediction and the correct response from the *data* variable.

5.4 System Testing

The models achieved an average of 97% accuracy in training.

```
Training Step: 6999  | total loss: 0.50227 | time: 0.043s
| Adam | epoch: 1000 | loss: 0.50227 - acc: 0.9777 -- iter: 48/51
Training Step: 7000  | total loss: 0.45254 | time: 0.050s
| Adam | epoch: 1000 | loss: 0.45254 - acc: 0.9799 -- iter: 51/51
--
INFO:tensorflow:D:\Msc\Final Project\Chatbot\model.tflearn is not in all_model_checkpoint_paths. Manually adding it.
```

Fig. 5.7 Model Training

The Chatbot performed well on the topics that it had been trained on. The evidence for increased efficiency was embodied in BLEU score. The chatbot scored highly (BLEU average score of 0.75) for questions that it had been previously exposed to. For new questions that touched on topics that the chatbot had not been trained on, it performed poorly (BLEU score of 0.20). The more the data that we exposed to the chatbot, the better were the results (BLEU Mean Score = 0.75)

```
chat()
```

```
Start talking with the University of Nairobi bot (type quit to stop)!

Scope of the bot is :
 1. Admission process
 2. Online Registration
 3. Check university Fees
 4. Key University contacts
 5. Helb Issues
 6. Student ID Issues

You: Hi

Unibot: Good to see you again

You:
```

Fig. 5.8 Chatbot Test UI

User Input	Chatbot Response (Generated Sentence)	BLEU Score
1. Jambo?	Hi there, how can I help?	0.90
2. Nice chatting to you, bye	Kwaheri, come back again soon	0.85
3. Blank input	Sorry, can't understand you	0.70
4. How you can be helpful?	I can guide you through the Admission Process, Registration Process, Students Fees, Student Loans Status, Courses Offered, Class Timetable, Examinations, Semester Start & End dates and, Share Key University contacts	0.90

5. How can I get my registration number?	Access the students' portal on https://smis.uonbi.ac.ke/. Then Scroll to the bottom of the page to the SMIS Allocated Registration Number section. In the box provided, Enter Application Reference No as it appears in your Letter of Offer to get your allocated Student Registration No. Click on Get Allocated Students number	0.80
6. Give me the key contacts for assistance	Academic Registrar 020 4913199/3027/3196/3067/3065 ; Graduate School 0204913128/3129/8030/8032; Customer Experience Centre 0204910000 or 0204914112, customercare@uonbi.ac.ke, SERVICE EMAIL ADDRESS: Undergraduate Admissions: admissions@uonbi.ac.ke , Graduate Admissions: gs@uonbi.ac.ke	0.75
7. How much is the university fee for Master's course?	Total Fees: Kshs. 552,000. For breakdown visit : https://www.uonbi.ac.ke/sites/default/files/postgraduate_booklet.pd	0.65
8. What's the application process for admission	To make an application for the courses offered at the University please visit application.uonbi.ac.ke register and follow the steps to make your application.	0.90
9. How can I replace a lost student ID	Apply online via SMIS portal. You'll require Police abstract, Clearance from the library, Letter from Dean/Faculty and Fee of 500/= for undergraduate students or Fee of 1000/= for post graduate students	0.85
10. What is my home address?	University Address is: Postal Address: P. O. Box 30197-01000, Nairobi, Kenya. Email: fo@uonbi.ac.ke Phone: (+254) 020-4910000 Fax: +254-020-2243660 Mobile: +254-0725077039	0.20

Table 5.1: Chatbot BLEU Results

CHAPTER 6: RESULTS AND DISCUSSIONS

6.1 Introduction

The UniBot was designed and developed as AI agent for handling frequently asked queries (FAQs) by MSc students at the Department of Computing and Informatics. It was a retrieval-based chatbot that matched text input with appropriate predefined responses and was thus able to help target students get answers to their queries via a chat interface. Students interacted with the UniBot via their smart devices (phones and PC) where they were able to ask a series of questions through text messages and/or keywords, and obtained accurate responses very fast. The chatbot's knowledge base covered Admission Process, Registration Process, Students Fees, Student Loans Status, Course Offered, Class Timetable, Examinations, Semester Start & End dates and Lecturers' contacts.

