CUSTOMER SUPPORT CHATBOT WITH MACHINE LEARNING

A PROJECT REPORT

Submitted by,

Manda Kuswanth Sri Sai Syamala Rao	**-20211CSD0017**
Posa Anjan Kumar	**-20211CSD0009**
Swathi M	**-20211CSD0014**
Anusha A	**-20211CSD0057**

Under the guidance of,

Dr. Manjula H M

in partial fulfillment for the award of the degree of

BACHELOR OF TECHNOLOGY

IN

COMPUTER SCIENCE AND ENGINEERING (DATA SCIENCE)

At

GAIN MORE KNOWLEDGE
REACH GREATER HEIGHTS

PRESIDENCY SCHOOL OF COMPUTER SCIENCE AND ENGINEERING,

PRESIDENCY UNIVERSITY

BENGALURU

JANUARY 2025

PRESIDENCY UNIVERSITY

PRESIDENCY SCHOOL OF COMPUTER SCIENCE AND ENGINEERING

CERTIFICATE

This is to certify that the Project report **"CUSTOMER SUPPORT CHATBOT WITH MACHINE LEARNING"** being submitted by "MANDA KUSWANTH SRI SAI SYAMALA RAO, POSA ANJAN KUMAR, SWATHI M, ANUSHA A" bearing roll number(s) "20211CSD0017, 20211CSD0009, 20211CSD0014, 20211CSD0057" in partial fulfillment of the requirement for the award of the degree of Bachelor of Technology in Computer Science and Engineering is a bonafide work carried out under my supervision.

20/1/25

Dr. MANJULA H M
Associate Professor
Presidency School of CSE&IS
Presidency University

20/Jan/25

Dr. SAIRA BANU ATHAM
Professor & HoD
Presidency School of CSE&IS
Presidency University

Dr. L. SHAKKEERA
Associate Dean
Presidency School of CSE
Presidency University

Dr. MYDHILI NAIR
Associate Dean
Presidency School of CSE
Presidency University

Dr. SAMEERUDDIN KHAN
Pro-Vc School of Engineering
Dean – Presidency School of CSE&IS
Presidency University

PRESIDENCY UNIVERSITY

PRESIDENCY SCHOOL OF COMPUTER SCIENCE AND ENGINEERING

DECLARATION

We hereby declare that the work, which is being presented in the project report entitled **Customer Support Chatbot With Machine Learning** in partial fulfillment for the award of Degree of **Bachelor of Technology** in **Computer Science and Engineering (Data Science)**, is a record of our own investigations carried under the guidance of **Dr. Manjula H M, Associate Professor, Presidency School of Computer Science Engineering & Information Science, Presidency University, Bengaluru.**

We have not submitted the matter presented in this report anywhere for the award of any other Degree.

Name	Roll No	Signature
Manda Kuswanth Sri Sai Syamala Rao	20211CSD0017	M.K.S.S Syamala Rao
Posa Anjan Kumar	20211CSD0009	P. Anjan kumar
Swathi M	20211CSD0014	Swathi.M
Anusha A	20211CSD0057	AnushaA

ABSTRACT

We aim to bridge the gap between the customer and his services and the domain specific website by introducing a chatbot used for customer inquiries. The time customers spend paying attention to the website is reduced with this innovation, which is an act of putting the process of serving you, their customers, in front of their time. In this project, we wanted to build an intelligent chatbot that can fetch relevant information out of, identify different intents and do predefined actions. In order to create a contextual assistant for this objective, we used the RASA framework. To perform training for models, we generated a custom dataset with varying number of intents and entities. Along with this, we also created a couple Python scripts (RASA actions) which will run when their respective intents are detected. To our solution, we built a pipeline with a chatbot and several actions that the chatbot played out. The user's request will be processed such that these actions will interface with the database to fetch the required data or make any changes as needed in order to satisfy the user's requests, then the responses will be displayed using the chat widget.

Keywords: Intelligent Chatbot, RASA Framework, Customer Inquiries, Chatbot, Intent detection, Contextual Assistant, Custom Dataset, Entities, RASA Actions, Database Interface, Chat Widget, User requests, Predefined Actions.

ACKNOWLEDGEMENT

First of all, we indebted to the **GOD ALMIGHTY** for giving me an opportunity to excel in our efforts to complete this project on time.

We express our sincere thanks to our respected dean **Dr. Md. Sameeruddin Khan**, Pro-VC, School of Engineering and Dean, Presidency School of Computer Science Engineering & Information Science, Presidency University for getting us permission to undergo the project.

We express our heartfelt gratitude to our beloved Associate Deans **Dr. Shakkeera L and Dr. Mydhili Nair,** Presidency School of Computer Science Engineering & Information Science, Presidency University, and **Dr. Saira Banu Atham**, Head of the Department, Presidency School of Computer Science Engineering & Information Science, Presidency University, for rendering timely help in completing this project successfully.

We are greatly indebted to our guide **Dr. Manjula H M, Associate Professor** and Reviewer **Dr. Madhusudhan M V, Associate Professor**, Presidency School of Computer Science Engineering & Information Science, Presidency University for their inspirational guidance, and valuable suggestions and for providing us a chance to express our technical capabilities in every respect for the completion of the project work.

We would like to convey our gratitude and heartfelt thanks to the PIP2001 Capstone Project Coordinators **Dr. Sampath A K, Dr. Abdul Khadar A and Mr. Md Zia Ur Rahman,** department Project Coordinators **Dr. Manjula H M** and Git hub coordinator **Mr. Muthuraj.**

We thank our family and friends for the strong support and inspiration they have provided us in bringing out this project.

Manda Kuswanth Sri Sai Syamala Rao

Posa Anjan kumar

Swathi M

Anusha A

LIST OF TABLES

LIST OF FIGURES

TABLE OF CONTENTS

CHAPTER-1

INTRODUCTION

1.1 Introduction

At its core, a chatbot can be viewed as an entity that interacts users in a human-like manner, whether through voice or text. It comprehends user requests and delivers a tailored set of services that align with those needs. Additionally, chatbots serve as valuable tools for organizations to gather information, helping them gain a clearer understanding of their customer base and their requirements. They are employed across a variety of fields, ranging from healthcare to customer relationship management.

A chatbot generally consists of three main components: Natural language understanding (NLU) – understanding what the user is saying and their intention; dialog management – keeping track of the conversation, remembering key things the user says, and deciding how to drive the discussion forward; natural language generation (NLG) – what the system says to the user.

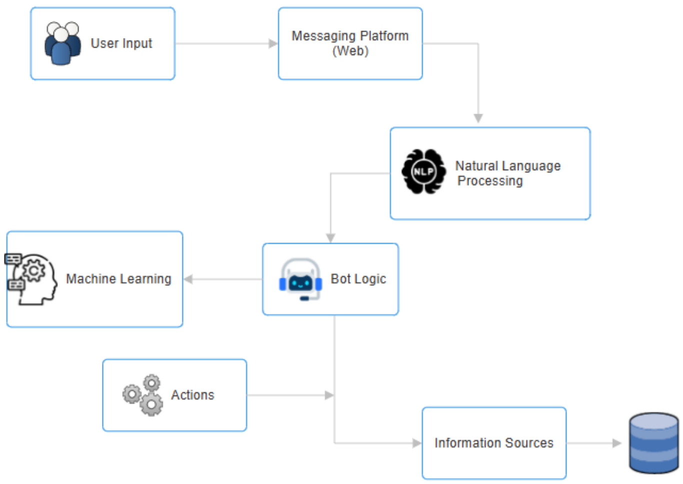


Figure 1.1: Chatbot workflow

RASA is an open source tool for Natural Language understanding (NLP) which we use to create virtual assistants and conversational chatbots. It includes two primary modules: And our open source Rasa and Rasa Action Server. Rasa Open Source has Rasa NLU for recognizing intents and Rasa Core for deciding how to respond to the intent. The hosting of Python scripts to be able to execute custom tasks such as processing or updating databases in the background are supported by Rasa Action Server.

1. **Environment Setup:**
   Install Conda and set up a new environment using Python 3.10.Activate this environment and install the appropriate version of Rasa (3.0).Confirm the installation to ensure everything functions correctly.
2. **Preparing Training Data:**
   Arrange the NLU training data in a file named (nlu.yml), which should include intents, examples, and entities.Create dialogue training data in a file called (stories.yml) to outline conversation flows.Define the bot's domain in the (domain.yml) file by detailing intents, entities, actions, responses, and slots.Specify the pipeline and policies in the (config.yml) file for the training process.
3. **Training the Models:**
   Independently train the NLU model to interpret user inputs and identify intents.Simultaneously train the dialogue model with the NLU model to manage conversations and foresee actions.
4. **Interactive Training:**
   Initiate an interactive session to polish the bot by trying out real-time conversations.Modify the training data based on feedback and save these changes to enhance precision.
5. **Running the Project:**
   Start the action server to enable any defined custom actions.Run the bot to engage with users through the command line or other interfaces.

1.2 Problem Statement : In the banking industry, customer support serves as a crucial point of contact for ensuring client satisfaction and trust. Conventional methods of customer support frequently lead to long wait times and restricted availability, which hampers the ability to meet customer needs effectively. As the of customer interactions and inquiries continues to rise, there is an urgent requirement for a smart, scalable, and effective solution.

The aim is to create a Machine Learning-driven chatbot specifically designed for the banking sector that can:

- **Respond to Frequently Asked Questions (FAQs):**
  Deliver immediate answers to common issues related to account management, interest rates, transactions, and services at branches.
- **Assess Loan Eligibility:**
  Evaluate a customer's loan eligibility using criteria such as income, credit score, and employment details, offering users a quick initial assessment.
- **Identify Messages as Spam or Legitimate:**
  Recognize and filter out spam messages to enhance the customer experience and secure communications.

This chatbot will enhance operational efficiency, decrease response times, and provide personalized assistance, ultimately improving customer satisfaction and loyalty. The implementation of such a system will not only optimize customer service but also build trust through the delivery of precise and timely information.

