Deep Learning-based Virtual Assistant in Sinhala Language - Phase II

An undergraduate project proposal submitted to the

Department of Electrical and Information Engineering
Faculty of Engineering
University of Ruhuna
Sri Lanka

in partial fulfillment of the requirements for the

Degree of Bachelor of Science in Engineering

by

Bandara H.M.P - EG/2018/3286
Kalyanarathne W.M.U.W - EG/2018/3357
Ransinghe K.K.P.M - EG/2018/3432

.....	.....
Dr. Kushan Sudheera	Dr. J.C.S. Kadupitiya
(Supervisor)	(External)

.....
Mr. W.G.C.A Sankalpa
(External)

Abstract

Virtual assistants are computer programs or systems that are designed to assist users with a variety of tasks and to provide information and support. Deep learning-based virtual assistants employ deep learning models to improve the accuracy and efficacy of activities such as language comprehension, image and video recognition, and decision-making. These virtual assistants have the ability to improve the accuracy and capabilities of a wide range of jobs, as well as to improve personalization, efficiency, and user experience. Our goal is to modify and improve the virtual assistant that approaches interactivity between the user and the machine utilizing Sinhala language using Deep Learning and Natural Language Processing. Within the training domain, the proposed system may accept and react to a query in either text or voice input. The proposed architecture can be implemented in many service areas if adequate training data is available. The proposed virtual assistant is capable of improving customer service and productivity while reducing human workload.

Acknowledgments

We would like to take this opportunity to express our heartfelt gratitude and deep appreciation to our module coordinators, Dr. Iromi Ranaweera and Dr. Keerthi Gunawickrama from the Department of Electrical and Information Engineering, for their assistance and valuable information and guidelines on the final year project. We would like to express our heartfelt gratitude to Dr. Kushan Sudheera, Department of Electrical and Information Engineering, for directing us to select the project and for providing assistance, guidance, documentation, and encouragement to our project.

Contents

List of Figures

List of Tables

Acronyms

AI	-	Artificial Intelligence
ML	-	Machine Learning
RNN	-	Recurrent Neural Network

Chapter 1

Introduction

Modern science relies heavily on machine learning and deep learning techniques to accomplish tasks more efficiently than humans. In the past, limited processing power and storage capacity hindered the use of these technologies. However, advancements in computer hardware have made it possible for developers to utilize self-learning models that are faster, more accurate, and more relevant for collecting, processing, and distributing information than traditional systems. Machine learning and artificial intelligence have become increasingly popular for a variety of applications, such as recommendation engines, facial recognition, and voice recognition.

As we interact with the global community, we often use English to engage with technical platforms like mobile, web, and desktop applications. In the past, these applications often had more static content and many of the applications used in Sri Lanka were also developed in English. However, modern applications are becoming more flexible and are incorporating artificial intelligence capabilities. In this environment, the global trend is towards using virtual assistants as the demand for AI grows.

1.1 Background

In order for algorithms to learn from data and make judgments or predictions, artificial intelligence (AI) techniques known as deep learning are used. Deep learning is inspired by the composition and operation of the human brain, Deep learning entail training artificial neural networks using vast data sets. These neural networks use numerous layers of interconnected nodes, which process and transfer information similarly to how neurons in the human brain do, to learn and recognize patterns in data.

A virtual assistant is an artificial intelligence system that uses natural language processing or deep learning to help people with tasks or provide answers. They can be accessed through a variety of platforms, such as websites, mobile apps, and smart home appliances. In recent years, Apple's Siri, Google Assistance and Amazon's Alexa are popular Deep learning-based virtual assistants that are being used.

Some research has been conducted on the creation of deep learning-based virtual assistants for the Sinhala language, which is spoken by the vast majority of Sri Lankans. In Phase I of this project, the virtual assistance is able to understand the context of the questions to some extent. Overall, the creation of deep learning-based

virtual assistants for the Sinhala language is an exciting opportunity to increase the availability and accessibility of language-based technologies.

1.2 Problem Statement

Virtual assistants are widely used around the world, but a major limitation is that they are typically only able to understand and respond to English. In Sri Lanka, where Sinhala and Tamil are the most commonly used languages, this can be a problem for many people who do not have strong English literacy skills. Singlish, a combination of English and Sinhala, is also commonly used by Sri Lankans. As a result, many people in Sri Lanka are unable to fully utilize virtual assistants due to their limited English proficiency. This initiative aims to address this issue and make virtual assistants more accessible to Sri Lankans who do not have strong English skills.