6.2 Students' Perceived Efficiency of the Chatbot

The chatbot was introduced to twenty students at the University of Nairobi, Department of Computing and Informatics. The students were encouraged to interact with the chatbot for a duration of three weeks and ask any question around Admission Process, Registration Process, Students Fees, Student Loans Status, Course Offered, Class Timetable, Examinations, Semester Start & End dates and Lecturers' contacts. The students interacted with the chatbot via their PCs and were then given an effectiveness questionnaire (See Table 6.3) at the end of the three weeks to establish their perceptions of the chatbot effectiveness. Student perceptions on the chatbot effectiveness and efficiency were summarized in Table 6.2.

Mean and Standard deviation of students' perceived Chatbot Efficiency

Students' Perceived Efficiency of the UniBot	Mean (ū)	Standard Deviation (SD)	Interpretation
Quickness	**4.6**	**0.55**	**Very High**
1. The Chatbot provides instant answers	*4.6*	*0.55*	Very High
2. The Chatbot shorten the time for getting responses from the university	*4.6*	*0.55*	Very High
Providing Information	**4.3**	**0.50**	**High**
3. The Chatbot responds with correct information	*4.2*	*0.45*	High

4.The Chatbot provides information	*4.4*	*0.55*	High
Relevancy	**4.6**	**0.50**	**Very High**
5. The Chatbot provides relevant information	*4.4*	*0.55*	High
6. Wrong information entered is appropriately answered	*4.8*	*0.45*	Very High
Interaction	**4.4**	**0.72**	**High**
7. The Chatbot interacts through textual chats	*4.4*	*0.55*	High
8. The Chatbot responds with good English	*4.4*	*0.89*	High
Satisfaction	**3.5**	**0.55**	**High**
9. I am satisfied with the information provided by Chatbot	*3.6*	*0.55*	High
10. The Chatbot is easy to use	*3.4*	*0.55*	High
Recommendation	**3.2**	**0.75**	**Low**
11. I'll use this Chatbot again	*3.6*	*0.55*	High
12. I can recommend the chatbot to someone else	*2.8*	*0.95*	Low

Table 6.2: Evaluation Results

Results in Table 6.2 shows that the students perceived the chatbot to be highly efficient as evidenced by the overall mean score (ῡ) of 4.10 (out of 5) and Standard Deviation SD of 0.59). The students' perception of the chatbot efficiency was measured through three key metrics; first was the ability of the chatbot to quickly provide relevant responses, secondly was on how intuitively they interacted with the chatbot's UI and lastly the study measured how satisfied the students were with the UniBot and if they could recommend it to fellow students. The specific results were as follows; Quickness and Relevancy scored **very high levels** (ῡ = 4.6, S.D. = 0.55; ῡ = 4.6, S.D. = 0.50 respectively). Perception of the chatbot in terms of Providing Right Information, Interaction and Satisfaction was also scored **high levels** (ῡ = 4.3, S.D. = 0.50; ῡ = 4.4, S.D. = 0.72 and ῡ = 3.5, S.D. = 0.55 respectively). Students however provided feedback that the User Interface of the chatbot was not friendly and as such said they wouldn't recommend it to non-IT students. This was reflected with a low score of ῡ = 3.2, S.D. = 0.75. Despite of this, 75% of the respondents were happy with the chatbot and indicated that the technology was easy to use, provided correct answers, was innovative and fun for making enquiries. They could seek and obtain instant answers relating to the DCI without having to wait for days for human agent to answer simple clarifying questions like semesters fees, semester start dates etc. 50% of the students also observed that the chatbot was not able to provide relevant links or direct them to a human agent, in cases where the Unibot did not have answers to some questions. The students also noted that the Chatbot was sensitive to typos and only provided responses to correctly typed questions.