1.3 Methodology

Software development methodologies are methods of working with the project development. There are several years worth of methodological models that developers and users alike will have to distinguish what to apply to each scenario. We must optimize project efficiency and manage potential issues to reduce the odds of being encountered. Sticking to this principle also helps us define the scope of our project and also achieve our project goals. This is the only thing to do before completing a project; very basic yet, how is it possible to create a project without knowing the needs of the stakeholders? A methodology or a set of steps or process through which the raw data is converted to the well defined data patterns to extract the user insights.

Four Phases of the Spiral Model:

1. **Planning:** This phase involves gathering requirements and assessing risk management. During this time, we consult with the project manager regarding the project title. Needs and risks are addressed after conducting research on existing studies.
2. **Risk Analysis**: With this phase, we find out what possible risks are and possibly alternative solutions. During this stage, a prototype is developed at the end. Alternative solutions are proposed if any risks are found.

3. **Engineering:** This stage concentrates on the act of implementation of the model.
4. **Evaluation:** The final phase is a software evaluation of a system that has already been demonstrated, to confirm that it satisfies our expectations and also our requirements. Errors arising can be reported through the system.

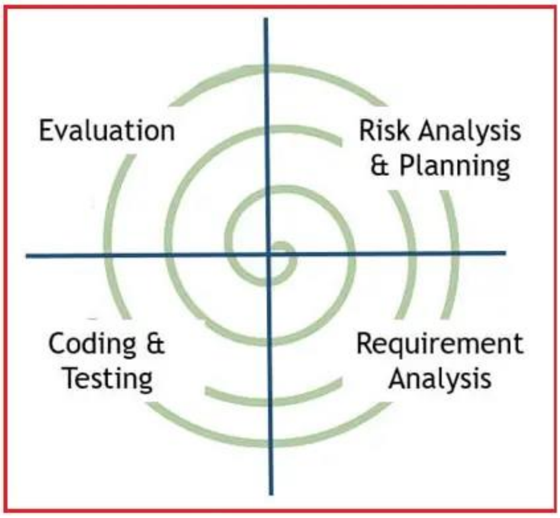


Figure 1.2: Spiral Model

1.3.1 Chatbot

Chatbots hold significant potential to deliver quick and straightforward support, directly addressing users' inquiries. The primary goal for many chatbot users is to enhance productivity, while others seek entertainment, social interaction, and engagement with new functionalities. To effectively achieve these objectives, a chatbot should function as a tool, a game, and a companion simultaneously. Businesses favor chatbots for their ability to lower customer service costs and manage multiple users simultaneously. They are now viewed not merely as helpful tools but as friendly companions, fostering closer connections with users.

Pattern matching is utilized to generate responses based on the user input. A user types in a query, and a corresponding result is produced. A major limitation of this approach is that the outcomes can become predictable, repetitive, and often lack adaptability. Additionally, it typically does not retain past responses, which can hinder meaningful interactions.

Natural Language Processing (NLP), a field within artificial intelligence, studies how computers can comprehend and manipulate human language text or speech. By understanding and applying human language, techniques are developed for computers to interpret and manage natural expressions effectively to execute desired actions. Most NLP methodologies rely on machine learning.

Natural Language Understanding (NLU) is at the heart of any NLP task. The technique also makes it possible to build natural user interfaces, such as chat bots. The NLU goal is to take a user input in unstructured natural language and interpret the context and meaning, without guessing, and to respond accordingly based on the user's intent. It understands the user objectives and picks the domain specific entities. In many ways, an intent is bridge (no pun intended) between user expression and the action the chatbot should take. These are the actions that a chatbot will perform when a defined specific intents are triggered by the user's query (with parameters for extra information). In general, intent detection is formulated as sentence classification wherein one or more intent labels are predicted for each input sentence.

1.3.2 RASA

RASA is an open-source machine learning framework for Python designed to automate text-based conversations and voicemail responses. It allows us to integrate chatbot workflows and automatically categorize user messages into their respective intents and entities. By utilizing the Rasa dialog flow pipeline, the chatbot executes various actions and responses (templates) to generate output. Rasa is built on top of Rasa NLU and Rasa Core, collectively referred to as Rasa Stack.

1.3.2.1 RASA ARCHITECTURE

The architecture of Rasa is structured as follows:

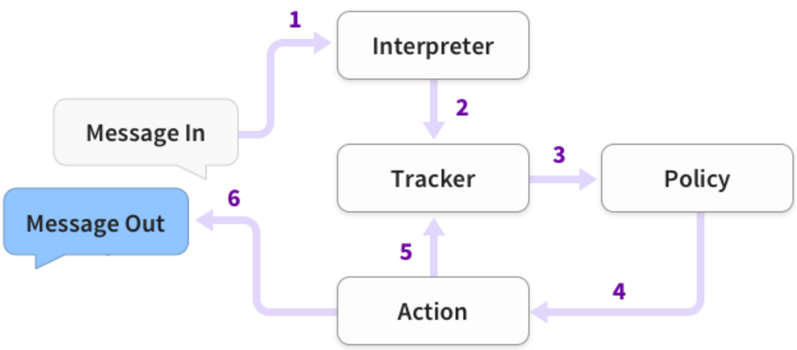


Figure 1.3: RASA Architecture

Incoming messages are received and sent to the interpreter, which translates them into a dictionary that captures the original text, the purpose, and any identified entities. This process occurs within the NLU component.

The tracker plays a crucial role in maintaining the chat environment by:

- Receiving details for new messages.
- Tracking the status of the conversation.
- Selecting the next action based on policies.
- Recording the selected action.
- Delivering a response to the user.

1.3.2.2 TRAINING RASA NLU:

Since the Rasa stack integrates NLU and Core components, Rasa NLU must be trained to understand anonymous user messages. We will need a Node.js server, Rasa X, or another suitable tool to accomplish this. When choosing Node.js, an application is necessary to create the training database. Note that this process generates training data in JSON format.

npm i -g rasa-nlu-trainer

1.3.2.3 PIPELINE IN RASA

A pipeline is essentially a series of procedures (known as Components in Rasa) executed sequentially in a queue. To define the pipeline, navigate to the project directory and update the config.yml file.

The components in the Rasa pipeline can include:

- **name: "Spacy NLP"**
- **name: "Spacy Tokenizer"**
- **name: "Spacy Featurizer"**
- **name: "Regex Featurizer"**
- **name: "CRF Entity Extractor"**
- **name: "Entity Synonym Mapper"**
- **name: "Sklearn Intent Classifier"**

It's also possible to add custom pipelines. The predefined pipeline in the Rasa NLU model is:
pipeline: "pretrained_embeddings_spacy"

Both pipelines achieve the same purpose, but if a custom component is necessary, the first option can be used without disrupting the other components.

Custom Pipelines in Rasa:

To implement custom pipelines in Rasa, it's important to understand the component lifecycle.

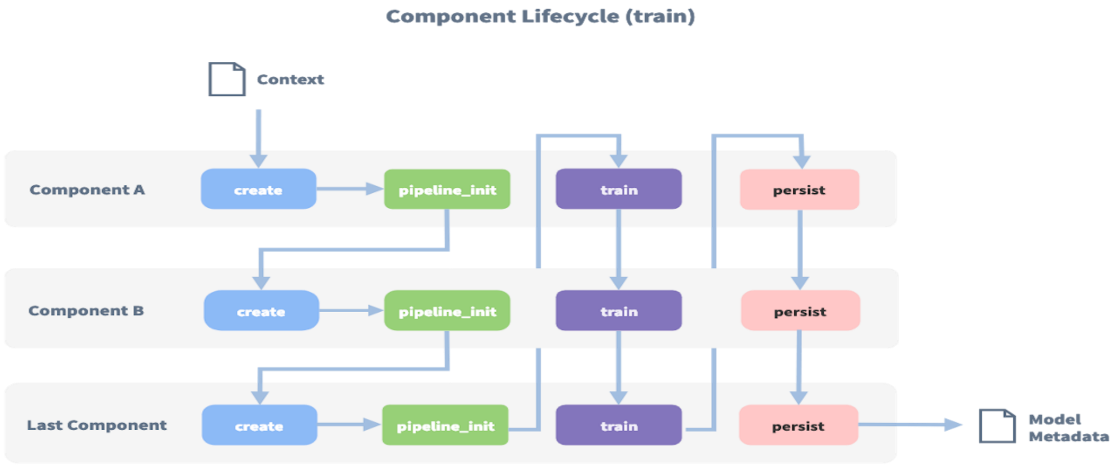


Figure 1.4: Component life cycle

This flowchart represents the Rasa NLU Pipeline, indicating where to insert custom items depending on what is needed before and after releasing each pipeline.

1.3.2.4 RASA CORE

Rasa Core serves as the dialog flow model within Rasa and manages the flow of conversations. To effectively operate, the Core model requires training using specific data files.

Stories: Stories outline the paths that a chatbot follows during interactions, driven by the intent recognized.

Domain: This component encompasses all the information necessary for the Rasa Core Model, ranging from intents to actions.

Intents:

<!-- Intents Declaration -->

- greet

- goodbye

- affirm

- deny

- mood_great

- mood_unhappy

Example of greet text:

text: "Hey! How are you?"

1.3.2.5 ACTIONS IN RASA

Actions consist of Python code executed upon being called. Templates are a subset of actions, but not all actions are templates. Here are a few built-in actions provided by Rasa Core:

1. action_listen
2. action_default_fallback
3. action_restart

Utterance Actions are static responses that begin with the prefix utter_. Example:
utter_greet

1.3.3 Flask

Flask is a micro web framework in Python, and hence is a micro one since it does not require any specific tools or libraries. It has no database abstraction layer component , and accepts database as an optional component , for another third party library which implements standard forms validation, authentication, etc. However, since this is a library, it’s possible to extend it with extensions that are recognizably as part of Flask. There are object relational mapping extensions, form validation extensions, file upload handling extensions, ideally across open authentication protocols and a host of common tools within your chosen framework extensions. Applications built with Flask include Pinterest and LinkedIn.

