

**ENHANCING CONVERSATIONAL AI MODEL
PERFORMANCE AND EXPLAINABILITY FOR
SINHALA-ENGLISH BILINGUAL SPEAKERS**

2022-056

Project Proposal Report

Sakalasooriya S.A.H.A.

(Dissanayake D.M.I.M., Hameed M.S., Jayasinghe D.T.)

B.Sc. (Hons) Degree in Information Technology Specialising in Data
Science

Department of Information Technology

Sri Lanka Institute of Information Technology
Sri Lanka

January 2022

**ENHANCING CONVERSATIONAL AI MODEL
PERFORMANCE AND EXPLAINABILITY FOR
SINHALA-ENGLISH BILINGUAL SPEAKERS**

2022-056

Project Proposal Report

B.Sc. (Hons) Degree in Information Technology Specialising in Data
Science



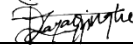

Department of Information Technology

Sri Lanka Institute of Information Technology
Sri Lanka

January 2022

DECLARATION, COPYRIGHT STATEMENT AND THE STATEMENT OF THE SUPERVISORS

We declare that this is our work, and this proposal does not incorporate without acknowledgement any material previously submitted for a degree or diploma in any other university or institute of higher learning and to the best of our knowledge and belief it does not contain any material previously published or written by another person except where the acknowledgement is made in the text.

| Name | Student ID | Signature |
|------------------------|-------------------|---|
| Dissanayake D.M.I.M. | IT19069432 |  |
| Hameed M.S. | IT19064932 |  |
| Jayasinghe D.T. | IT19075754 |  |
| Sakalasooriya S.A.H.A. | IT19051208 |  |

The above candidates are carrying out research for the undergraduate Dissertation under my supervision.

Name of the supervisor: Dr. Lakmini Abeywardhana

Signature of the supervisor:

Date: 11/02/2022

Name of the co-supervisor: Ms. Dinuka Wijendra

Signature of the co-supervisor:

Date: 11/02/2022

ABSTRACT

Nowadays conversational AIs (Artificial Intelligence) are playing a key role in business. So, businesses introduce conversational AIs also known as “chatbots,” as their interface between customers and businesses to improve customer interaction. Integrability of chatbots with existing systems is a main advantage to increase the use cases of chatbots. Named entity recognition is used by conversational AIs to give a more accurate response by understanding name entities in the input text. Custom Named entity tagging is a data pre-processing step in the named entity recognition (NER) process. Annotating a corpus by tagging domain-specific named entities is a very time-consuming task it also requires expert knowledge. Annotation time can be reduced by using a software tool that can provide suggestions during the text annotation task. This research component introduces a tool, “SIENA” (Sinhala – English entity annotator) that can increase the efficiency of named entity tagging tasks when the data set contains Sinhala – English code-switching data. This text annotation tool has a CLI version and GUI version (web application).

When corpus was partially tagged with custom named entities by human interaction, the tool will be able to automatically tag or suggest further words by looking at the already tagged words by checking the similarity between tagged words and untagged words which are supposed to tag by human interaction.

Keywords: custom named entity tagging, text annotating, named entity recognition, chatbots

TABLE OF CONTENTS

| | |
|---|-----|
| Declaration, Copyright Statement and The Statement of The Supervisors | i |
| Abstract | ii |
| Table of Contents | iii |
| List of Figures | v |
| List of Tables..... | vi |
| List of abbreviations..... | vii |
| 1. Introduction | 1 |
| 1.1 Background & Literature survey | 1 |
| 1.2 Research Gap..... | 7 |
| 1.3 Research Problem..... | 8 |
| 2. Objectives..... | 9 |
| 2.1 Main Objectives | 9 |
| 2.2 Specific Objectives..... | 9 |
| 3. Methodology | 10 |
| 3.1 Requirements Gathering and Analysis | 10 |
| 3.1.1 Functional requirements..... | 10 |
| 3.1.2 Non-functional requirements | 11 |
| 3.2 Feasibility Study..... | 11 |
| 3.2.1 Technical feasibility | 11 |
| 3.2.2 Financial feasibility..... | 11 |
| 3.2.3 Legal feasibility..... | 11 |
| 3.2.4 Operational feasibility..... | 12 |
| 3.2.5 Scheduling feasibility..... | 12 |
| 3.3 Preparation of Datasets..... | 12 |
| 3.3.1 General dataset for machine learning model training | 12 |
| 3.3.2 Domain specific dataset for conversational AI training..... | 13 |
| 3.4 Individual Component Architecture..... | 13 |
| 3.5 SIENA algorithm development | 15 |
| 3.5.1 Revere stemming approach..... | 15 |
| 3.5.2 Word-wise cosine similarity | 15 |

| | | |
|-------|---|----|
| 3.5.3 | N-gram approach..... | 15 |
| 3.6 | SIENA tool evaluation | 16 |
| 3.7 | User interfaces and visualizations | 16 |
| 3.8 | Tools and Technologies..... | 17 |
| 4. | Description of Personal and Facilities | 18 |
| 5. | Gantt Chart | 19 |
| 6. | Commercialisation plan | 20 |
| 7. | Budget and Budget Justification | 24 |
| | References | 25 |
| | Appendices..... | 27 |
| | Appendix A: Survey Form..... | 27 |
| | Appendix B: Supervision Confirmation Emails | 38 |

LIST OF FIGURES

| | |
|---|----|
| Figure 1.1: Chatbot with named entity recognition | 4 |
| Figure 1.2: Chatbot without named entity recognition | 5 |
| Figure 1.3: Summary of survey responses received for the question "Do you prefer if chatbots can identify Dates / Names / Registration numbers / Places / Lecture Halls and other similar data automatically or do you prefer filling forms instead?" | 6 |
| Figure 1.4: Summary of survey responses received for the question "What languages would you prefer to use the most for chatting if you had to interact with a chatbot?" | 7 |
| Figure 3.1: Summary of survey responses received for the question "say there is a tool to annotate data when preparing them to train a ML model. What kind of a tool do you prefer out of the given options?" | 10 |
| Figure 3.2: SIENA tool high-level architecture | 13 |
| Figure 3.3: High-level architectural diagram of the purposed solution with all research components integrated | 14 |
| Figure 3.4: Wire frame - text annotation user interface | 16 |
| Figure 4. 1: Work breakdown structure of the individual research component..... | 18 |
| Figure 5.1: Gantt chart for overall project and the individual component..... | 19 |
| Figure 6.1: Commercialisation plan..... | 23 |

LIST OF TABLES

| | |
|--|----|
| Table 1.1: Examples for domain specific named entities | 3 |
| Table 1.2: Comparison with existing research related to text annotation tools | 8 |
| Table 3.1: Summary of Tools and Technologies to be used according to the tasks .. | 17 |
| Table 6.1: Feature variations and cost difference of packages proposed by the commercialisation plan | 22 |
| Table 7.1: Budget justification for the overall research project..... | 24 |

LIST OF ABBREVIATIONS

The list of all the abbreviations used in this report is in the following table.

| Abbreviation | Description |
|--------------|--|
| AI | Artificial intelligence |
| AIML | Artificial Intelligence Mark-up Language |
| CLI | command line interface |
| CNN | Convolutional neural network |
| GUI | graphical user interface |
| LSTM | Long short-term memory |
| ML | Machine learning |
| NER | named entity recognition |
| NLP | Natural language processing |
| NLU | Natural language understanding |
| RNN | Recurrent Neural Network |
| XML | Extensible markup language |
| CDD | Conversation Driven Development |
| SaaS | Software as a service |
| CaaS | Conversational AI-as-a-service |

1. INTRODUCTION

1.1 Background & Literature survey

Digital transformation is a common thing in this era. Computers with artificial intelligence take the place of human activities. Conversational AIs, also known as chatbots are widely used as virtual assistants for foods selection, booking tickets, online shopping, website guidance and entertainment purposes. It is software that simulates human conversations. Through the advancement of NLP and ML technologies, chatbots can understand human needs through textual inputs.[2] Therefore, chatbots can use to increase customer service and satisfaction at anything by placing a chatbot when customer interaction happens with the business.

The first idea of chatbots came in 1950 by Alan Turing from his article "Computing Machinery and Intelligence". [3] According to this article, he described the concepts of intelligent machines which can interact with a human by textual conversations. In 1966 Joseph Weizenbaum created ELIZA. ELIZA is the first implementation of chatbots in history and it was a key point in history that leads to the development of chatbots. It can identify some pre-defined clue words or word patterns, according to identified words and patterns ELIZA can reply with pre-prepared responses which can continue the conversation. However, the pattern matching approach is not appropriate due to variations of sentence patterns in different domains. Another key point in chatbot history is the implementation of A.L.I.C.E. (Alice Bot, or Alice) in 1995 by Richard Wallace using the extension of an XML schema called "Artificial Intelligence Mark-up Language" (AIML). This is the first use of artificial intelligence in chatbots. AIML schema consists of a set of rules which determines the conversational capabilities of chatbot by natural language understanding. when more rules are added to the AIMA schema the chatbot became more intelligent. AIML schema helps to decide a reply to a given text. Therefore, AIML is the backbone of A.L.I.C.E. Developers can expand the knowledge base of AIML based chatbots by adding Data objects into the AIML schema.

There are two types of chatbots

- Rule-based chatbots
- Machine Learning techniques based chatbots

Rule-based chatbots are comparatively easy to develop but they can only give answers to questions within the ruleset. Because they are built with a predefined ruleset. Therefore, rule-based chatbots have some disadvantages. Lack of naturality when replying, unable to learn from previous conversations are some of them. Machine learning helped to reduce the drawbacks of rule-based chatbots. Machine learning based chatbots are much more intelligent than rule-based chatbots. They can handle several types of sentence patterns, learn from their previous conversations, and train themselves for future conversations. Machine learning based Conversational AIs use various kinds of deep neural network architectures to train a model. Machine learning based Conversational AIs are continually evolving due to the vast improvement of NLP technologies and machine learning techniques within the past few years.

When a user asks something from a chatbot it needs to understand the input text. This process is called natural language understanding (NLU). The named entity recognition (NER) approach can be used for NLU tasks. [7] [6] There are neural network-based models (Bidirectional LSTM, and Bidirectional LSTM-CNNs) and statistical models (conditional random field) to identify name entities. In this research, neural network-based models are considered. Neural network-based models require a considerable amount of training data to converge by adjusting weights. Usually, when building a Conversational AI (Artificial Intelligence), it should focus on a domain, otherwise, chatbots will not perform well according to business requirements. The data set should be created according to the selected domain, by covering the domain-specific vocabulary.

[4] Adding linguistic information to a corpus called text annotation or named entity tagging. Named entity tagging is responsible for indicating the class of words in a corpus. As an example, named entities are predefined categories of real-world objects,

such as a person, location, time, expressions, organization, product. also, it can be a domain-specific class of words.

Table 1.1: Examples for domain specific named entities

| Domain | Name entity | Example words |
|-----------------------|----------------|---|
| Biological domain [5] | virus | Coronavirus, Cocksackievirus, Dugbe |
| | bacteria | Hafnia spp, Escherichia coli, Citrobaacter koseri |
| Commerce domain | currency | Yen, LKR, USD (United States Dollar), ₹ |
| | payment method | visa card, master card, PayPal, crypto |

Named entity recognition helps to determine the most suitable intent for a given text by identifying named entities in the text. In this case, the text is the user's question, the intent is the chatbot's reply. If the chatbot does not have an ability to identify name entities chatbot will work in form filling way as shown in Figure 1.2.