Several companies in Sri Lanka, have developed basic versions of automated Sinhala agents that can provide limited responses to selected user inputs. However, these assistants are not capable of serving the needs of all customers. Even when people are fluent in other languages, they often prefer to communicate in their native language, especially when seeking assistance from a personal assistant. The goal of this framework is to make it easier for people to interact with machines in their own language, and to provide fully automated support in Sinhala to improve the customer experience.

The propose of this framework is designed to be adaptable to a variety of use cases, including customer service and personal assistant applications such as Amazon Alexa. Its flexibility is derived from its ability to be retrained on different data sets, allowing it to be customized to fit the requirements of a specific service. Our current focus is on creating a general framework that can be easily modified to suit a range of contexts. The potential applications of the virtual assistant are numerous, depending on the availability of data resources. For instance, by training it on a dataset of queries between clients and operators, the model could be deployed as a customer service platform with minimal additional preprocessing.

1.3 Objectives and Scope

Objectives

- Finding the most appropriate data set for training the model.
- Construct a general framework that has the ability to be retrained for specific services by replacing the data provided.
- The developed framework should be able to accurately capture and effectively respond to the unique characteristics of the Sinhala language, and to provide the optimal solution to a given query.
- The framework should possess the capability to detect human vocalization and respond in a suitable manner.

Scope

- The framework can be further trained to accommodate a specific service sector.
- General framework, comprising natural language processing and deep learning components, that is able to accept and respond to both spoken and written Sinhala and Singlish natural languages.

Chapter 2

Literature Review

The automated method of text analysis known as Natural Language Processing is based on a number of theories and technologies. Natural language processing has been a hot topic since 1950. People demanded that the machines behave like people. The machines needed to be able to comprehend human voice and text. Thus, one of the most discussed subjects in this context was natural language processing, or NLP for short. NLP is a part of the AI system. However, in order to move further with the limited computational resources at their disposal, the developers initially faced major obstacles. Assembler was the language of choice for practically all programming because there were no successful higher-level languages. Machines were extremely slow, had little storage, and frequently had limited access. However, thanks to the enormous advancements in computing technology in the twenty-first century, NLP has overcome many of its difficulties and made significant progress while continuing to be a hot topic. Natural language processing has been a hot issue in the 21st century. Today, a variety of applications, including chatbots, information extraction, spelling correction, keyword searching, and translation, use natural language processing.

"An application that utilizes information, such as the user's speech and logical data to supply support by noting questions in standard dialect, making ideas, and doing tasks" is what an Intelligent Virtual Assistant (IVA) is. As AI personal assistants can enhance workflows by automating and streamlining routine tasks, as well as as an AI assistant developed on the basis of business can be used to simplify and automate basic customer service to customers or sales transactions, it has been one of the most significant and beneficial endeavors in the world. As a result, it has been among the most extensively used communication channels. Experts claim that there are many virtual assistants available in today's world, including Alexa, Apple SIRI, Microsoft Cotana, and others.

The development of a human-computer dialog system that is capable of using natural language has been deemed a more challenging area of artificial intelligence research. This is because natural languages are complex and because the system needs to be able to comprehend and produce natural language responses. Eventually, the engineers were able to connect ELIZA and the intelligence.

2.1 Systems based on Machine learning Approaches

This is an illustration of a template-based architecture, where the system uses questions and answers that have already been created. The system generates the final result by comparing the user's input queries to pre-existing patterns using established criteria. a major Only if the user's input inquiry is strikingly similar to an existing pattern will the user receive a response.

Joseph Weizenbaum created ELIZA, a ground-breaking AI program, in 1966. It utilizes the MIT MAC time-sharing system. It was a family of programs, but "Therapist ElIZA," which was based on Weizenbaum's DOCTOR screenplay, was the most well-known. Decomposition rules are used to examine the input texts after being activated by the input's keywords. When it found a keyword in the input, the computer altered the sentence using the scripted rules for that keyword. Although ELIZA was a ground-breaking invention at the time, it had serious NLP limitations.