Flask is a lightweight Python web framework and offers a wide range of utilities and consciousnesses for developing your own web application in Python. Using it, developers are flexible and can quickly step their way into what it is capable of while beginners can start with just one Python file to quickly build web apps.

1.3.4 Chat Widget

A chat widget is a feature which is incorporated into a website, this enables visitors to chat with sales or service providers on real time basis. This is a chat bubble that usually appears in the bottom right corner of a webpage, which tells users to chat with a company representative. Whilst staffed by live agents, chatbots are now outranking many business conversations opting to be the preferred route to offer businesses which require expanded chat functionality or around the clock assistance.

Live chat widgets are common on websites, but can be included in social media platforms and even on mobile apps.

Half of all customers said they prefer to chat online with a company rather than using alternatives such as email, phone support and social media. It's an immediate response capability of chat widgets that allows visitors to connect with a company just as they have a question.

CHAPTER-2

LITERATURE REVIEW

T. Bocklisch et al. [1] discusses Rasa, an open-source framework that can be used to develop conversational artificial intelligence. It comprises Rasa NLU, which is responsible for determining the user’s intent and extracting relevant information, and Rasa Core which is responsible for the direction of the interaction and staff actions expectations. Developers are also able to teach the system by providing feedback within Rasa. The authors demonstrate how to use Rasa in a stock financing chatbot, emphasizing its usability as chatbots contextually-oriented; however, the system requires work with complex conversations and multiple entities.

S. Johnson et al. [2] an intelligent chatbot was introduced to address the domain of wealth management, which can be sophisticated in terms of the complexity of its core concept under the changing financial paradigm. The bot has the capability to converse with clients about investment, taxation planning and risk analysis among others using cutting edge stuffs being developed in the field of Natural Language Processing (NLP) and Artificial Intelligence (AI). Thus, it is machine learning (ML) that enables it to determine user intents, parse information for its use, and provide custom recommendations. The chatbot aims to improve the existing traditional forms of wealth management services. This tool focuses on the client, the areas considered are high efficiency and low cost.

Frommert et al. [3] explores the underused potential of enterprise social networks (ESNs) to improve knowledge sharing and communication in cooperative networks. Even though these systems are interconnected, there are still no efficient and user-friendly methods for using the data across these networks. Chatbots are a practical solution to this issue, despite their primary application in consumer communication. Not enough research has been done on their advantages in cooperative networks. This study reviews existing chatbot technologies and proposes a prototype to show how chatbots may intelligently use data from company social networks to significantly improve corporate communication.

Jiao et al. [4] discusses the method of developing a chatbot that uses RASA NLU to comprehend and process user input is described in detail in the paper, with an emphasis on entity extraction, or locating important textual elements (such names, dates, and locations).

By strengthening the model's capacity to identify and comprehend different kinds of things in a range of conversational scenarios, the author explains how neural networks improve the extractionprocess.

This study emphasises how crucial precise entity extraction is to building chatbots that can manage increasingly intricate and dynamic exchanges , Neural network integration with RASA NLU is highlighted as a way to manage ambiguity and variances in customer enquiries, enhancing the chatbot system's resilience and effectiveness.

S. F. Suhel et al. [5] provide a strong case for the employment of artificial machine intelligence (AMI) in banking to improve customer service in the area of chatbot applications for automation. According to their research, chatbots can automate standard banking functions like managing transaction histories and checking balances, increasing productivity and cutting expenses. Developing intelligent chatbots for service automation, especially in high-traffic industries like banking and finance, is the goal of my research. Their study offers recommendations on enhancing automation in customer support applications and offers insightful information about how machine learning and natural language processing techniques can be used to address complex client queries.

Fathima et al. [6] provides the use of artificial machine intelligence (AMI) in the banking industry to automate customer care tasks via chatbots is examined by Fathima et al. (2020). In order to reduce the need for human agents, the authors demonstrate how chatbots, which are fuelled by AMI and NLP, can manage a variety of banking functions, including account management and transaction queries. My study on automating customer support in high-demand industries like banking, where prompt, real-time replies are crucial, is especially pertinent to this paper. As evidenced by their work, the incorporation of AMI into chatbot systems offers important insights into expanding chatbot skills for managing more complex queries and boosting customer service effectiveness.

Singh et al. [8] examine how AI-powered chatbots are being used in Indian banks, emphasising how they might enhance customer service and operational effectiveness. The study focusses on how chatbots can perform a range of banking duties, including processing transactions and responding to client enquiries, thanks to machine learning and natural language processing (NLP). This article offers a case study of how artificial intelligence (AI) can improve client interactions in the banking industry, which is pertinent to my research on chatbot implementation in financial services. The difficulties noted, like data protection and language assistance, will also guide the creation of my own chatbot system, which will handle related problems in various settings.

Young T et al. [9] development of deep learning based methods for natural language processing (NLP): its important consequences for banking among other industries. In the study, we highlight ways of enhancing human language processing and comprehension using deep learning models, such as Convolutional Neural Networks (CNNs), Long Short Term Memory (LSTM) networks, and Recurrent Neural Networks (RNNs). These models enable chatbots, virtual assistants, and automated transaction systems to act as the face of the banking industry and power conversation across a broad spectrum of client enquiries and activities.

A. Singh et al. [10] provides an extensive review of the three main chatbot creation platforms—Google Dialogflow, Rasa, and Microsoft Bot Framework—is given by Singh, Ramasubramanian, and Shivam (2019). Their study evaluates these platforms' usability, adaptability, and integration potential, providing insightful information for choosing the best platform for creating business chatbots.

CHAPTER-3
RESEARCH GAPS OF EXISTING METHODS

Rasa Framework for Dialogue Management

Focus: Explores the features of the Rasa framework for open-source language comprehension and dialogue management.

Gap: Currently, there is a lack of deep discussion regarding how the input of advanced deep learning techniques (e.g., transformer models) can be combined with the Rasa pipeline to enhance language understanding of complex scenarios and context preservation.[1]

Chat Finance and Conversational Bots for Financial Tracking

Focus: Creating a financial chatbot equipped with basic wealth tracking capabilities.

Gap: There is inadequate emphasis on addressing multilingual financial conversations, reducing biases in financial advice, and connecting with real-time market data for up-to-date insights. [2]

Chatbots in Collaborative Networks

Focus: The application of chatbots to support communication within collaborative networks.

Gap: There is a deficiency in thoroughly assessing chatbot scalability within large and dynamic collaborative environments, alongside their ability to adapt to various network structures and roles. [3]

Entity Extraction with Rasa NLU

Focus: The development of smart chatbots utilizing entity extraction and neural networks.Gap: There is limited investigation into unsupervised or self-supervised methods to lessen the reliance on data for training entity recognition systems. [4]

Banking Chatbots with Automation [5] and [6]

Focus: Utilizing artificial intelligence to automate customer interactions in banking.

Gap: There is a lack of emphasis on improving fraud detection, managing regulatory compliance, and considering the ethical aspects of AI decision-making in banking services.

Rasa Documentation

Focus: Provides thorough documentation for creating chatbots using Rasa

.Gap: There is a need for integration guides that incorporate cutting-edge technologies like GPT-based models, better context-switching techniques, and tailored solutions for specialized industries. [7]

Chatbots in Indian Banks

Focus: Investigates the use and implementation of chatbots within the context of Indian banking.

Gap: There is a missed opportunity to discuss cross-cultural adaptation and personalizing services for a diverse clientele in a multilingual country. [8]

Trends in NLP with Deep Learning

Focus: Reviews recent developments in deep learning-driven NLP techniques.

Gap: There is a scarce focus on applying these advancements directly to conversational AI systems for specific fields, such as healthcare or legal technology. [9]

Enterprise Chatbot Development

Focus: Introduces widely-used frameworks like Microsoft Bot, Rasa, and Dialogflow.

Gap: There is limited comparative analysis of real-world effectiveness, user satisfaction levels, and the adaptability of custom workflows among these frameworks. [10]

CHAPTER-4
OBJECTIVES

- Create a strong customer support chatbot for the banking sector using the RASA framework, integrating machine learning for precise intent classification and entity recognition.
- Offer responses to common inquiries (FAQs) regarding banking services, including account setup, card blocking, and balance checks.
- Establish a system for assessing loan eligibility based on user information such as salary, gender, and other financial factors.
- Categorize user messages as spam or legitimate to maintain secure and effective communication channels.
- Ensure smooth integration with web interfaces for easy access and improved user experience.
- Employ a flexible knowledge base that updates in real-time to enhance the accuracy and relevance of responses.
- Integrate feedback systems for ongoing learning and enhancement of the chatbot over time.

CHAPTER-5

PROPOSED METHODOLOGY

The proposed methodology for creating a customer support chatbot for banking services utilizing the RASA Framework consists of several important stages to guarantee functionality, scalability, and efficiency.During the Requirements Analysis and System Design phase, the chatbot’s objectives are established. These objectives include answering common banking inquiries, such as how to block a card or set up an account, evaluating loan eligibility based on user information like income and credit history, and differentiating messages between spam and non-spam to identify phishing attempts. The system design specifies an architecture that uses RASA’s NLU and Core components for intent recognition and dialogue management, along with tailored machine learning models for spam classification and loan eligibility assessment. Furthermore, the chatbot will be linked to a web interface to improve user engagement.

The subsequent phase, Data Collection and Preparation, centers on accumulating and structuring datasets. A thorough FAQ dataset is compiled by gathering typical banking-related questions and answers, organizing them into intents, entities, and response schemas. Loan eligibility data is derived from publicly accessible datasets or generated simulations, incorporating essential factors like income, employment status, and credit score. For spam detection, a dataset like the SMS Spam Collection is utilized and supplemented with banking-specific examples to enhance its relevance. All datasets are labeled to aid the training of the natural language understanding (NLU) component and classification models.