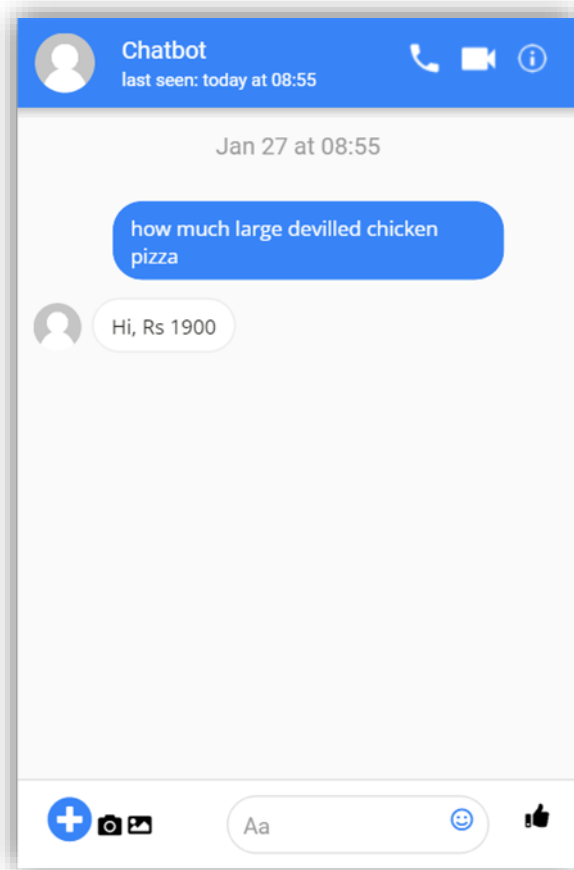


Figure 1.1: Chatbot with named entity recognition

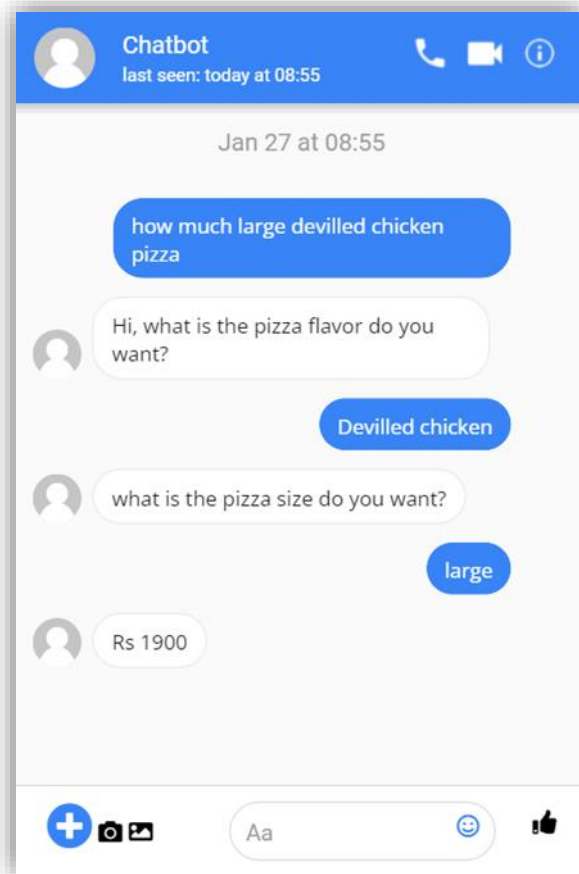


Figure 1.2: Chatbot without named entity recognition

According to the survey results, most people like to see chatbots with automatically identifying name entities instead of form filling style.

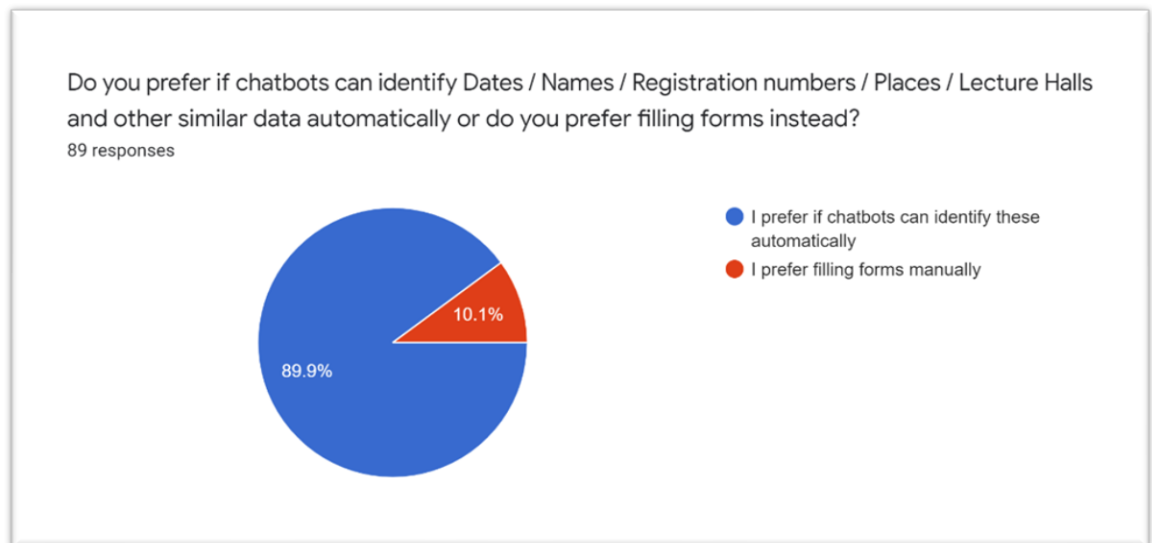


Figure 1.3: Summary of survey responses received for the question "Do you prefer if chatbots can identify Dates / Names / Registration numbers / Places / Lecture Halls and other similar data automatically or do you prefer filling forms instead?"

If the corpus contains large number of words, it takes a considerable amount of time to do the tagging process. Because every word in the corpus should be read and identified by the person who is doing the annotation task. Therefore, creating data sets for named entity recognition models is a very time-consuming task and it requires expert knowledge. In this research, Sri Lankan peoples are focused therefore most Sri Lankans are using the Sinhala-English code-switching language style when they interact with a chatbot. This statement is proven by the survey.

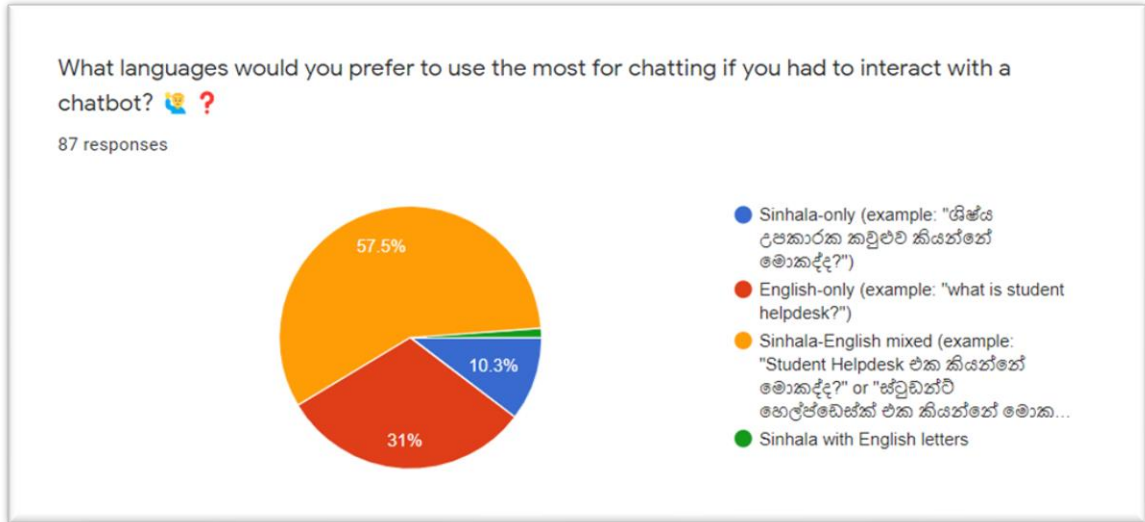


Figure 1.4: Summary of survey responses received for the question "What languages would you prefer to use the most for chatting if you had to interact with a chatbot?"

1.2 Research Gap

Researchers in named entity recommendation area are used different kinds of approaches to implement automated and semi-automated text annotation tools. 'ANEA' tool [8] inherits knowledge from Wiktionary (online dictionary) to derive the most suitable named entity type for given word. If Wiktionary doesn't contain the required word this tool cannot guess the word entity. 'brat' [9] is a web-based annotating tool that cannot be used when GUI is available in some cases. 'bart' has a named entity recommendation system using semantic class disambiguation algorithm. GATE [10] mainly focus on collaborative annotation which is not focused on this research component. YEDDA [11] has entity recommendations during the text annotation process by Maximum Matching algorithm which is a text segmentation algorithm. There is no optimized name entity tagging tool for Sinhala – English bilingual text annotation.

Table 1.2: Comparison with existing research related to text annotation tools

| Tool name | Collaboration features | Name Entity recommendations | Sinhala word variation identification |
|---|-------------------------------|--|--|
| ANEA | no | yes (<i>via external knowledge source</i>) | no |
| brat | no | yes (<i>via semantic class disambiguation</i>) | no |
| GATE | yes | no | no |
| YEDDA | no | yes (<i>via maximum matching algorithm</i>) | no |
| SIENA (<i>This research component</i>) | no | yes | yes |

SIENA tool designed for Sinhala English code-switching text annotation tasks. It can identify variations of Sinhala words. By using the SIENA tool, text annotating persons can do their task effectively on Sinhala English code-switching corpus when they require a domain adaptation of named entity recognition.

1.3 Research Problem

[12] Sinhalese language, also known as Sinhala (සිංහල) is one of the two official languages of Sri Lanka, with about 16 million speakers out of the total population of 21 million also Sinhala is not a worldwide spread language like English. Due to those reasons, there is a lack of NLP tools designed for Sinhala. According to the survey results, Most of Sri Lankan people use the Sinhala-English code-switching language style (Figure 1.4). Therefore, the SIENA tool is a semi-automated tool that can identify name entities during custom named entity tagging in an English - Sinhala code-switching corpus. There are few named entity tagging tools are available, but none of them are used. Most of the tools are language-specific or domain-specific. Therefore, those tools are not optimized for Sinhala - English code-switching corpus.

2. OBJECTIVES

2.1 Main Objectives

The main objective of implementing the SIENA tool is to increase the efficiency of the text annotation process in a Sinhala-English code-switching corpus by providing accurate name entity recommendations. Efficiency is calculated by analysing the correct name entity recommendations.

2.2 Specific Objectives

To archive the main objective, the specific objectives that need to be fulfilled are identified as follows

1. Define the recommendation hierarchy

By evaluating the results of the proposed three methods in methodology subsection, need to assign a score to each recommendation algorithm to rank the suggested name entities.

2. Make SIENA compatible with frameworks

After annotation is done. The tool should be able to export data with the compatibility of famous libraries which are used to build NER models. For this research component spaCy was selected as the compatible NER library.

3. Develop visualizations technique to provide user friendly suggestions

Suggestions should be displayed in a user-friendly manner.

4. Make knowledge base as module component

Users could be able to import/export knowledge base. Therefore, users can attach existing knowledge base into the SIENA tool.

3. METHODOLOGY

3.1 Requirements Gathering and Analysis

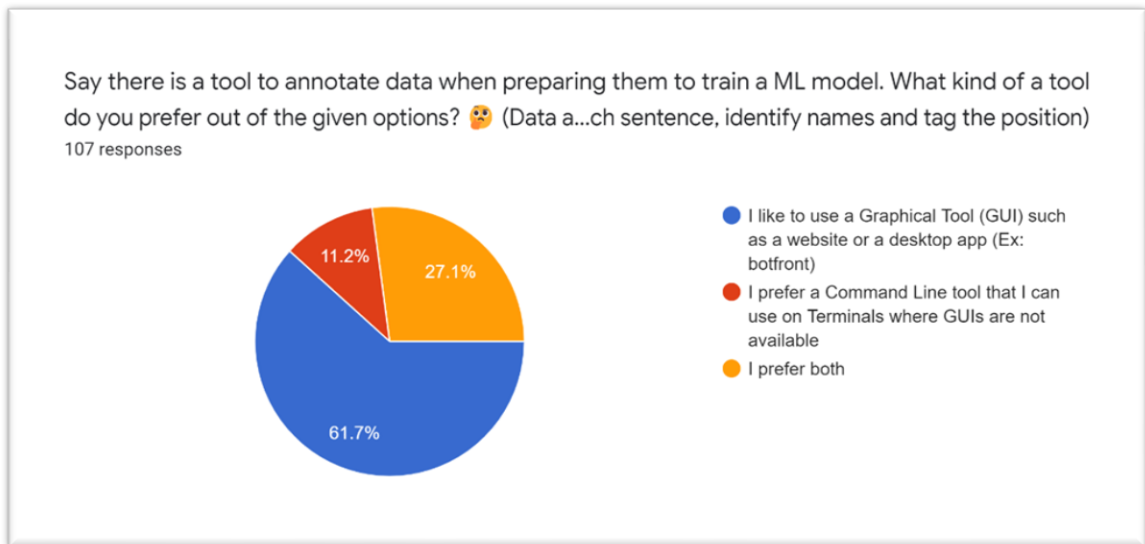


Figure 3.1: Summary of survey responses received for the question "say there is a tool to annotate data when preparing them to train a ML model. What kind of a tool do you prefer out of the given options?"