6.3 Effects of the Chatbot on Students' Query Support

Currently, University of Nairobi students have to call the University or write an email to admissions@uonbi.ac.ke or any query, regardless of how repetitive the enquiry is. This process is not real time and can take hours depending on how busy the administrator is. With the developed chatbot, the target students obtained answers to their questions on a real time basis with a calculated response time of under 10 milliseconds. The chatbot only refer students to the human agent (department administrator) for questions that the chatbot had not been trained on. The main findings indicated that a chatbot as an online channel for providing answers to frequently asked questions, was efficient and effective according to students' response after interacting with the University chatbot. They gave an overall efficiency score of 82%, with standard deviation of 0.59. The findings of this study were in concurrence with Winkler and Söllner (2018) conclusion that chatbots, when used as personalized online support tool, had positive effect on successful learning and learner's satisfaction. It was observed that the more the chatbot model learned from the new FAQs and different keywords, the better it performed. This was in line with Knill et al. (2004) who observed that chatbots played a big role in helping teachers to provide better personalized support when it learns from students' previous questions and feedback. Also, the findings can help universities better integrate chatbot technology in their websites and social media handles to support particularly new students who may not know where to direct their questions. This study further provides guidelines of using chatbot technology as a powerful digital tool for supporting students with knowledge necessary for fast and easy interaction with the university's knowledge base hence enhancing their smooth stay at the university.

CHAPTER 7: CONCLUSIONS AND RECOMMENDATIONS

In conclusion, the study has established that chatbot technology can be used as virtual assistants to offer instant responses to frequently asked questions with high efficiency rate (Mean Score of 4.10/5.00 and Standard Deviation of 0.59). This can make more easy and comfortable for students to ask and obtain immediate answers to their questions (Cameron et al., 2017) thus freeing up university (staff/admins) from handling many repetitive questions and focus their time on other pressing tasks. The AI chatbot also gives students a friendly user interface for solving queries and this could be accessed from anywhere.

The UniBot's architecture has integrated deep learning model to offer students simple human and computer interaction using natural language. The chatbot provides fast and efficient answers to students queries and relevant links where the students can get detailed information. The Chatbot knowledge base keeps information about frequent questions, their answers, some keywords and logs for improving the bot.

The chatbot developed in this study had some limitation in the sense that it was not able to address texts or keywords that it hadn't been previously trained on. The study therefore recommended that a more robust chatbot with the ability to provide correct and relevant answers even on previously unseen texts should be considered for future research. Voice-based queries can also be incorporated where students can give voice input and get text output from the bot.

REFERENCES

1. Adam, M., Wessel, M. & Benlian, A. (2021). AI-based chatbots in customer service and their effects on user compliance. *Electron Markets 31, 427–445 (2021).* https://doi.org/10.1007/s12525-020-00414-7. last accessed 2021/08/15

2. Aleedy, Moneerh & Shaiba, Hadil & Bezbradica, Marija. (2019). Generating and Analyzing Chatbot Responses using Natural Language Processing. *International Journal of Advanced Computer Science and Applications.* https://doi.org/10.10.14569/IJACSA.2019.0100910 last accessed 2022/01/25

3. Alharthi, A.D., Spichkova, M. & Hamilton, M. (2019). *Sustainability requirements for eLearning systems: a systematic literature review and analysis*. https://doi.org/10.1007/s00766-018-0299-9 last accessed 2022/03/20

4. Alireza, S.N, (2014). Fundamentals of System Analysis & Design. Retrieved from https://www.oreilly.com/library/view/systems-analysis-and/9781118037423/05_chapter001.html

5. Charlton, G. (2013). *Consumers prefer live chat for customer service: stats.* Retrieved from https://econsultancy.com/consumers-prefer-live-chat-for-customer-service-stats/. last accessed 2022/04/01

6. Csaky, Richard. (2017). Deep Learning Based Chatbot Models. DOI: 10.13140/RG.2.2.21857.40801.

7. Gunawardhana, L.K.P.D. (2019). Process of Requirement Analysis Link to Software Development. Journal of Software Engineering and Applications, 12, 406-422.

8. Hoffer, Jeffrey A., George, Joey F., and Valacich, Joseph S. (1999). *Modern Systems Analysis & Design, 2nd ed.* Addison Wesley Longman, Inc.

9. IBM. (2019). AI Chatbot - IBM Watson Assistant. https://www.ibm.com/cloud/watson-assistant/client-stories last accessed 2021/08/15

10. Kahiga A.K. (2019). A model for adoption of chatbots in Kenya: A case study of Zuku Telegram Bot. Retrieved from http://erepository.uonbi.ac.ke/bitstream/handle/11295/107169/Kahiga

11. Kyeremeh, K. (2019). Overview of System Development Life Cycle Models. DOI http://dx.doi.org/10.13140/RG.2.2.27713.71521 last accessed 2021/11/03

12. Manyu Dhyani & RajivKumar (2020). An intelligent Chatbot using deep learning with Bidirectional RNN and attention model. Retrieved from https://doi.org/10.1016/j.matpr.2020.05.450. Last accessed May 2022

13. Molnár, G. and Z. Szüts. (2018). The Role of Chatbots in Formal Education. IEEE 16th International Symposium on Intelligent Systems and Informatics. Subotica, Serbia.