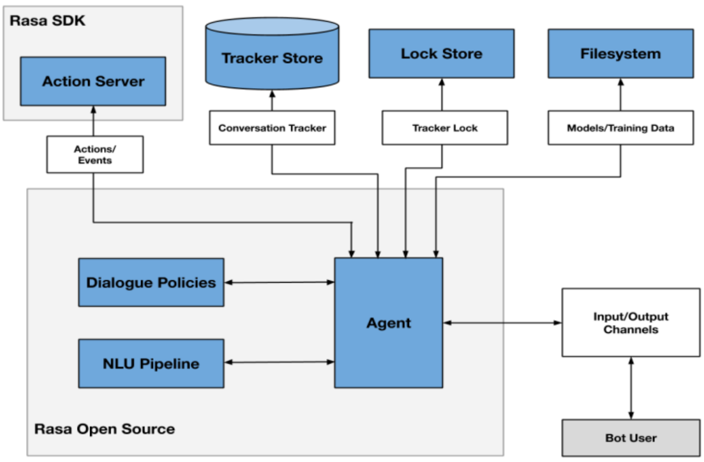


Figure 5.1: Proposed architecture

In the Chatbot Development with RASA stage, the NLU component is trained to recognize various intents, such as card blocking and account creation, and to identify entities like account type or branch location using tools like spaCy or the DIET Classifier. The management of dialogue is performed by establishing multi-turn stories that include fallback and clarification options. Policies such as RulePolicy for fixed responses and FormPolicy for dynamic loan eligibility inquiries are established. Custom actions are implemented in Python to carry out tasks such as loan eligibility forecasting via a machine learning model and spam detection using a trained classifier.

The Machine Learning Model Development phase includes the creation of two essential models. The loan eligibility prediction model is developed using methods such as logistic regression or decision trees, and the final model is either made accessible through an API or directly integrated into the RASA action server. The spam classification model is constructed using techniques like Naive Bayes, SVM, or transformer-based models such as BERT, which are fine-tuned to identify banking-related spam and phishing attempts.

Integration with the Web Interface guarantees smooth user interaction. A responsive front-end is crafted using frameworks like React or Angular, featuring a chat widget for real-time communication. On the back-end, the chatbot server is deployed with Flask, FastAPI, or Django, with endpoints that facilitate interaction between the chatbot and the web interface. APIs are also established to allow for database interactions and integrate third-party services when necessary.

The Testing and Validation phase assesses the chatbot's performance through functional testing. The accuracy and relevance of FAQ responses are evaluated, loan eligibility predictions are checked against test scenarios, and the spam detection model is tested on new messages to ensure its efficacy.

Finally, in the Deployment and Maintenance phase, the chatbot is launched on a scalable cloud platform like AWS and packaged using Docker for efficient deployment. After deployment, the system is maintained with regular updates to the knowledge base, retraining of models with new data, and monitoring logs for performance insights and issue resolution.

CHAPTER-6
SYSTEM DESIGN & IMPLEMENTATION

6.1 SYSTEM DESIGN

The design of the system incorporates a layered architecture that supports user interaction with the chatbot, facilitating functions such as handling FAQs, detecting spam, and assessing loan eligibility. It consists of the following components:

User Interface Layer

A web-based platform or mobile application enables users to engage with the chatbot through natural language. This interface delivers user inquiries to the backend for further processing.

Processing Layer

Built on top of RASA Framework, which includes RASA NLU for understanding natural language, and RASA CORE for managing the dialogue flow, this layer is. The features include spam/ham classification, intent detection and evaluation of loan eligibility. Machine learning models for spam identification and a loan prediction are enabled during query handling.

Knowledge Base Layer

The way is a database layer with information regarding loan eligibility and other predefined data, loans and policies. The chatbot is made to provide accurate information, so the database is updated in real-time.

6.2 Workflow

The user initiates a query through the web interface.

The query is routed to RASA NLU for both intent classification and entity extraction. The processing of the query depends on:

- **FAQs**: The system retrieves appropriate responses from the knowledge base.
- **Loan Eligibility:** User-supplied parameters are input into a trained machine learning model for evaluation.
- **Spam Detection:** A text classification model is used to discern between spam and non-spam messages.

RASA CORE oversees dialogue management to maintain logical conversation flow. The resulting response is then sent back to the user through the user interface.

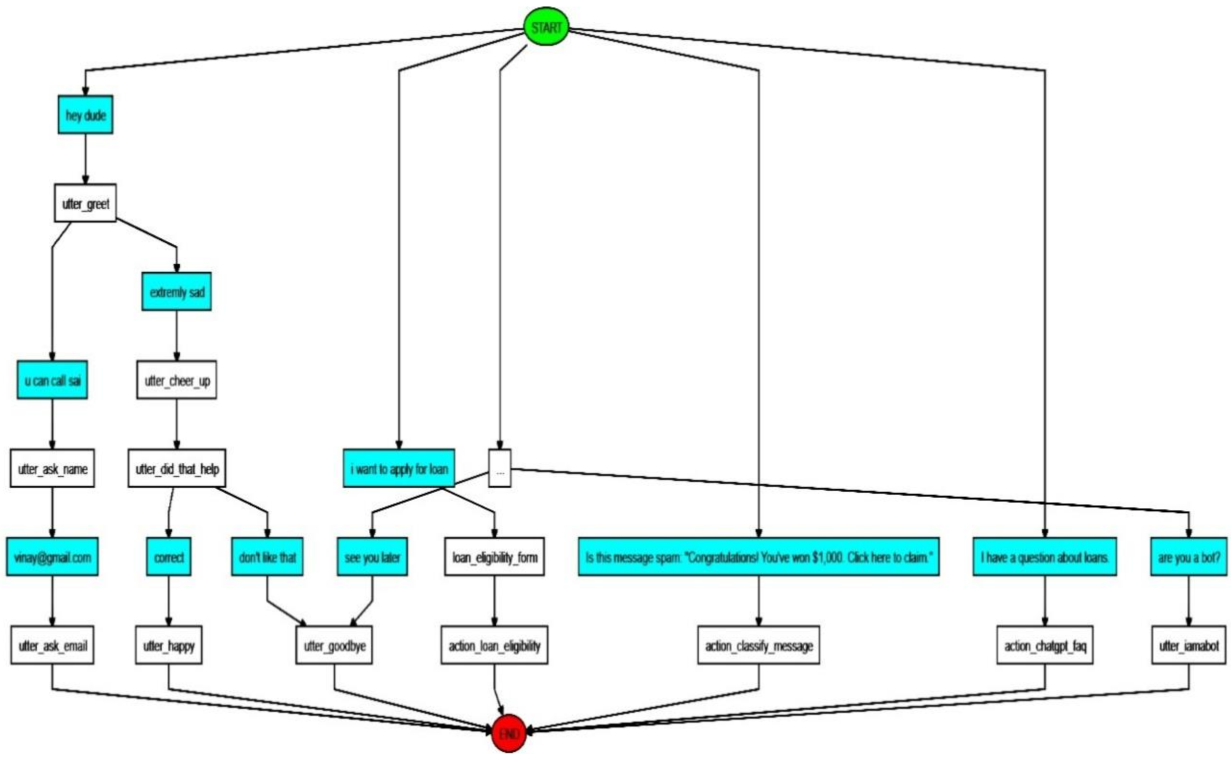


Figure 6.1: Flow chart

6.3 IMPLEMENTATION

6.3.1 SOFTWARE AND HARDWARE REQUIREMENTS

HARDWARE REQUIREMENTS

The hardware requirements align with the minimum specifications necessary for the successful development and testing of the project, as detailed in Table 6.1

Table 6.1: System Hardware Level Configurations

Processor	**i3/i5 (6th generation or newer)**
RAM	**Minimum 8GB**
System Type	**64-bit OS, x64-based processor**

SOFTWARE REQUIREMENTS

- **PYCHARM / ANACONDA: for Python 3**
- **TensorFlow**

- **Ngrok**
- **Git**
- **Visual Studio Code**
- **RASA framework**

6.3.2 ASSUMPTIONS AND DEPENDENCIES

The project operates under the assumption that the chatbot can efficiently handle all inquiries related to the college. There are no extra dependencies required beyond the Python packages listed above for the development of this project.

6.3.3 IMPLEMENTATION DETAILS

Custom Actions and Model Integration:

Set up the RASA action server to handle specific tasks such as assessing loan eligibility and identifying spam, leveraging pre-trained machine learning models.

Intent and Entity Recognition:

Employed the DIET classifier in RASA to achieve precise intent recognition and entity identification, thereby improving the chatbot's understanding of natural language.

External API Integration:

Connected APIs to retrieve up-to-date banking information (such as account details and loan rates) and facilitate notifications through SMS or email.

Data Management:

Established a database to keep track of user interactions, context, and feedback, promoting personalized and context-aware dialogues.

Deployment and Scalability:

Launched the chatbot on a cloud service using Docker, allowing for scalability and dependable performance during peak usage times.

Testing and Feedback:

Performed extensive testing to ensure precision and reliability, and established a feedback system to continuously enhance the chatbot based on user responses.