According to the survey, most of the participants vote for GUI based tools. And there are considerable number of participants like to use both GUI and CLI tools. Therefore, SIENA tool will be developed as GUI as well as a CLI tool.

3.1.1 Functional requirements

1. User should be able to find recommended name entities
2. User should be able to import / upload corpus into SINEA
3. User should be able to export annotated text from SIENA
4. User should be able to import / upload portable knowledge base into SINEA
5. User should be able to export portable knowledge base from SINEA

3.1.2 Non-functional requirements

1. SIENA should be able to handle large text corpus
2. SIENA should be able to easily maintain
3. SIENA should be able to easily install on user's computer
4. SIENA should be reliable
5. SIENA should be secure

3.2 Feasibility Study

A feasibility can be separated into technical, financial, legal, operational, and scheduling feasibility of this research component.

3.2.1 Technical feasibility

The research component requires having knowledge in spaCy library, spaCy supported data types, NLP techniques, and software development. There should be enough processing power to run algorithms in SINEA tool. Training machine learning models in spaCy requires some amount of processing power. Computer storage space should be an enough to save model and corpus.

3.2.2 Financial feasibility

The cloud infrastructure for the SIENA tool or deployment should not be costly. Charges for domain name purchasing and cloud infrastructure resource usage will be covered by market value of the final product.

3.2.3 Legal feasibility

Datasets used in the research components should be publicly available datasets. Any scraped data from websites that are not public should have the approval of the original owners and should have written consent. The research component should not extract content as it is from previous work done without the written permission of the original authors. Any python packages or other software used must be open source or legally purchased, and the original authors must be credited if requested.

3.2.4 Operational feasibility

This research component should not conflict with other research components of the same research project or other research projects. The research component should fulfil the gap in the suggesting named entities for Sinhala – English text annotation task. The research component should be viable to fulfil the scope stated by the research project requirements.

3.2.5 Scheduling feasibility

The tasks of this research component should be aligned with the grant chart as shown in Figure 5.1

3.3 Preparation of Datasets

All research components require two datasets in total. Both datasets are domain-specific datasets with Sinhala-English code-switched text data. However, there are notable differences between the datasets and subsection 3.3.1 and subsection 3.3.2 explain these differences clearly. Both datasets utilize data augmentation techniques mainly to overcome the low-resource nature of Sinhala text data gathered, capture as many as code-switched phrases and collect as many distinct writing patterns as possible. All research group members will generate four versions of the samples in two datasets according to different code-switching styles. Duplicate data removal will be employed to ensure the quality of the collected data.

This research component only utilizes the General dataset for machine learning model training mentioned in subsection 3.3.1 to do the name entity tagging and the tool evaluation.

3.3.1 General dataset for machine learning model training

Scraped Sinhala-English code-switched textual data from websites which are related to SLIIT and SLIIT news articles are used to make the domain-specific dataset for doing machine learning model training and NLP text pre-processing tool creation. (A few examples are <https://support.sliit.lk/>, <https://sliitinternational.lk/>, and <https://www.sliit.lk/>). Furthermore, because the above SLIIT-based websites are both

public and official, publicly available documents such as PDFs and Docx files were used to tackle the low-resource issue.

3.3.2 Domain specific dataset for conversational AI training

Handcrafted Sinhala-English code-switched textual are used to make the second domain-specific dataset for text annotation through the SIENA tool. This dataset should be properly prepared as a training dataset for the intent classification problem of conversational AI. There will be around seventy-eight intents (classes), with each class containing a minimum of ten examples as a starting point. It's worth noting that the quantity of training instances may grow in the future to improve conversational AI's overall performance.

3.4 Individual Component Architecture

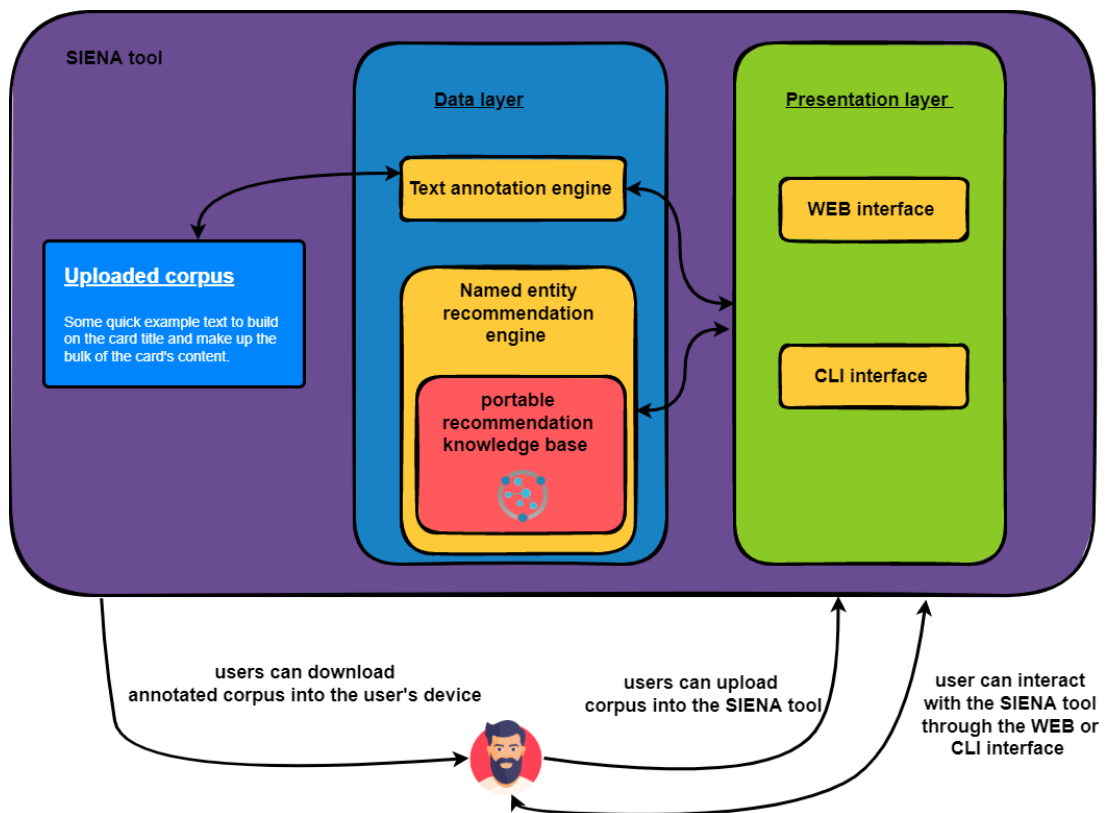


Figure 3.2: SIENA tool high-level architecture

SIENA tool will consist of two subcomponents data layer and presentation layer. Data layer is responsible for identifying name entities, presentation layer is responsible for provide interface to user interaction. SINEA tool will be used to annotate corpus which is used as training data by custom name entity recognition model in RASA NLU pipeline as shown in Figure 3.3.

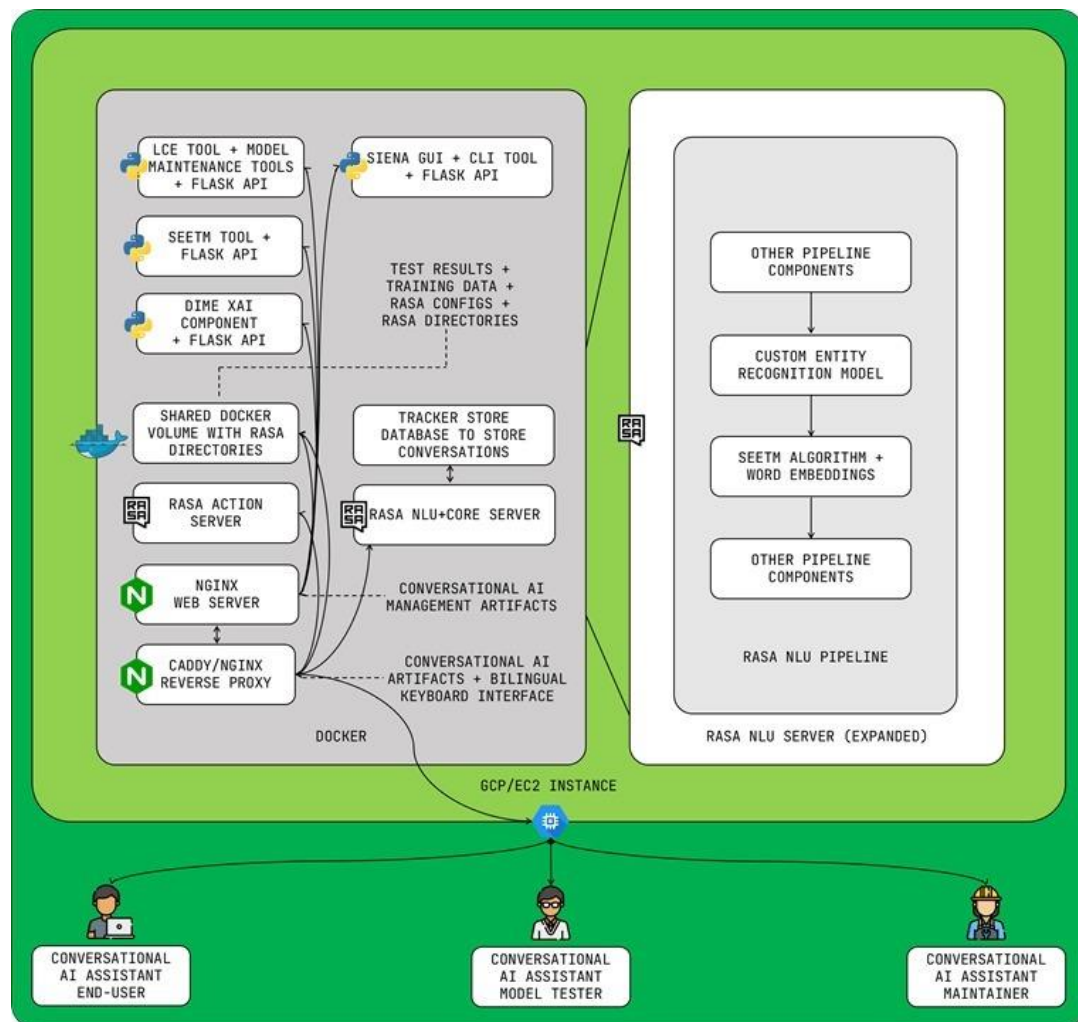


Figure 3.3: High-level architectural diagram of the purposed solution with all research components integrated

3.5 SIENA algorithm development

3.5.1 Revere stemming approach

Stemming is a pre-processing step in natural language processing. It is used to reduce grammatical forms of words. [12] This approach will convert words into the root form of the words. Revere stemming approach is assigning a Custom name entity to a base form of the word, then the tool can automatically tag words that are mapping into the same root form of the word. As an example, when annotator tagged “කපුටා” as a “bird” and suppose “කපුටා” will be “කපුට්” after the stemming. Then the tool will be able to map “කපුට්” to the “bird” class. Therefore, tool will suggest “bird” class to all the words which are mapped into “කපුට්” after the stemming.