14. Niv Ahituv, Seev Neumann & Moshe Zviran (2002). A System Development Methodology for ERP Systems, Journal of Computer Information Systems, 42:3, 56-67, DOI: 10.1080/08874417.2002.11647504

15. Nyongesa, Geoffrey & Omieno, Kelvin & Otanga, Daniel. (2020). Artificial Intelligence Chatbot Adoption Framework for Real-Time Customer Care Support in Kenya. *International Journal of Scientific Research in Computer Science, Engineering and Information Technology*. DOI: 100-117. 10.32628/CSEIT20667

16. Papineni, K et al., (2002, Jul 311-318). *BLEU: a Method for Automatic Evaluation of Machine Translation.* ACL 2002, Philadelphia, USA.

17. Raj V.S., (2021). Seq2Seq with Attention mechanism for chatbots. *University of Calgary*. Retrieved from https://arxiv.org/abs/2006.02767 last accessed 2021/10/21

18. Rajamalli, KR. Fathima, G. (2021). ACEbot – A University Chatbot. *IRJET 2021, 08-01*. Retrieved from www.irjet.net last accessed 2022/01/11

19. Rastogi, V. (2015). Software Development Life Cycle Models Comparison, Consequences. Retrieved from http://citeseerx.ist.psu.edu/viewdoc/summary?doi=10.1.1.667.9896 last accessed 2022/02/19

20. Sandu, N and Gide E. (2019). *Adoption of AI-Chatbots to Enhance Student Learning Experience in Higher Education in India.* ITHET 2019, Magdeburg, Germany. https://doi.org/10.1109/ITHET46829.2019.8937382 last accessed 2021/10/21

21. Tarus, John & Gichoya, Prof & Muumbo, Alex. (2015). Challenges of Implementing E-Learning in Kenya: A Case of Kenyan Public Universities. Retrieved from http://www.irrodl.org/index.php/irrodl/index. 16. 10.19173/irrodl.v16i1.1816 last accessed 2021/10/22

22. United Nations Educational, Scientific and Cultural Organization. (2020a). *COVID-19 impact on education data.* COVID-19 education disruption and response. Paris, France

23. Vijayaraghavan V., Jack Brian Cooper, Rian Leevinson J. Algorithm Inspection for Chatbot Performance Evaluation. Retrieved from http://www.sciencedirect.com/ last accessed February 2022

24. Yang, S., & Evans, C. (2019). Opportunities and Challenges in Using AI Chatbots in Higher Education. *ICEEL 2019, 79-83.* https://doi.org/10.1145/3371647.3371659 last accessed 2021/10/25

APPENDIX

Sample Questionnaire for Evaluating the Efficiency of the University Chatbot

Assessment of the Efficiency of the UniBot in answering student's questions in the following topics; Admission Process, Registration Process, Students Fees, Student Loans Status, Course Offered, Class Timetable, Examinations, Semester Start & End dates and Lecturers' contacts.

Instruction: Please answer the following questions. Use the order below.

1. Strongly Disagree
2. Disagree
3. Neutral
4. Agree
5. Strongly Agree

Students' Perceived Efficiency of the Chatbot	**1**	**2**	**3**	**4**	**5**
Quickness					
1. The Chatbot provides instant responses					
2. The Chatbot shorten the time for getting responses from the university					
Providing Information					
3. The Chatbot responds with correct information					
4.The Chatbot provides information					
Relevancy					
5. The Chatbot provides relevant information					
6. Wrong information entered is appropriately answered					
Interaction					
7. The Chatbot interacts through textual chats					
8. The Chatbot responds with good English					
Satisfaction					
9. I am satisfied with the information provided by Chatbot					
10. The Chatbot is easy to use					
Recommendation					
11. I'll use this Chatbot again					
12. I can recommend the chatbot to someone else					

Table 6.3: Sample Evaluation Questionnaire