6.3.3.1 SNAPSHOTS OF INTERFACES

Setting Up RASA

After installing Anaconda, we initiate a virtual environment:

```
conda create -n rasa_env python=3.10 -y
conda activate rasa_env
pip install rasa
pip install rasa-sdk
```

Figure 6.2: Virtual Environment

CLI commands

```
rasa init
rasa train
rasa interact
rasa shell
rasa run
rasa run actions
rasa visualize
rasa test
rasa data split nlu
rasa data convert
rasa data validate
rasa export
```

Figure 6.3: CLI Commands

Here, we establish a new RASA project that includes example training data, actions, and configuration files. We proceed by training and saving our model with our NLU data and conversation stories. Next, we conduct an interactive learning session to generate fresh training data by conversing with our assistant through the interactive command. The trained model is loaded in the command line using the rasa shell command. Subsequently, we start both a server and an action server with the Rasa SDK. A general visual representation of stories is created using the visualize command. Finally, we carry out testing, data splitting, validation, conversion, and exportation of the training data.

```
├── actions
 |    ├── __init__.py
 |    └── actions.py
 ├── config.yml
 ├── credentials.yml
 ├── data
 |    ├── nlu.yml
 |    └── stories.yml
 ├── domain.yml
 ├── endpoints.yml
 ├── models
 |    └── <timestamp>.tar.gz
 └── tests
     └── test_stories.yml
```

Figure 6.4: Basic File Structure from Rasa Init

Major Files

1. nlu.yml

We provide example messages which the assistant could learn from, so user input can be recognized by the assistant regardless of how they are phrased. Grouping these examples by their intent, the backward goal of what the message is intended to accomplish. The code block below shows that we created an intent whose names are things like 'Hi', 'hey', and 'Good morning' under the name 'greet'. These are the intents and their examples that the assistant absorbs as training data of its Natural Language Understanding(NLU)model.

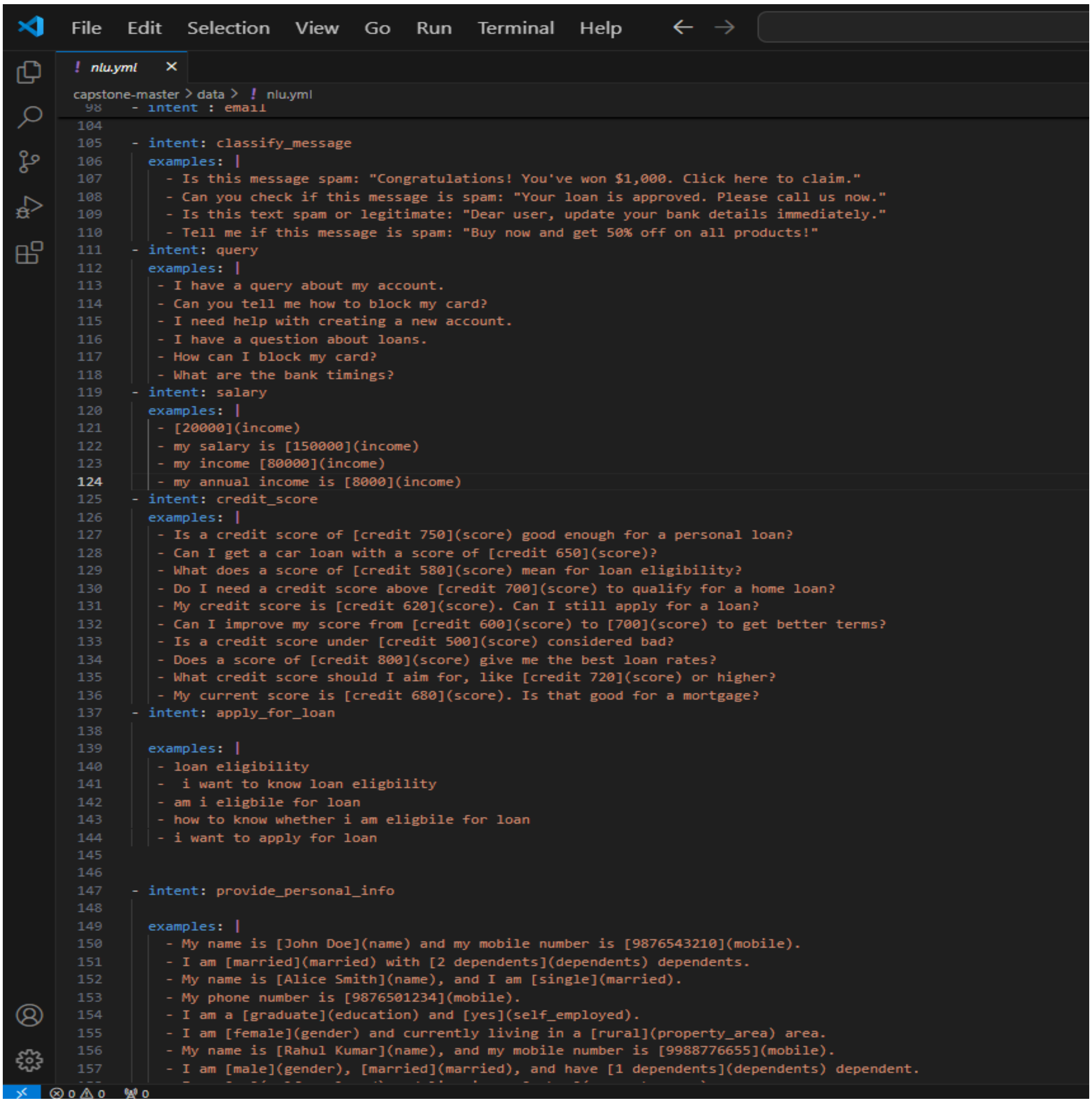


Figure 6.5: NLU.yml

2. domain.yml

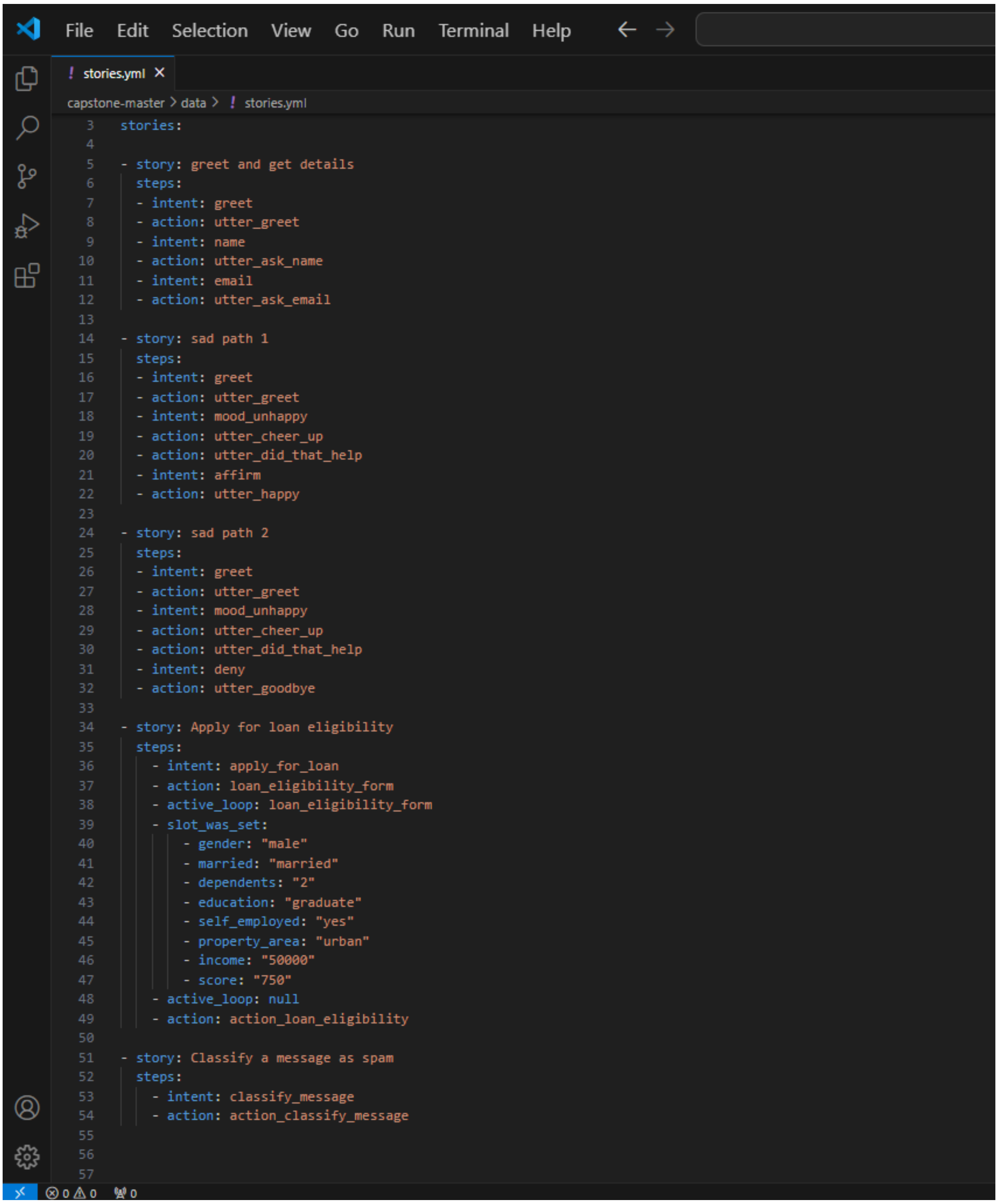
Once the assistant has a few potential users messages that it understands, it also needs to be able to give a response. It declares all of the intents, entities, responses and actions for this file. RASA distinguishes between two response types: utterances and actions. An utterance is a standard text response to a query, and actions are a classic way for custom logic for data manipulation or creating API calls. We provide several possible responses / txt responses, where applicable. If a response has more than one option, the one will be randomly chosen when that part of the response is triggered.

```yaml
utter_iamabot:
- text: "I am a bot, powered by Rasa."
utter_ask_name:

- text: "Hi there! What's your name?"
- text: "Could you please tell me your name?"

utter_ask_email:
- text: "Please provide your mail id."
- text: "Could you share your mail id  with me?"

utter_ask_gender:
- text: "What is your gender? (e.g., male, female, other)"

- text: "Can you tell me your gender?"
- text: "Please let me know your gender."

utter_ask_married:
- text: "What is your marital status? (e.g., married, single)"

- text: "Are you married? Please share your marital status."
- text: "Could you tell me if you're married?"

utter_ask_dependents:

- text: "How many dependents do you have?"
- text: "Can you share how many dependents you have?"

utter_ask_education:
- text: "What is your highest level of education? (e.g., graduate, ungraduate)"
- text: "What is your highest level of education?"
- text: "Can you tell me about your education?"

utter_ask_self_employed:
- text: "Are you self-employed? (yes or no)"

- text: "Are you self-employed? Please confirm."
- text: "Could you let me know if you're self-employed?"

utter_ask_property_area:

- text: "What type of area do you live in? (Urban, Semiurban, Rural)"
- text: "Can you tell me the type of area you live in?"

utter_ask_income:
- text: "What is your monthly income?"
- text: "Please provide your monthly income."
- text: "Could you share your income details?"

utter_ask_score:

- text: "What is your credit score?"
- text: "Can you share your credit score with me?"
```