To extract the Sinhala base form of words, suffixes should be removed by creating a Sinhala suffix list and analysing Sinhala grammar. Suffix list and Sinhala grammar rules are referred by [13] “Basaka Mahima”. English words are stemmed by using “Porter Stemmer”

3.5.2 Word-wise cosine similarity

Generally, cosine similarity is used to find the best matching document for a given text-based on cosine distance. In this case, it is used to find character-wise similar words. By using the whole alphabet of a language as dimensions of vector space, the position of a given word can be marked in the vector space and find similar words by using cosine distance.

3.5.3 N-gram approach

The N-gram approach is a traditional natural language processing method for extracting N-gram word chunks from a given sample of text. As an example, by considering N as 2 (bigram),

“SLIIT එකේ” will be transformed into “SL”, “LI”, “II”, “T<space>”, “<space>එ”, “එකේ”.

Therefore, by calculating percentage of matching n-gram similar words can be identified

3.6 SIENA tool evaluation

SIENA tool is a named entity suggestion tool. This tool can be evaluated by measuring the F1 score. F1 score can be calculated by manually checking the correctness of the recommended name entities.

$$F1 = 2 \times \frac{\text{precision} \times \text{recall}}{\text{precision} + \text{recall}}$$

3.7 User interfaces and visualizations

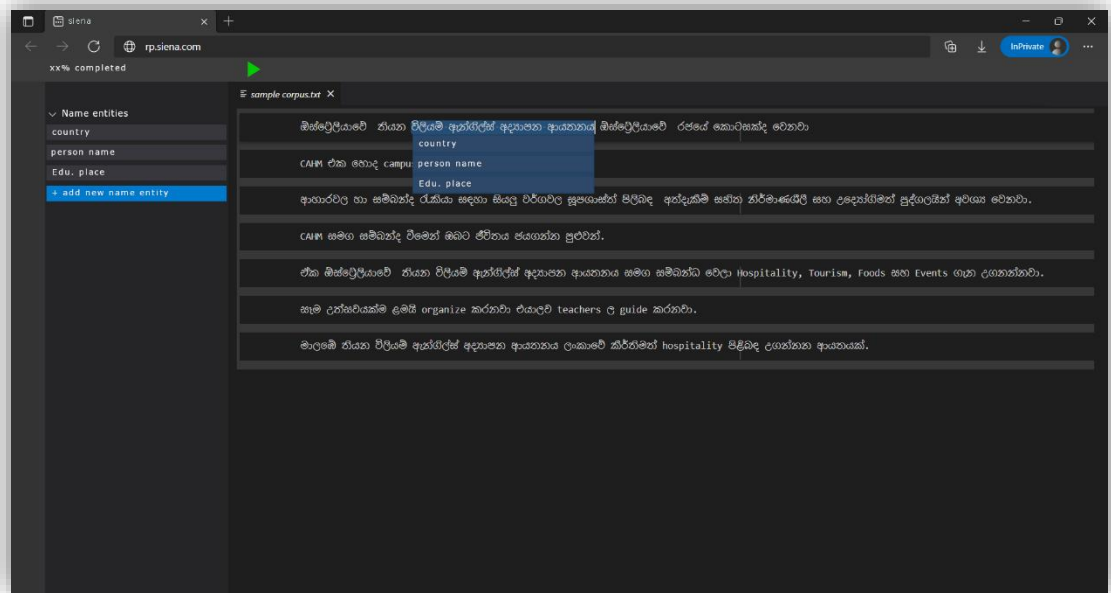


Figure 3.4: Wire frame - text annotation user interface

As shown in Figure 3.4, user can annotate text with the help of named entity auto identification

3.8 Tools and Technologies

SIENA tool will be developed using Python 3.8 with the help of Pandas, NumPy packages. SIENA will be deployed as a flask app inside a docker container. With the help of Google Cloud Platform or Amazon Web Services.

Table 3.1: Summary of Tools and Technologies to be used according to the tasks

| Task | Tools to be used |
|--|--|
| Algorithm Implementation and SIENA tool development | Python 3.8, NumPy, Pandas, PyCharm, Visual Studio Code, Google CoLab |
| Server development as a standalone frontend for the SIENA UI | Flask, JavaScript, CSS, React JS (optional) |
| Cloud Infrastructure management | One out of Google Cloud Platform and Amazon Web Services |
| Conversational AI development by Integrating all research components (Backend) | Rasa 2.8.12, MongoDB Atlas, Gensim, spaCy, Docker |
| Conversational AI frontend development | React JS, Bootstrap, socket.io, JavaScript |
| Overall system deployment | Caddy 2, NGINX, Git, Docker, docker-compose |

4. DESCRIPTION OF PERSONAL AND FACILITIES

Figure 4. 1 shows that all tasks of this individual research component under 5 main tasks.

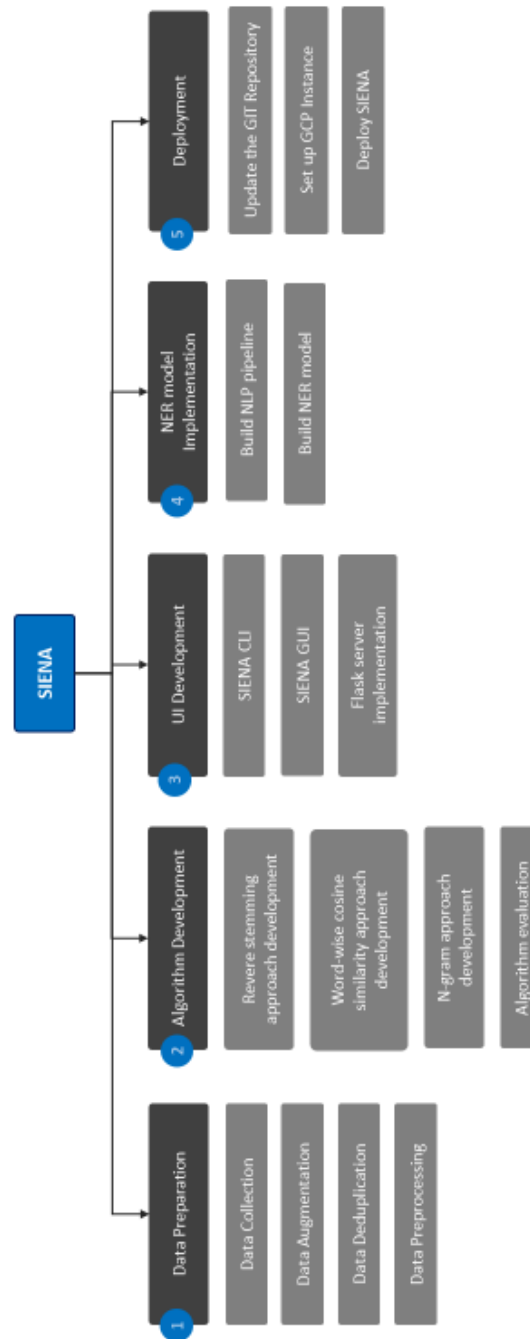


Figure 4. 1: Work breakdown structure of the individual research component

5. GANTT CHANT

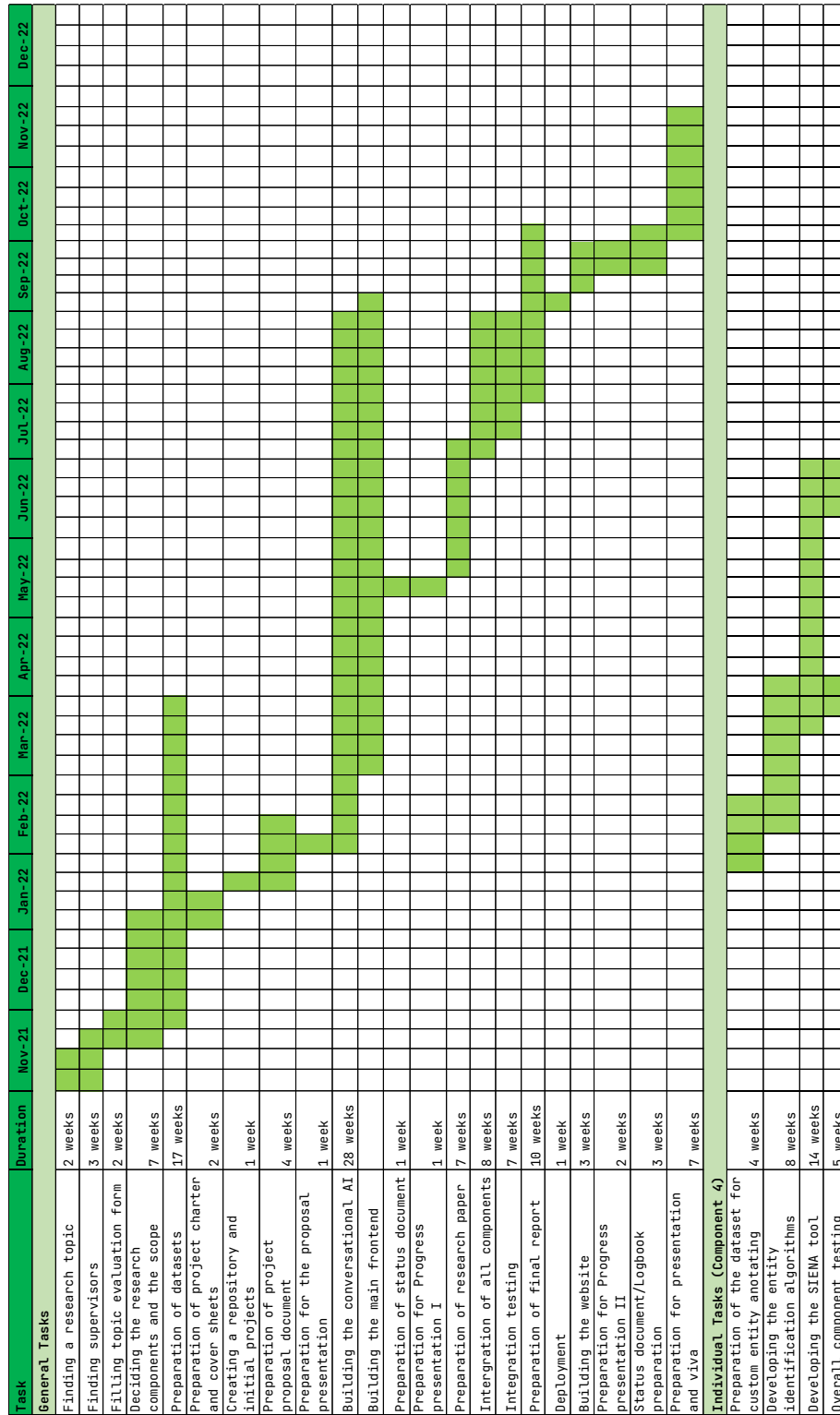


Figure 5.1: Gantt chart for overall project and the individual component

6. COMMERCIALISATION PLAN

Each research component of the overall research project provides applications for various Natural Language Processing and Machine Learning model evaluation-related tasks. Although the outputs of each research component can help design many products and services, they were all integrated to build a conversational AI that fully supports Sinhala-English code-switching. The main reason for developing a conversational AI are as follows.

1. Conversational AIs are a trending topic, and there is an increasing demand for Sinhala-based conversational AIs. Many businesses are looking to increase their customer reach by using conversational AIs for a vast range of business tasks, including providing technical assistance, automating manual tasks such as booking tickets, and providing the information requested by the customers. Although that is the case, it is hard to find Sinhala-based conversational AIs, especially in Sri Lanka. Thus, developing Sinhala-based conversational AIs was identified as a potential business opportunity.
2. Although there are many conversational AI development frameworks, most of them lack evaluation tools to debug machine learning models. In frameworks like Rasa, model evaluations are highly technical, and the average developers without machine learning knowledge fail to identify problems and debug the machine learning models. Solving this issue by allowing non-technical users to maintain and evaluate machine learning models and conversational AIs is another business opportunity where evaluation tools can be built and released as add-on features.
3. Businesses are moving towards cloud-based solutions, especially SaaS products from traditional standalone applications, web applications, and on-premises solutions, where the maintenance cost is high. A cloud-based highly configurable conversational AI would be an appropriate solution for many businesses where the effort for maintenance is considerably low.