A.L.I.C.E. (Artificial Linguistic Internet Computer Entity), which is regarded as the first personality program built on AIML (Artificial Intelligence Mark-up Language), a derived of XML language for developing stimulus-response architecture, is a language for developing stimulus-response architecture. One of the most well-known chatbots is ALICE, which has been declared "The most human computer" for the past three years, 2000, 2001, and 2004. Between 1995 and 2000, the Alicebot free software community created it to enable users to input dialogue pattern information into chatbots built using the ALICE free software technology. By applying basic pattern matching methods, ALICE—which could be seen as a modification of ELIZA—responds to an input sentence. Two stages are required for each AIML interpreter input query before the sentence is ready for pattern matching. For each sentence's input path, these are the normalization and production operations. The problem with ALICE is that it doesn't understand and reply to the user's questions or input sentences. It merely extracts the matching response from the knowledge base and outputs it.

Eliza is modeled like Elizabeth, but with more adaptability and versatility. Elizabeth's knowledge is scripted and saved in a text file. There are four sections in the script file. It provides command lines for the system's reaction in situations such as Welcome (which the system randomly selects at the start.), Void (which the system randomly selects if the user input is an empty string or if the user simply passes by pressing "enter"), and those in which the user simply passes. and No keyword messages (when the system is unable to recognize any keywords in the user's input using pattern matching). The input transformation rules are included in the second portion and are what allow user input to be compliant with the set of established keyword patterns. The rules for output transformation, where the outputs are moved in accordance with the nouns and pronouns of the phrase, are contained in the third part of the script file. Finally, the final result is based on the keyword matching patterns. As a result, input rules, keyword patterns, and output rules all contribute to the final response. Elizabeth is a pattern matching rule-based system as well, and the system does not adapt to the user's preferences. In 2006, B. Hettige et al. used Prolog-based Natural Language Processing Modules to create a Sinhala chatbot.

The following was included in this NLP module:

- Sinhala Morphological Analyser: This program word-by-word analyzes input sentences to identify the morphological information of Sinhala words, such as nouns and verbs.
- The Sinhala Parser analyzes the text provided as input and determines the structure of the Sinhala text.
- Based on Sinhala grammar, the Sinhala Morphological Generator generates pertinent Sinhala words.
- Sinhala Sentence Composer: creates the text in Sinhala.

All of the aforementioned Language Support Modules were developed using SWI-Prolog. This system is entirely text-based and does not accept voice inputs.

2.2 Systems using pattern matching techniques

One of the ML algorithms, the TF-IDF technique, rates each word in the corpus according to how frequently it appears. The most popular ones are thought to have a good chance of coming up in a search. This is widely utilized in information retrieval and keyword extraction. Anupam Mondal et al. created a chatbot, an automated conversation system for the educational sector, employing the TF-IDF technique and the random forest ensemble learning strategy. Data cleaning, integration, transformation, and discretization were used as preparatory processes to organize the initial 1000 pairs of unique questions and replies from educational chat data. The NLTK package and WordNet lexicon were used during the feature extraction process. Instead of employing a decision tree model, random forest was used to train the model because it has the potential to overcome the overfitting issue. The proposed chatbot demonstrated an average F-measure 0.870 score on various K-values in the validation system. The system was failing to create a dialog-style dialogue with the user because it was only responding to input requests. The bot has been set up as a Telegram bot. Construction of an Expert Recommender A suggestion management system called a chatbot was created by Cerezo at el to help developers within a specific developer community locate the right person to contact within open source projects. The input sentences were categorized using NLP techniques, and the main concepts were found using TF-IDF. Similar to , this one was unable to detect queries outside of its purview, could only respond to certain questions, and could not start a discussion with the user.

A comprehensive Question Answering system for the travel domain was provided by Hasangi Kahaduwa et al. Finding the query and searching the knowledge base for an answer to the classified question are the two steps of the system. A machine learning approach is utilized to identify the question, and a rule-based approach is used to search the knowledge base for the solution. The question identification phase produced good results, but the rule-based approach failed to produce encouraging results because there was no rule basis.

Oatmeal By Giovanni Campagna et al., a system called The Architecture of an Open, Crowdsourced, Privacy-Preserving, Programmable Virtual Assistant was created and put into action. Among the key ideas presented are the open-source Thingpedia knowledge base, the open-source ThingSystem for handling user data, the high-level ThingTalk language for connecting devices, and finally a machine-learning-based translator that can convert commands in natural language into trigger-action code. It is designed to be the first virtual assistant that can comprehend trigger-action commands in natural language.