Figure 6.6: Domain.yml

3. stories.yml

Stories are a set of example dialogues which instructs the assistant to reply in the most suitable way from previous inputs of the user. The format of this message is user's intent followed by an action or an assistant's reply. In the code block below, we see a story where the user and the assistant greet one another and the assistant helps the user achieve his or her goals simply.



```yaml
stories:

- story: greet and get details
  steps:
  - intent: greet
  - action: utter_greet
  - intent: name
  - action: utter_ask_name
  - intent: email
  - action: utter_ask_email

- story: sad path 1
  steps:
  - intent: greet
  - action: utter_greet
  - intent: mood_unhappy
  - action: utter_cheer_up
  - action: utter_did_that_help
  - intent: affirm
  - action: utter_happy

- story: sad path 2
  steps:
  - intent: greet
  - action: utter_greet
  - intent: mood_unhappy
  - action: utter_cheer_up
  - action: utter_did_that_help
  - intent: deny
  - action: utter_goodbye

- story: Apply for loan eligibility
  steps:
    - intent: apply_for_loan
    - action: loan_eligibility_form
    - active_loop: loan_eligibility_form
    - slot_was_set:
        - gender: "male"
        - married: "married"
        - dependents: "2"
        - education: "graduate"
        - self_employed: "yes"
        - property_area: "urban"
        - income: "50000"
        - score: "750"
    - active_loop: null
    - action: action_loan_eligibility

- story: Classify a message as spam
  steps:
    - intent: classify_message
    - action: action_classify_message
```

Figure 6.7: Stories.yml

4. config.yml

This file includes all components used in the NLU pipeline for parsing and making sense of messages as Natural Language, along with mapping intents and entities. It also outlines the policies for the CORE that guide decision-making after the NLU model interprets the message through its pipelines.

Figure 6.8: config.yml

5. rules.yml

The rules.yml file outlines organized conversational pathways for the chatbot by detailing rules activated by user intentions. When it comes to loan applications, the chatbot triggers a loan_eligibility_form to gather user information, establishes particular slots (such as gender, income, and score), and performs the action_loan_eligibility to assess eligibility. For message classification, the chatbot reacts to the classify_message intent by executing the action_classify_message. Finally, for inquiries related to loans, the chatbot processes the query intent by calling the action_chatgpt_faq to deliver appropriate responses. These regulations guarantee uniform and coherent replies for the specified intents.

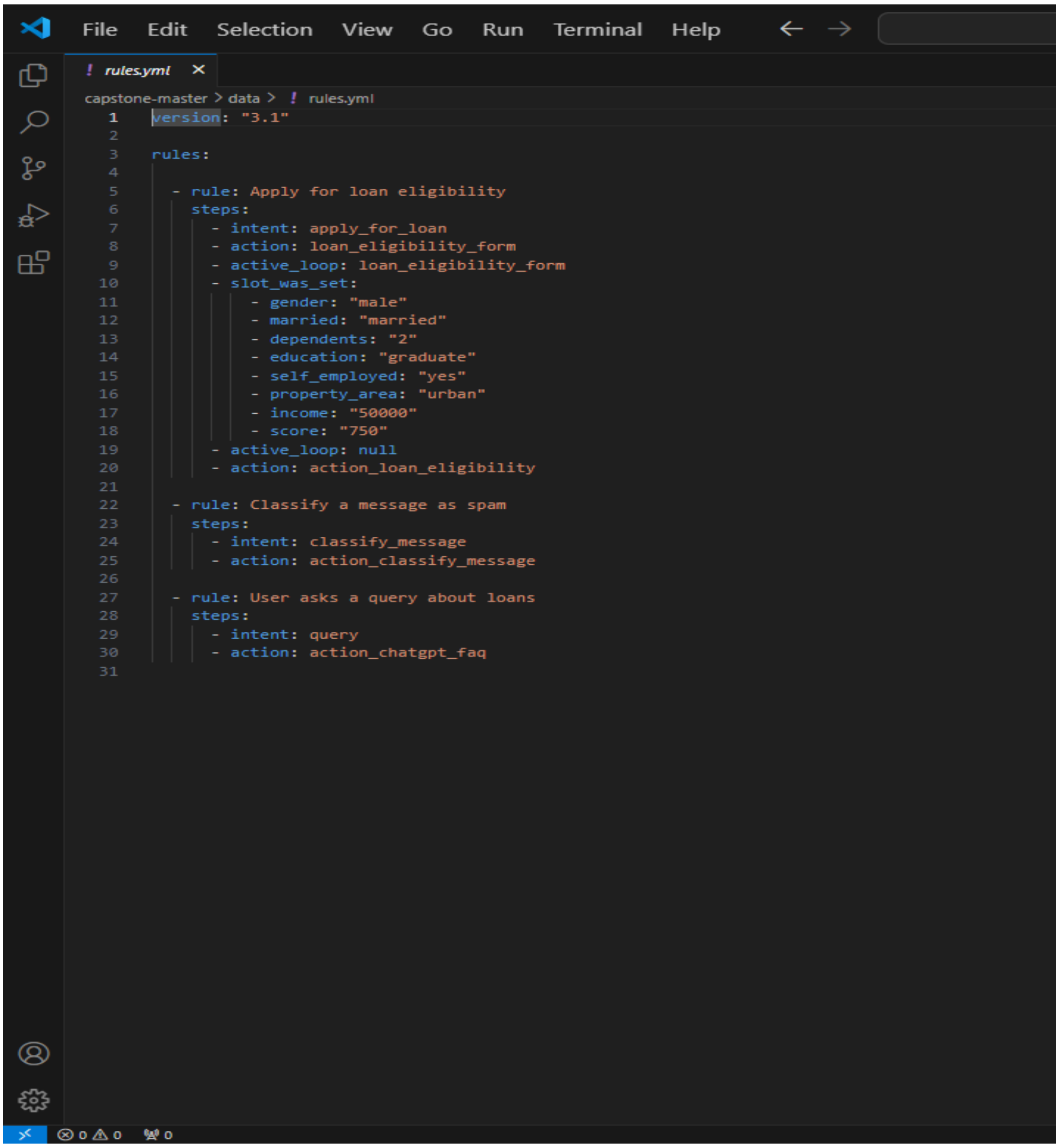


Figure 6.9:rules.yml

6. credentials.yml

The credentials.yml file sets up the chatbot's connection with different messaging and voice platforms. It contains placeholders for the credentials needed by platforms such as Facebook, Slack, SocketIO, and Mattermost. Each section for a platform details the required tokens, secrets, or URLs that are essential for authentication and communication. The rest channel is automatically supported, requiring no extra credentials. Furthermore, there is a specific entry for the rasa channel that directs to a local URL, facilitating interactions through Rasa Enterprise. This file guarantees that the chatbot can smoothly connect to various channels for effective communication.