The end product of the overall research concentrates on designing a solution for the above potential business ideas and opportunities. The conversational AI developed as the research end-product is mainly a SaaS or simply a CaaS (conversational AI-as-a-service) product that a business can purchase with a set of add-on features, as illustrated in . In addition to the CaaS packages, on-premises and demo cloud-based conversational AI packages are available for affordability and convenience. The end product of the overall research has the following archivable user benefits.

1. Businesses can purchase a CaaS package and eliminate conversational AI maintenance efforts.
2. Developers can use code-less maintenance tools to maintain conversational AIs they purchase without having in-depth knowledge about the backend deployment and the conversational AI development framework.
3. Businesses can purchase evaluation tools as add-ons and generate model explanations and evaluate machine learning models themselves to avoid extra maintenance costs. Here, the generated evaluations are easy to understand by non-technical users, which is not a feature of any existing chatbot framework.
4. Users of conversational AI can easily use the Sinhala-English code-switchable keyboard interface, and businesses can attract more users from having this feature as it eliminates the need to use third-party Sinhala typing services.
5. Businesses have the freedom to purchase either the CaaS packages or on-premises packages as per their need, while anyone can test the demo conversational AI for a period of 1 month before deciding to purchase any of the other packages that have a cost assigned.
6. Businesses can reach a vast customer range through Sinhala-English code-switching-based conversational AIs, especially within the Sri Lankan market, and it is possible to eliminate the need for having a dedicated customer care staff. It will dramatically lower the expenses of the business.

The proposed commercialisation plan of the end product of the overall research component contains a set of convenient packages and the offered features of each package differ based on the cost attached to it. The distribution of the features and the package cost were carefully planned and designed by analyzing the existing purchasable packages of similar SaaS products and conversational AIs. Table 6.1 clearly illustrates the feature variation and the cost difference of the packages offered by the proposed commercialisation plan.

Table 6.1: Feature variations and cost difference of packages proposed by the commercialisation plan

| Feature | Packages | | | | |
|---------------------------|----------|-------------------------------------|---------|---------|---------|
| | Demo | On-Prem | CaaS | | |
| | | | Starter | Pro | Genius |
| Intents | 10 | Unlimited | 20 | 180 | 400 |
| API Integrations | 2 | Unlimited | 2 | 110 | 200 |
| Bot Analytics | ✓ | ✓ | - | ✓ | ✓ |
| CDD | ✓ | ✓ | ✓ | ✓ | ✓ |
| Sinhala Entity Annotating | ✓ | - | - | ✓ | ✓ |
| ML Evaluation Tools | - | - | - | - | ✓ |
| Maintenance Fee | - | 2 Free + \$9.99 per additional call | - | - | - |
| Trial Duration | 1 Month | - | - | - | - |
| Package Price | Free | \$199.99 | \$9.99 | \$34.99 | \$49.99 |

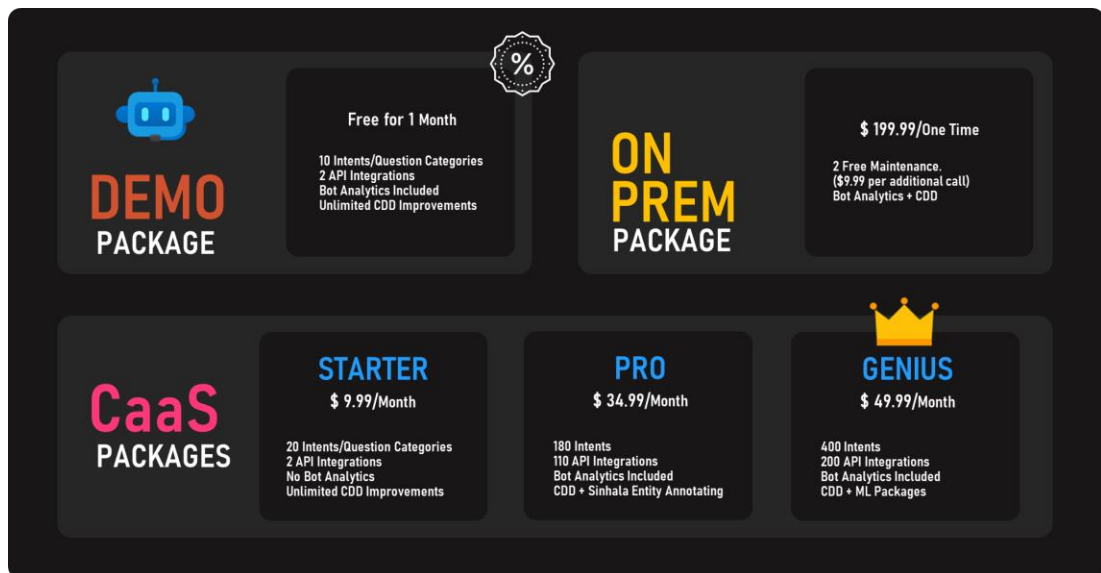


Figure 6.1: Commercialisation plan

7. BUDGET AND BUDGET JUSTIFICATION

The budget justification for all four research components is as shown in Table 7.1.

Table 7.1: Budget justification for the overall research project

| Component Name | Individual Item Price (LKR) | Number of Items | Duration | Total Item Price (LKR) |
|---|------------------------------------|------------------------|-----------------|-------------------------------|
| Domain Name | 2148.43/year | 1 | 1 year | 2148.43 |
| GCP Instance | 10683.84/month | 1 | 6 months | 64103.06 |
| Reference Book: Basaka Mahima by J.B. Dissanayake | 1250.00 | 1 | - | 1250.00 |
| Research Paper Publication | 25000.00 | 1 | - | 25000.00 |
| Grand Total | - | - | - | <u>92,501.49</u> |

REFERENCES

- [1]. ieee-dataport.org, 'How to Cite References: IEEE Documentation Style', [Online]. Available: <https://iee-dataport.org/sites/default/files/analysis/27/IEEE%20Citation%20Guidelines.pdf> [Accessed: 20-Jan-2022]
- [2]. Siddhant Meshram, Namit Naik, Megha VR, Tanmay More, Shubhangi Kharche, Conversational AI: Chatbots, Available: [Online]. <https://ieeexplore.ieee.org/document/9498508>
- [3]. A. M. Turing, 'computing machinery and intelligence', [Online]. Available: <https://academic.oup.com/mind/article/LIX/236/433/986238>
- [4]. Geoffrey Leech, Lancaster University, Developing Linguistic Corpora: a Guide to Good Practice, Available: [Online]. <https://users.ox.ac.uk/~martinw/dlc/chapter2.htm>
- [5]. viralzone.expasy.org, Human viruses table, Available: [Online]. <https://viralzone.expasy.org/678>
- [6]. Jason P.C. Chiu, Eric Nichols, 'Named Entity Recognition with Bidirectional LSTM-CNNs', [Online]. Available: [acl a 00104.pdf \(silverchair.com\)](https://acl.a00104.pdf(silverchair.com))
- [7]. Guillaume Lample, Miguel Ballesteros, Sandeep Subramanian, Kazuya Kawakami, Chris Dyer, 'Neural Architectures for Named Entity Recognition' [Online]. Available: <https://arxiv.org/pdf/1603.01360.pdf>
- [8]. Anastasia Zhukova, Felix Hamborg, Bela Gipp, 'ANEA: Automated (Named) Entity Annotation for German Domain-Specific Texts', [Online]. Available: <https://arxiv.org/pdf/2112.06724.pdf>
- [9]. Pontus Stenetorp, Sampo Pyysalo, Goran Topic, Tomoko Ohta, Sophia Ananiadou, and Jun'ichi Tsujii, 'BRAT: a Web-based Tool for NLP-Assisted Text Annotation', [Online]. Available: <https://aclanthology.org/E12-2021.pdf>

- [10]. Kalina Bontcheva, Hamish Cunningham, Ian Roberts, Angus Roberts, Valentin Tablan, Niraj Aswani, Genevieve Gorrell, ‘GATE Teamware: a web-based, collaborative text annotation framework’, [Online]. Available: <https://www.jstor.org/stable/42636386>
- [11]. Jie Yang, Yue Zhang, Linwei Li, Xingxuan Li, ‘YEDDA: A Lightweight Collaborative Text Span Annotation Tool’, [Online]. Available: <https://aclanthology.org/P18-4006.pdf>
- [12]. G. Thilini Weerasuriya , Supunmali Ahangama, Maheshi Nandathilaka, A Rule-based Lemmatizing Approach for Sinhala Language, [Online]. Available: https://www.researchgate.net/publication/333769052_A_Rule-based_Lemmatizing_Approach_for_Sinhala_Language
- [13]. J.B Dissanayake, Basaka mahima, ISBN: 9789556963656