Almond Deep A more sophisticated version of Almond was created by Giovanni Campagna and Rakesh Ramesh based on Language-to-code synthesis of Trigger-Action programs using Seq2Seq Neural Networks. In place of Almond's foundational semantic parsing framework, SEMPRE, this study looked at cutting-edge deep learn-

ing methods to increase Almond's accuracy. Each stage is a Long Short Term Memory (LSTM) unit in a 1-layer Recurrent Neural Network (RNN) model. The seq2seq neural network model analyzes the input natural language and forecasts the program's output. To make it logical to give some inputs more weight when producing different parts of the output, an attention layer is introduced.In this study, it is demonstrated that the suggested deep learning model has the potential to outperform the present SEMPRE-based system in identifying a larger set of trigger-action programs. The proposed parser's lack of the ability for the user to correct the assistant is a drawback compared to SEMPRE. Almond Deep An improved version of Almond, known as a Deep Learning-based Virtual Assistant, was created using Trigger-Action Language-to-Code Synthesis.

A paradigm for answering questions using DL has been put into practice by Sroh and Mathur . The algorithm has been taught to glean information from stories in order to answer queries. Two different baseline models had been tested. A GRU model that uses Keras as a baseline model is shown first, followed by the seq2seq model as a second strategy. Finally, they decided on the seq2seq model over the GRU model since it worked better with the bAbI dataset and the GRU model read the response as a single word. GloVe word vectors were used to construct the seq2seq model, which can respond to queries based on a given tale. They used an RNN encoder to process the question first, then the story, with the question-start sign coming after the story (Q). The decoder uses cross-entropy error to give a result. The decoder starts decoding with a particular GO symbol and ends with a unique STOP symbol. For questions with yes-or-no answers, the proposed system has lived up to expectations, but it has fallen short when dealing with queries that require longer solutions.

Chapter 3

Methodology

Project Requirements and Scope are clearly grasped after having meetings with the supervisors. Phase I of the project was studied and able to get clear context of the current state of the project. Learning about similar implementations and researches carried out by scholars around the world, background research and a literature study were done in the domain of deep learning-based virtual assistance.

The major criteria of the Deep learning-based virtual assistant were established based on the knowledge gained.

- Improve the data set for model training and validation
- The current model should be enhanced for improved accuracy of the answers.
- Improve user experience of the virtual assistance.
- Integrating the voice recognition and speak capability to the system.
- Integrating a virtual assistant into the software based application.
- Integrating the virtual assistance to an embedded device.

As the first step, we become acquainted with the technologies essential for project implementation and gain a thorough understanding of the major mechanisms pertaining to phase one of the project.

3.1 System Architecture

The system architecture of a deep learning-based virtual assistant will typically consist of several key components, including Data input and preprocessing, Deep learning model, Decision-making and task execution, Output and response generation, User interface and Infrastructure and support systems.

3.1.1 Data input and preprocessing

The virtual assistant must accept and process data input data. Preprocessing the data to extract key features or cleaning and formatting it for use by the model may be required.

3.1.2 Deep learning model

Deep learning model is the core component of the Virtual assistance application. The model could be made up of numerous layers of artificial neural networks and trained using approaches like supervised learning, unsupervised learning, or reinforcement learning. Deep learning models for natural language processing include RNN, Word2Vec, LSTM and transformer models. Our primary goal is to develop a generative pre-trained transformer with the goal of processing natural language input and providing the appropriate answer. We must map the projected answer to the user-provided query.

3.1.3 Decision-making and task execution

The deep learning model will make decisions or perform activities based on the processed data inputs, such as responding to user requests or regulating device functionalities.

3.1.4 Output and response generation

The virtual assistant must generate outputs and answers based on the model's decisions or tasks. This could include responding in natural language, displaying information on a screen, or controlling device tasks.

3.1.5 Infrastructure and support systems

Infrastructure and support systems, such as servers or cloud resources, may be required for the virtual assistant, as well as tools for monitoring and maintaining the system. Google Colab, Azure ML or Amazon Saga maker can be used for this purpose.

Chapter 4

Preliminary Results

Chapter 5

Work Plan and Budget

Chapter 6

Significance and Implication of the Project

Bibliography