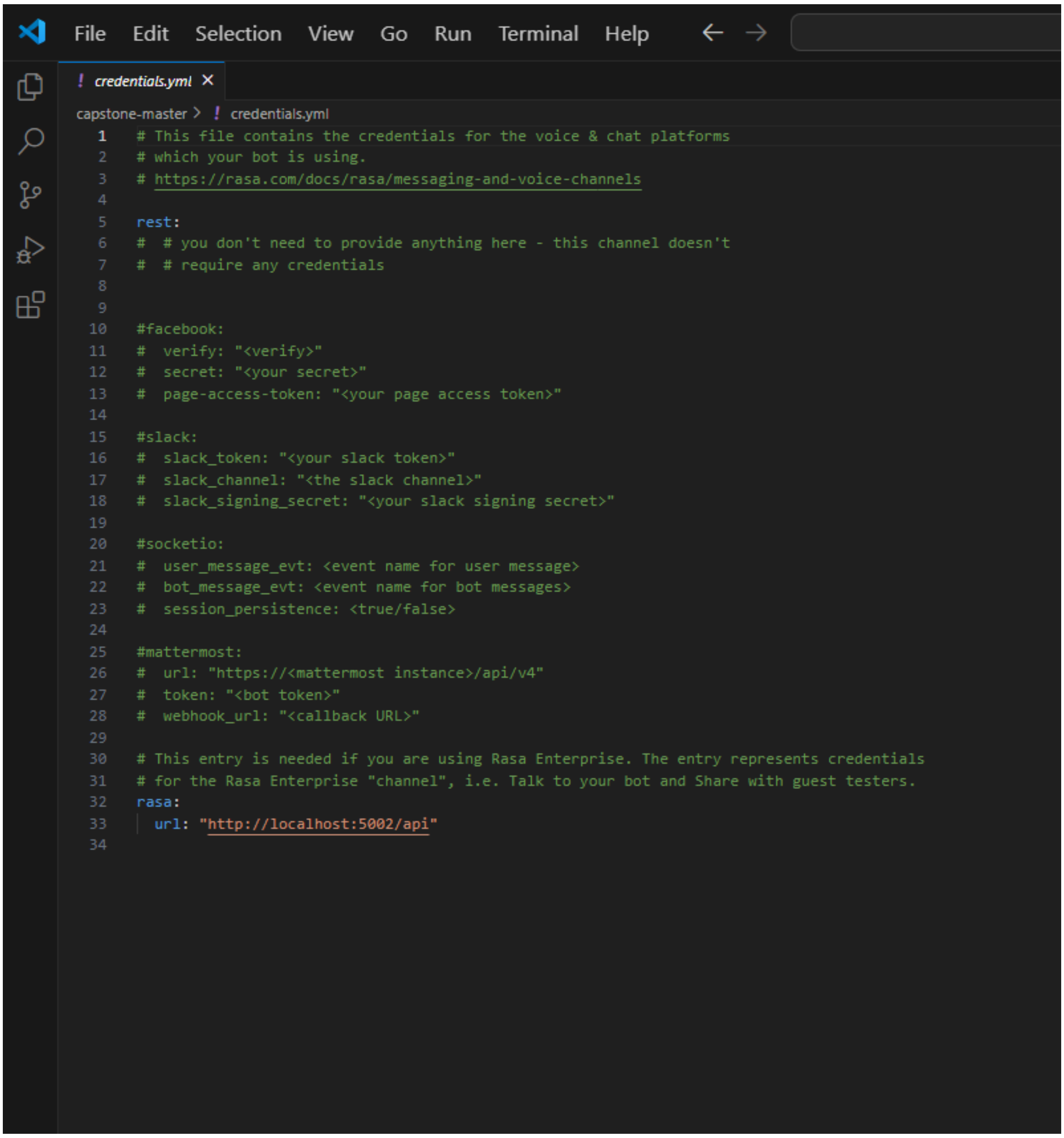


Figure 6.10: credentials.yml

7. endpoints.yml

The endpoints.yml file outlines the external services that the chatbot connects with. It details the action_endpoint URL, which links to a server that runs custom actions (e.g., http://localhost:5055/webhook). There are optional settings for model storage that permit retrieving models from a remote server. It also includes configurations for various tracker stores, like Redis or MongoDB, to save conversation data rather than relying on the standard in-memory storage. Moreover, it features placeholders for an event broker, which allows the streaming of conversation events to external platforms. This file facilitates smooth integration between the chatbot and its external services.

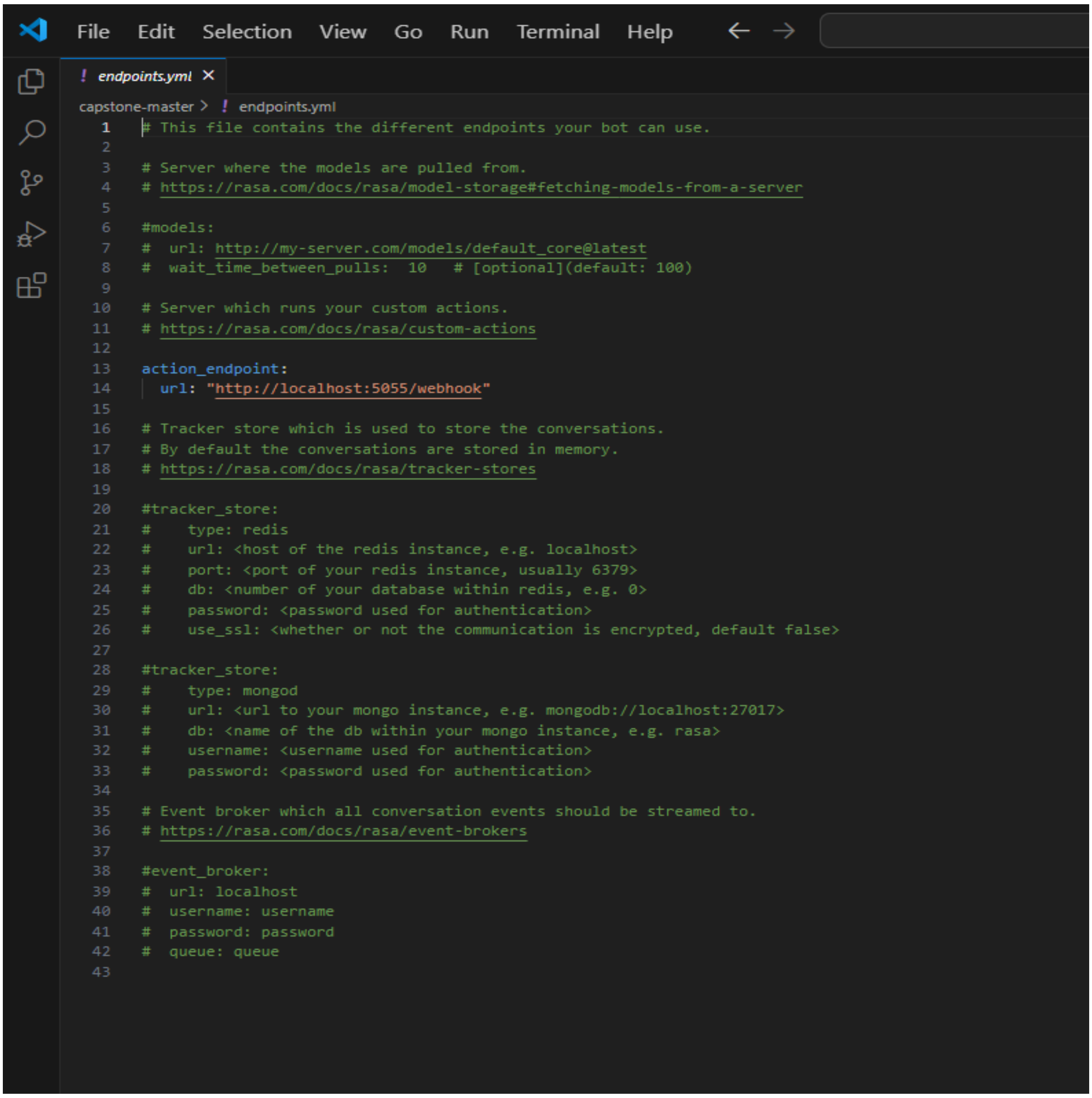


Figure 6.11: endpoints.yml

CHAPTER-7
TIMELINE FOR EXECUTION OF PROJECT
(GANTT CHART)

Project Timeline: The timeline for the Customer Support Chatbot with ML mission is structured by several evaluation levels, each focusing on specific factors of the venture's development. These reviews are crucial checkpoints that make certain the undertaking progresses in accordance to plan and meets the required standards. Moreover, the final Viva-Voce serves because the remaining evaluation of the venture's outcomes and your information of its ideas on Chatbot.

1. Task 1 (Completing the Title)

During this phase, the project name will be finalized in consultation with your supervisor. You will conduct a literature review to obtain relevant information and insights. The goals of the project will be finalized and the methodology to achieve these goals will be decided

Review-0: September 12, 2024 to September 15, 2024

During this phase, main focus on project initiation, planning, and initial research. Defining the scope of the project, objectives and outputs. Identification of stakeholders and their roles. They began creating a detailed plan for the project.

2. Task 2 (Abstract, Literature review, Objectives and proposed method)

In this review phase we focus on refining the project abstract by conducting a comprehensive literature survey with a minimum of 10 research articles, identifying project objectives, highlighting the shortcomings of existing methods, designing a new method, creating an architecture diagram, defining modules, specifying hardware and software details, creating a timeline using a Gantt chart, compiling references and submitting a Review- 1 report.

Review-1: October 15, 2024 to October 21, 2024

In this phase we focus for in-depth research and requirements gathering. Dive into existing chatbot technologies, and gather detailed requirements from stakeholders and potential users.

3. Task 3 (Algorithm Details, Source Code, Implementation Details and Message Submission)

In this phase, you will dive into the details of the algorithm, provide insight into the source code, and demonstrate 50% of the implementation details through a live project demo, and submit 50% of the report in softcopy format.

Review-2: November 19, 2024 to November 22, 2024

In this phase is used for design and architecture. Design the User interface and User experience (UI) of Customer Support Chatbot with ML. Defining the technical architecture and components needed for the Customer Support Chatbot with ML.

4. Task 4 (Algorithm details, source code, full implementation and report submission)

In this review phase includes algorithm details, source code specifics, demonstration of 100% project implementation, submission of a complete report in both paper and electronic form, and demonstration of a fully implemented live demo project.

Review-3: December 17, 2024 to December 20, 2024

At this stage can be a stage of development, Setting up the development environment and implementing the Customer Support Chatbot with ML using chat widget integrated with back end machine learning, knowledge base, actions and thoroughly testing the functionality of the Customer Support Chatbot with ML.

5. Assignment 5 Final project Viva-Voce

The Final Viva-Voce is the culmination of our project journey. At this stage a comprehensive assessment where you will present your project, discuss its various aspects, defend your choices and methodologies and demonstrate your understanding of the project's development and outcomes of Customer Support Chatbot with ML.

Final Viva-Voce: January 10, 2024 to January 17, 2024

This phase for project completion and deployment. Deployment on target platform (e.g. Web), final demonstration of Customer Support Chatbot with ML in a production environment and execution of final testing and user training. presentation of completed projects during Viva-Voce.