APPENDICES

Appendix A: Survey Form

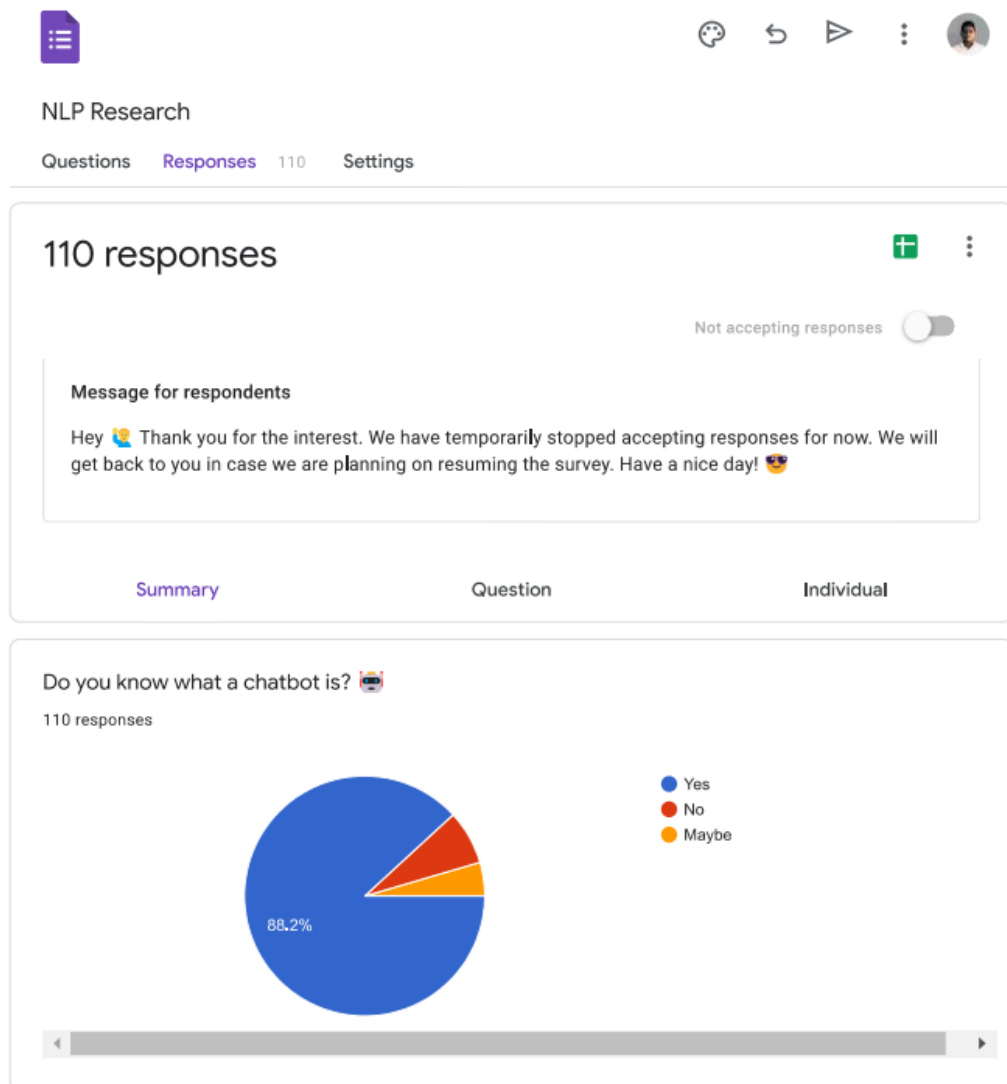


Figure A.1: Complete survey form questions and responses – part 1

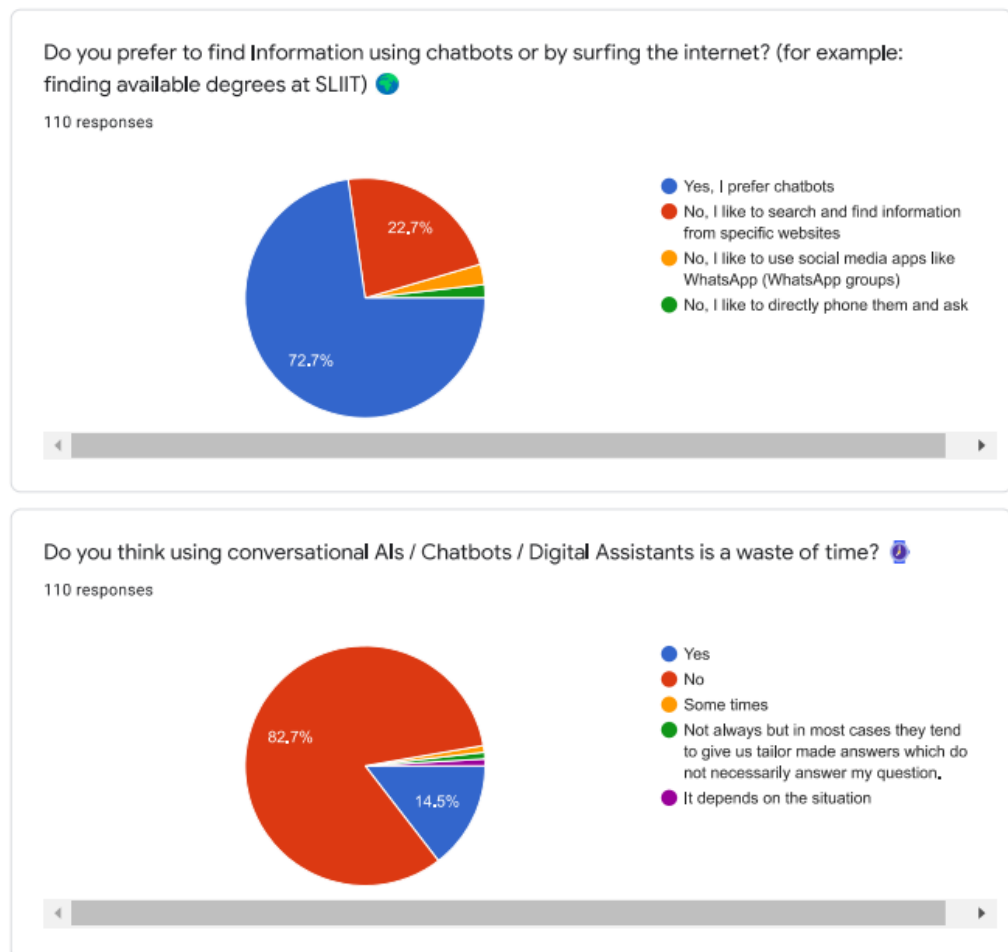


Figure A.2: Complete survey form questions and responses – part 2

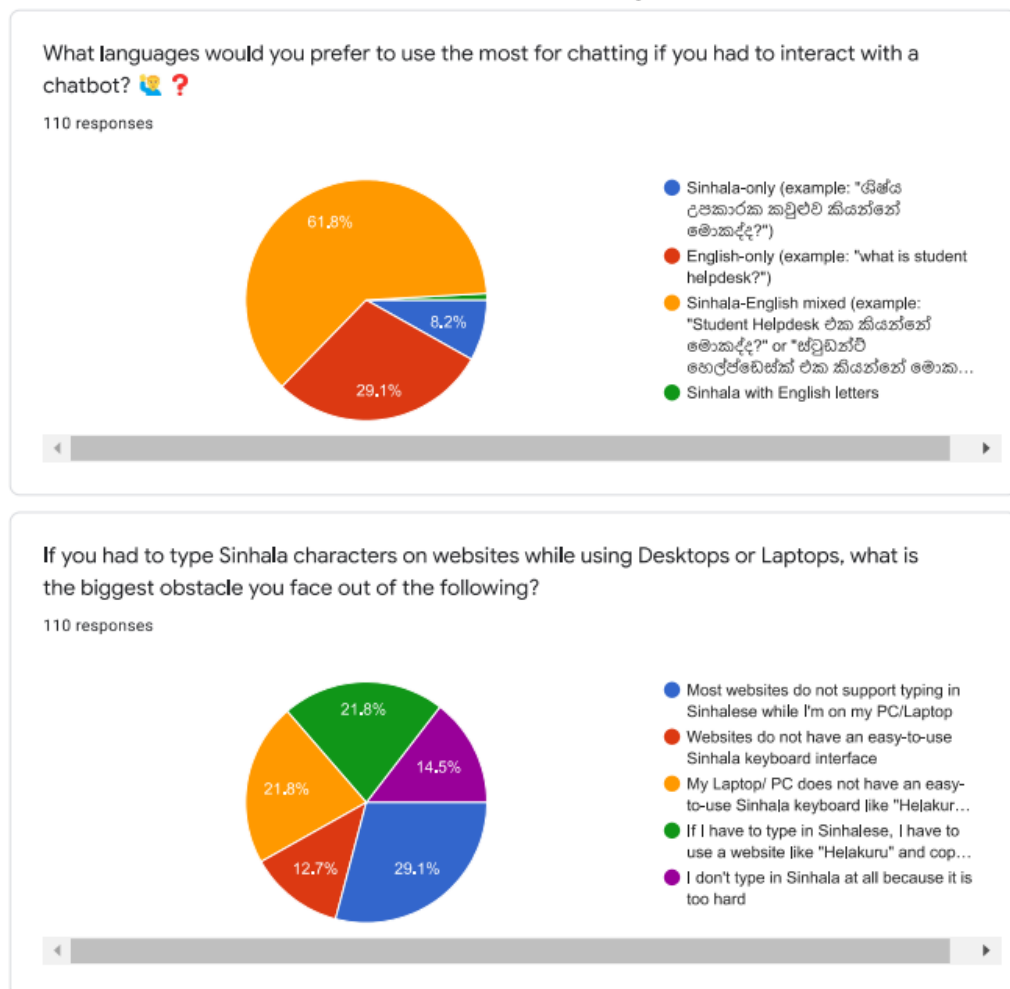


Figure A.3: Complete survey form questions and responses – part 3

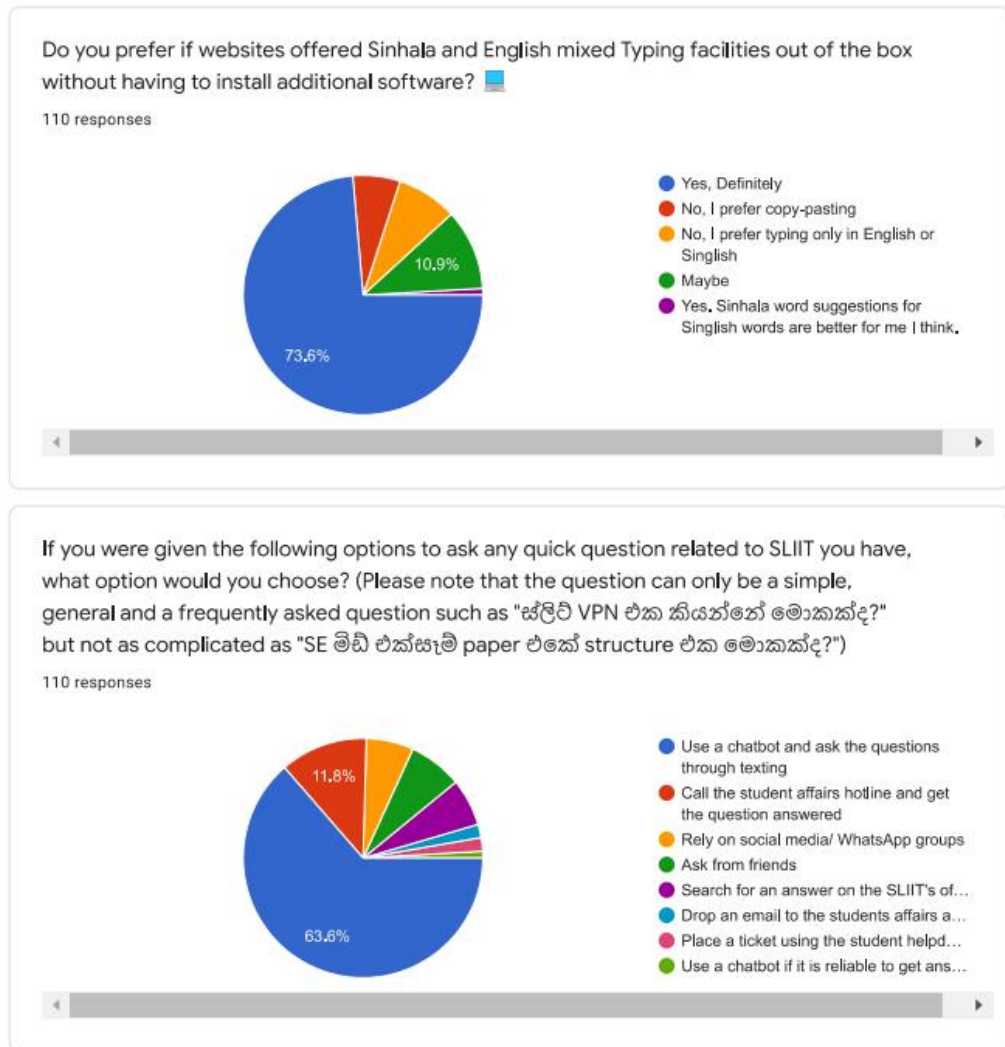


Figure A.4: Complete survey form questions and responses – part 4

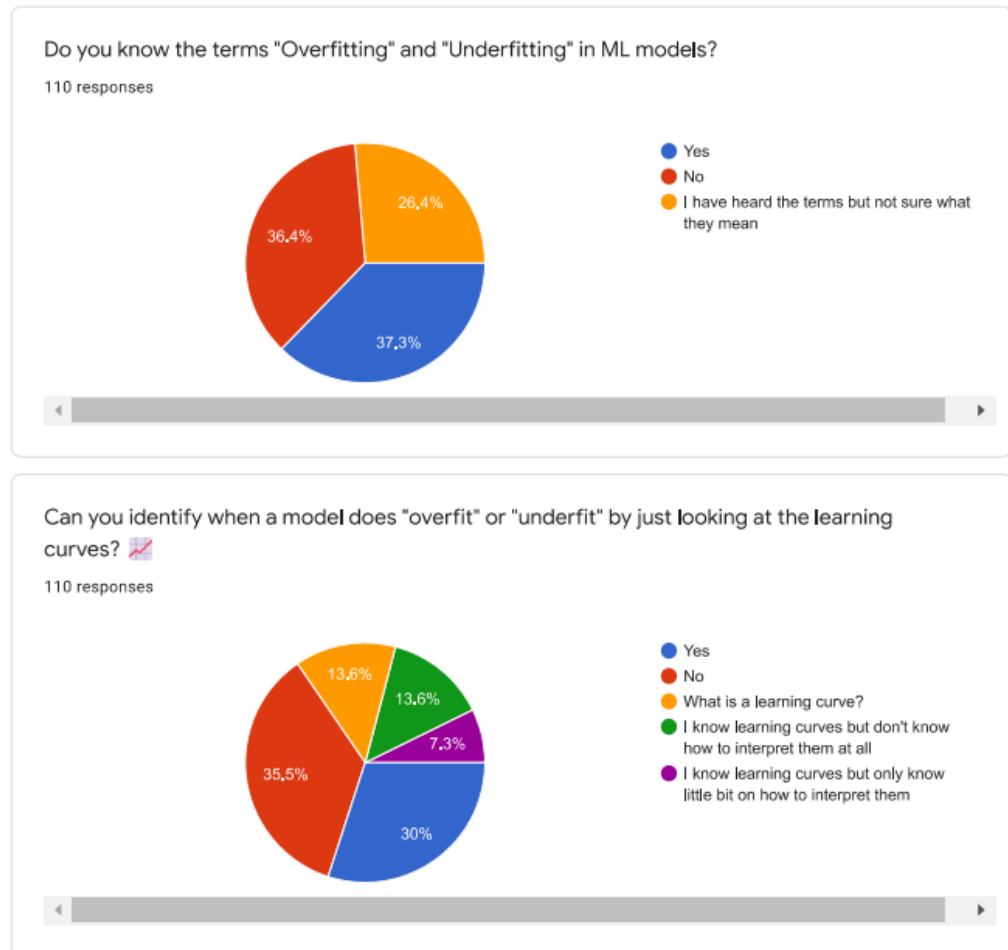


Figure A. 5: Complete survey form questions and responses – part 5

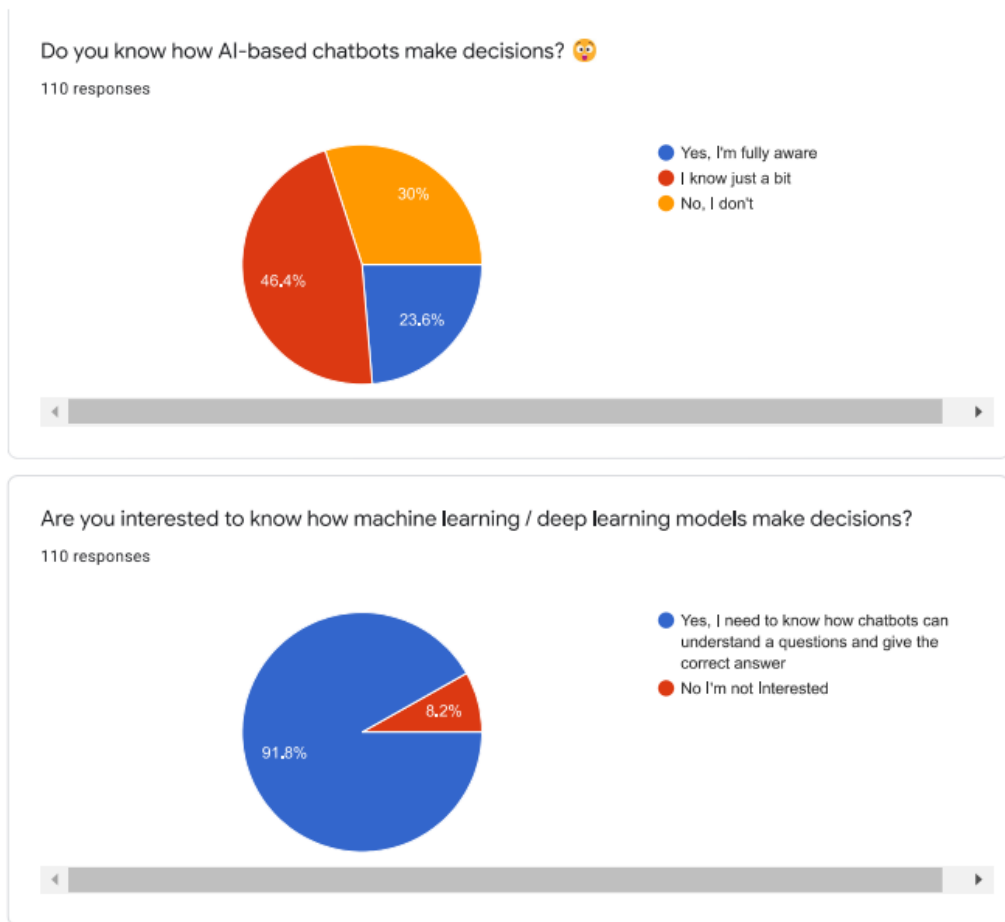


Figure A.6: Complete survey form questions and responses – part 6

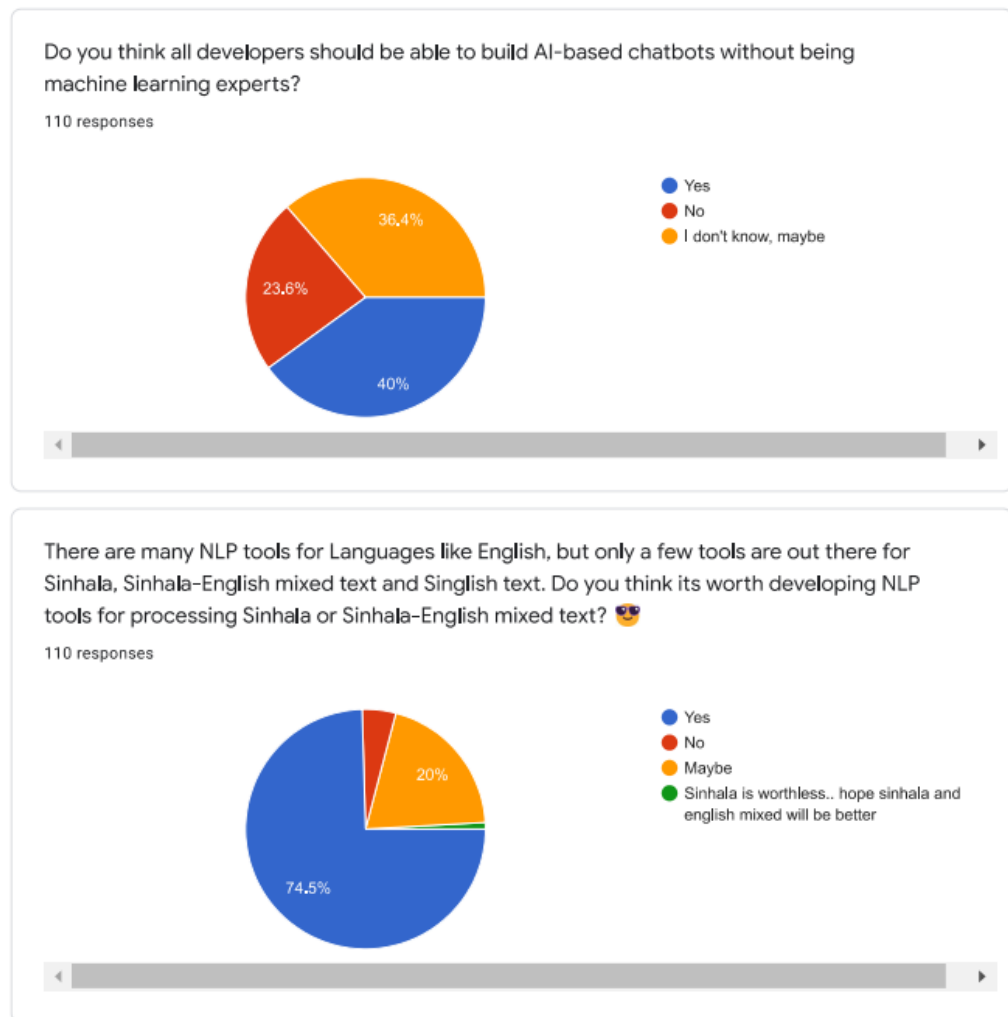


Figure A.7: Complete survey form questions and responses – part 7

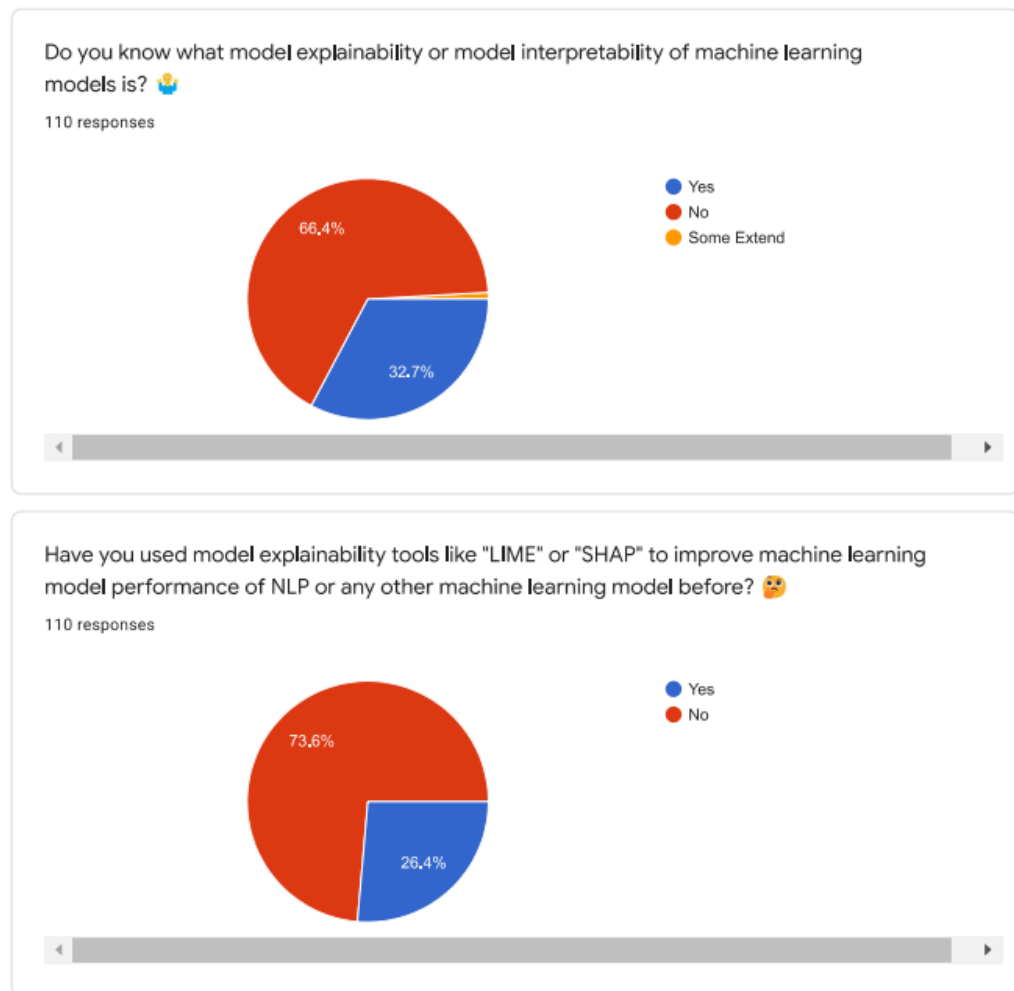


Figure A.8: Complete survey form questions and responses – part 8

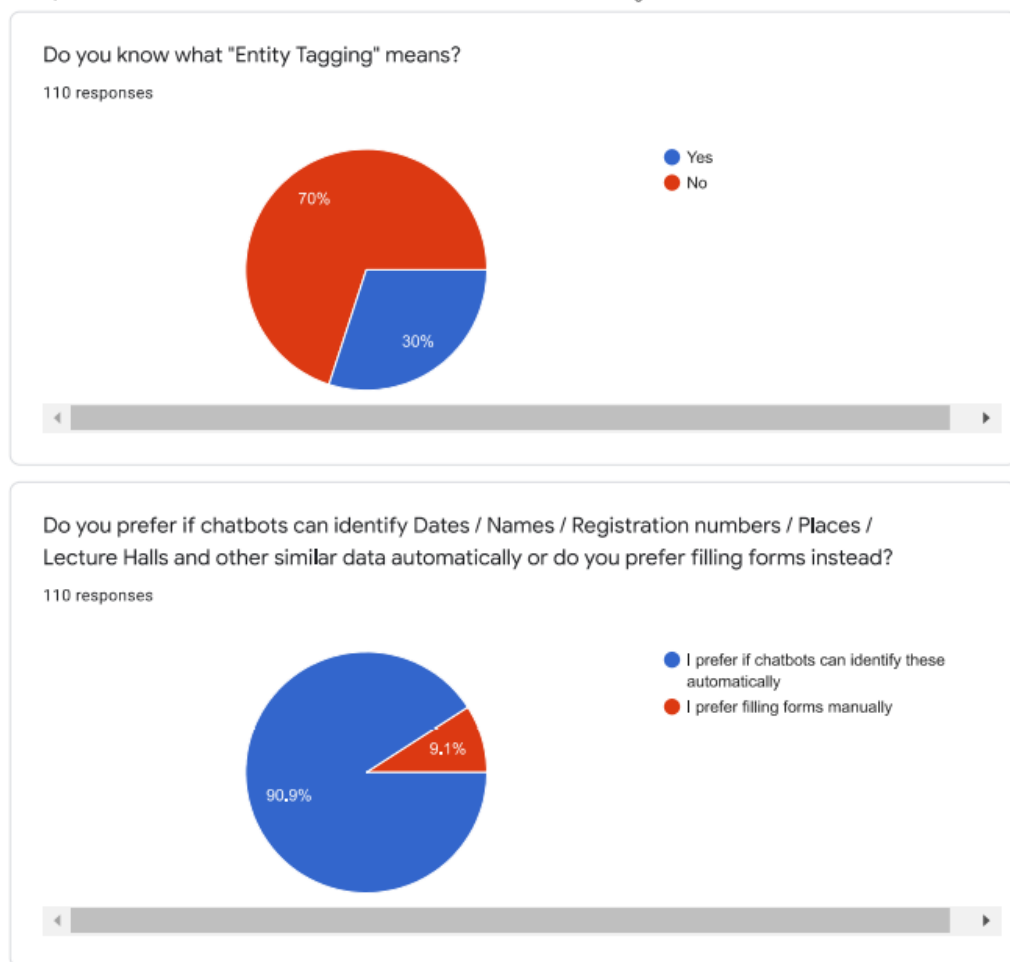


Figure A.9: Complete survey form questions and responses – part 9

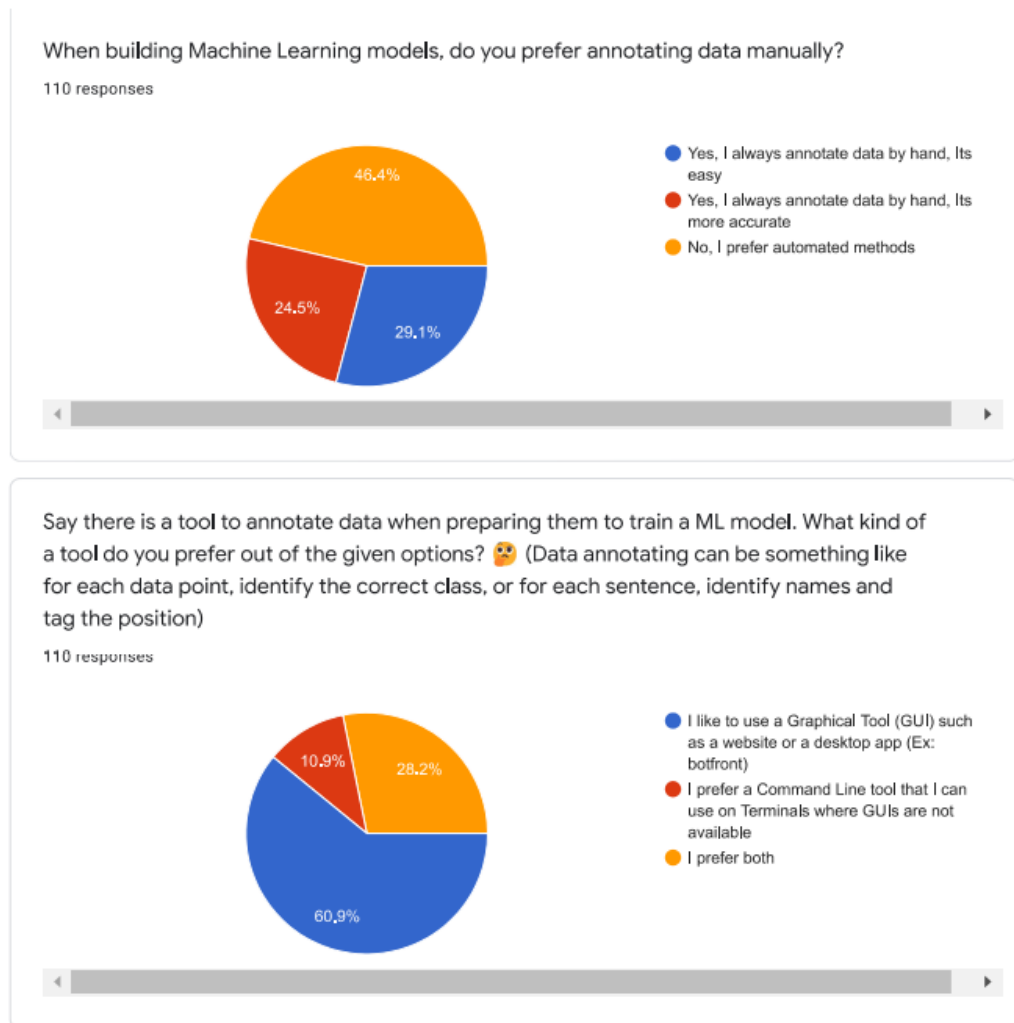
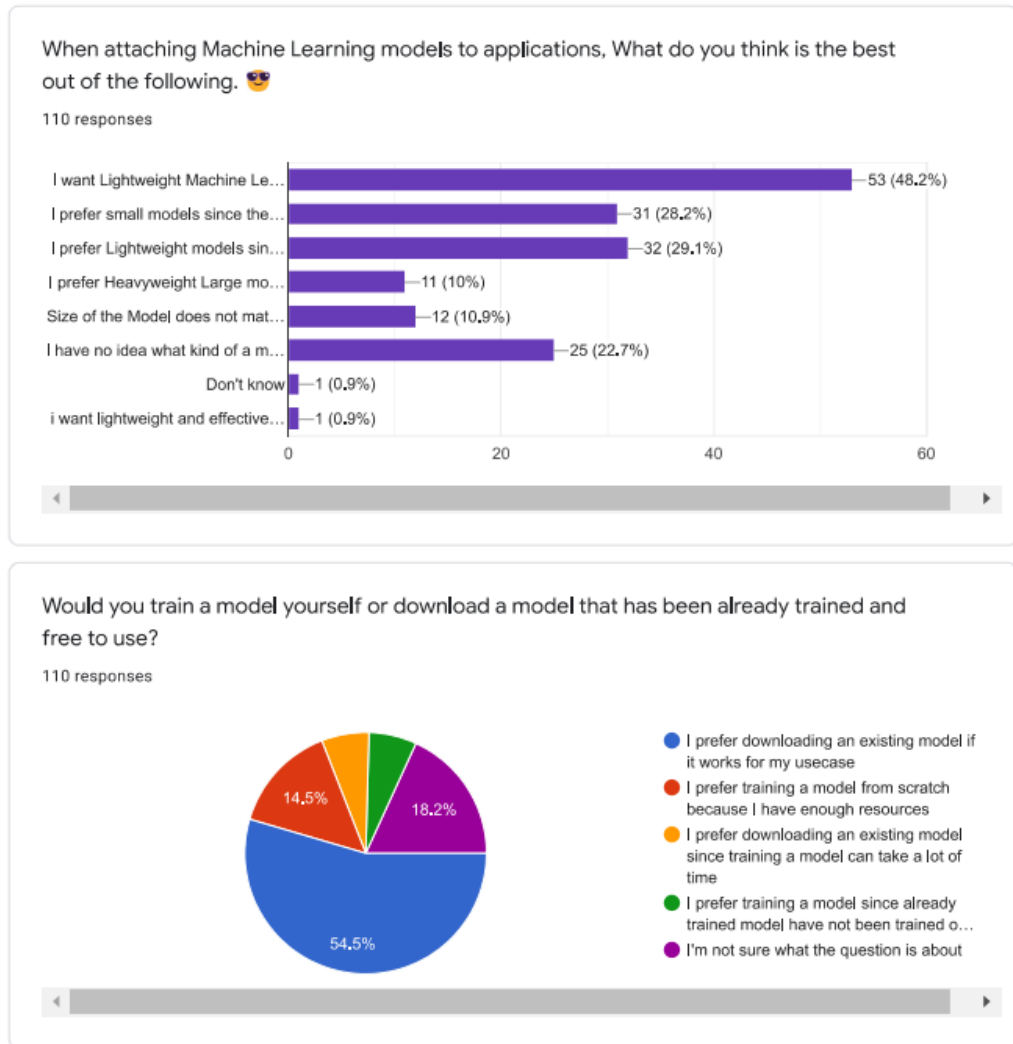


Figure A.10: Complete survey form questions and responses – part 10



Thank you! 🙏

Figure A.11: Complete survey form questions and responses – part 11

Appendix B: Supervision Confirmation Emails

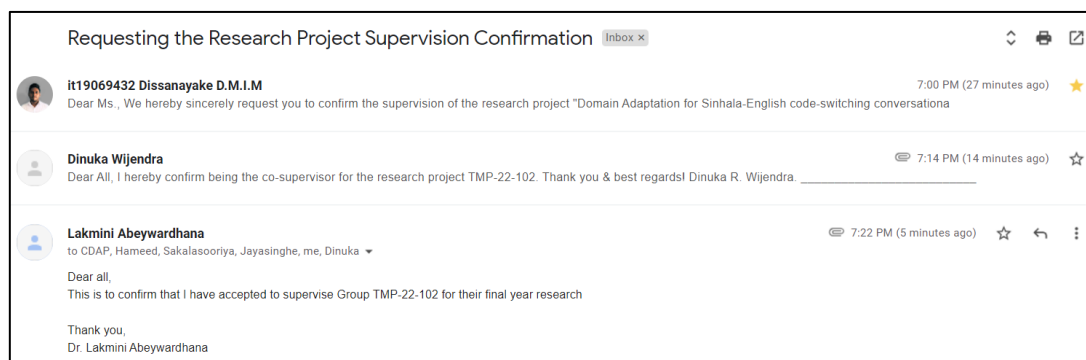


Figure B.1: Research project supervision confirmation email

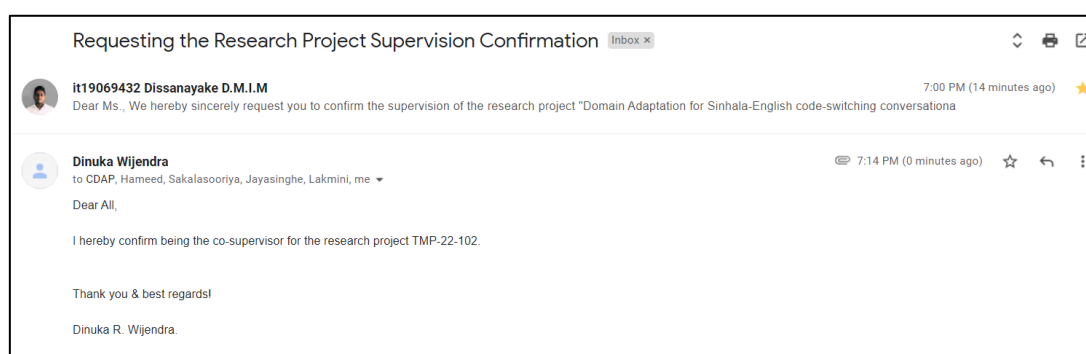


Figure B. 2: Research project co-supervision confirmation email