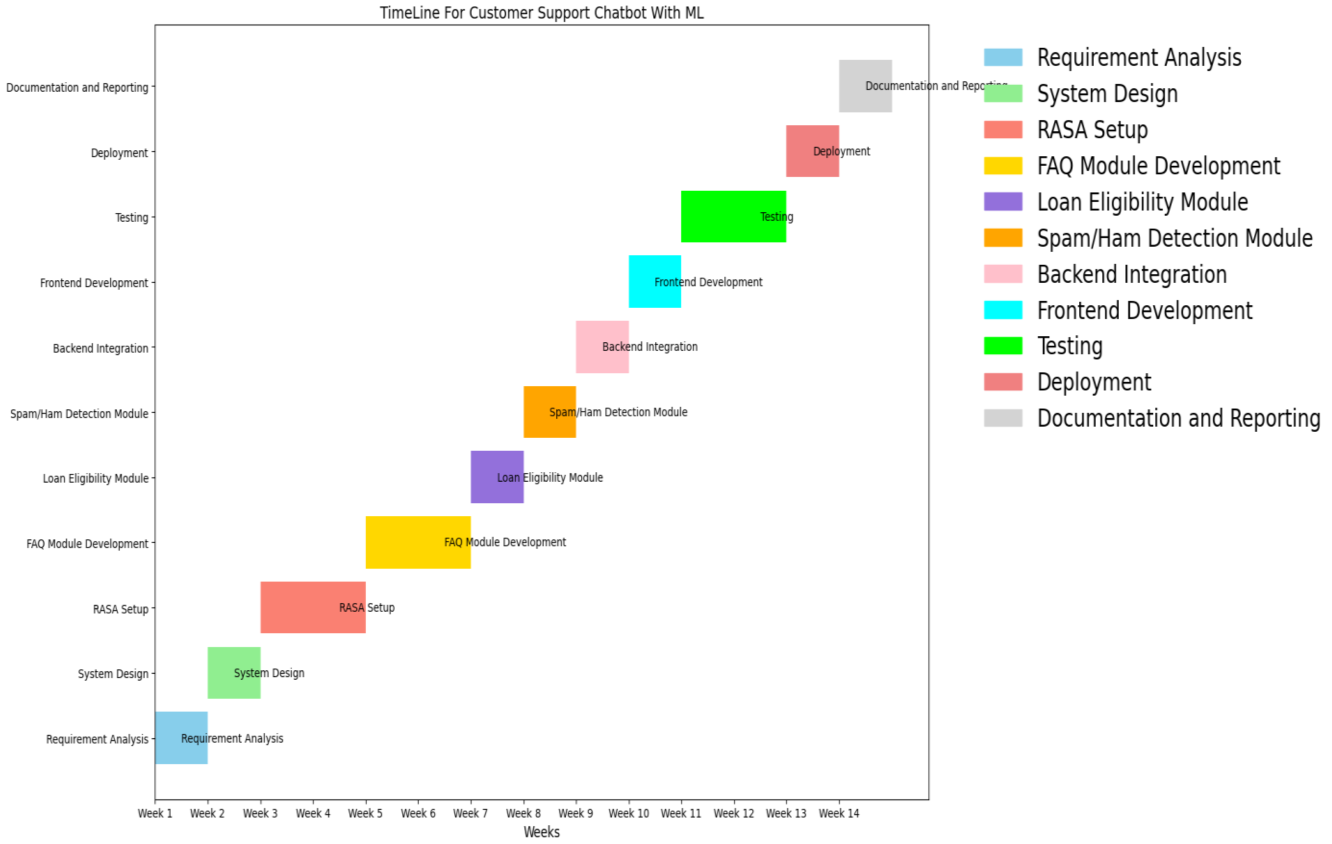


Figure 7.1: Timeline

CHAPTER-8

OUTCOMES

Intent Classification Accuracy:

The intent classification system aims for an accuracy range of 85-95%. This guarantees that the system can accurately recognize the user's intent with high precision, particularly in multi-turn conversations where context is vital. The notable precision and adequate recall rates demonstrate the system's ability to effectively manage various conversational dynamics.

Entity Extraction Accuracy:

The entity extraction functionality operates with an accuracy of 80-90%, resulting in few errors when pinpointing important entities. This level of accuracy is crucial for effectively gathering key details, which is essential for creating personalized responses and facilitating data-driven decisions within the system.

Response Accuracy:

The target response accuracy for the system is set at 85-95%, with the potential to surpass 95% for frequently asked questions (FAQs). This ensures that users receive responses that are both accurate and relevant to the context, boosting the overall reliability and credibility of the chatbot.

Spam/Ham Classification Accuracy:

The classification of spam and ham is an important capability, achieving an accuracy rate of 90-96%. The precision remains at 95%, while the recall is at 92%, allowing for the effective distinction and separation of spam from legitimate user messages while avoiding major false positives or negatives.

Loan Eligibility Prediction Accuracy:

In terms of predictive analysis, such as for loan eligibility, the system reaches an accuracy of 85-90%. Precision and recall may fluctuate, reflecting the variable nature of input data. This accuracy range ensures solid decision support when determining eligibility.

User Satisfaction Rate:

User satisfaction stands as a critical measure, with a rate of 80-90%. This reflects the system's capability to fulfill user expectations in responsiveness, accuracy, and general experience.

Confusion Matrix Insights:

Data derived from the confusion matrix offers deeper insights into misclassifications, allowing for focused improvements in system accuracy and dependability. While specific insights are not provided, they serve as a basis for enhancing overall performance.

Dialog Success Rate:

The system records a dialog success rate of 85-90%, signifying effective fulfillment of user requests and successful resolution of queries in a variety of situations.

Fallback Accuracy:

Fallback systems are designed to operate with an accuracy of 80-85%, ensuring that when the system is unable to fulfill a request, it addresses the situation gracefully by offering alternative options or clarifications.

Latency:

The system's response times are optimized to remain below 1 second for simple intents and under 2 seconds for API-driven operations. This low latency ensures a smooth and efficient user experience, reducing wait times and enhancing the flow of interaction.

CHAPTER-9

RESULTS AND DISCUSSIONS

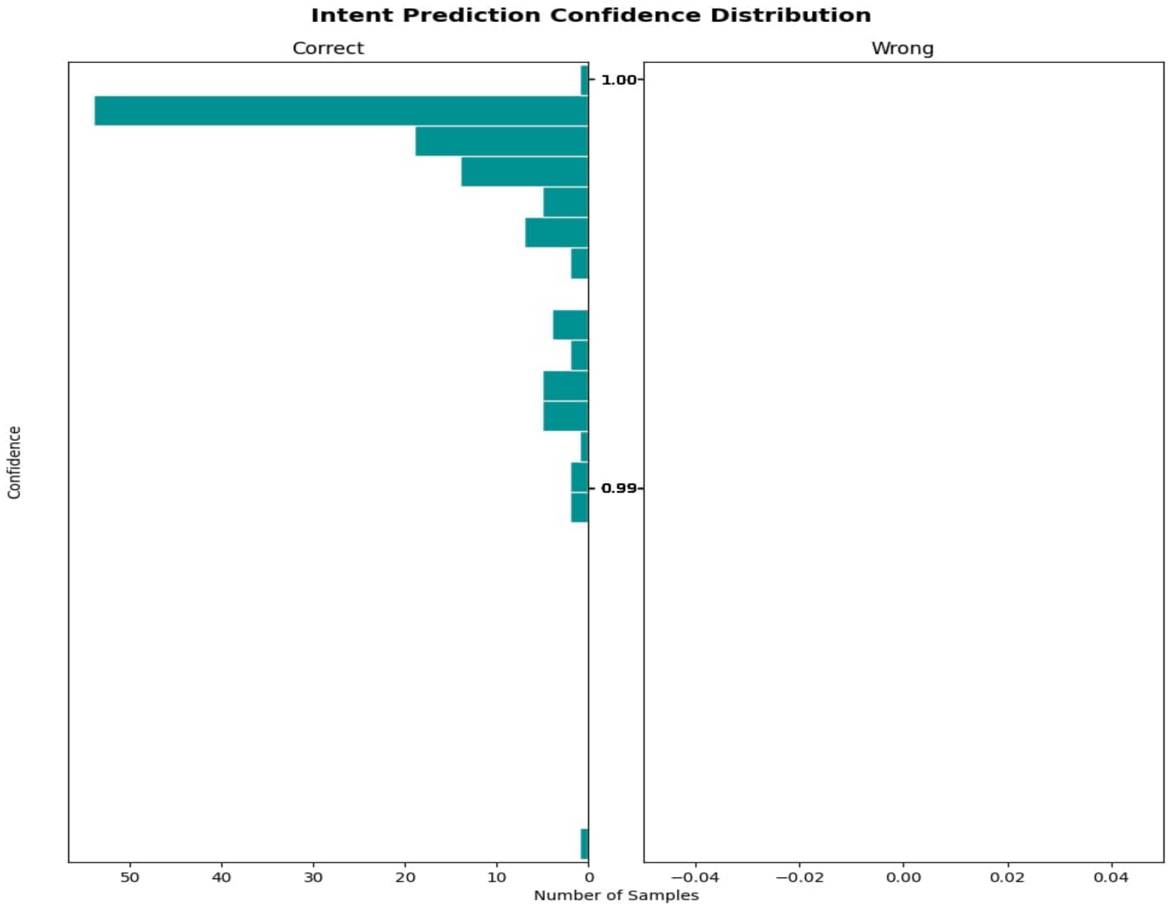


Figure 9.1: Intent Prediction Confidence Distribution

1. Overall Performance

Accuracy: 1.0

The chatbot achieved an accuracy rate of 100%, indicating that all intents were accurately classified. This reflects a strong correlation between the training dataset and the test dataset.

2. Per-Intent Performance

Each intent scored a precision, recall, and F1-score of 1.0, demonstrating flawless classification performance.

Metrics Explanation:

Precision: The ratio of correctly identified samples to the total classified as that intent.

Recall: The ratio of correctly identified samples to the actual samples of that intent.

F1-Score: The harmonic mean of precision and recall, offering a comprehensive measure.

Key Takeaways:

Every intent, such as apply_for_loan, credit_score, greet, and provide_personal_info, received perfect scores.

There was no confusion among intents, as indicated by ("confused_with": {} for all intents).

3. Macro, Weighted, and Micro Averages

Macro Avg: The arithmetic mean of precision, recall, and F1-score across all intents.

The perfect macro averages (1.0) indicate consistent performance across all intents, regardless of their frequency.

Weighted Avg: Averages that take into account the number of samples for each intent.

As the weighted average is also 1.0, this shows strong performance for both frequently occurring and less common intents.

Micro Avg: Summarizes overall performance by treating all intents collectively.

The micro average confirms that there were no classification errors.

4. Support

The total support is 124, which corresponds to the sum of support values for all intents.

Intents such as provide_personal_info (19), mood_unhappy (14), and greet (13) have higher support, showing more test cases for these intents.

Even intents with lower support (e.g., email, salary, bot_challenge, with 4 each) achieved perfect scores, demonstrating the model's strength despite fewer samples.

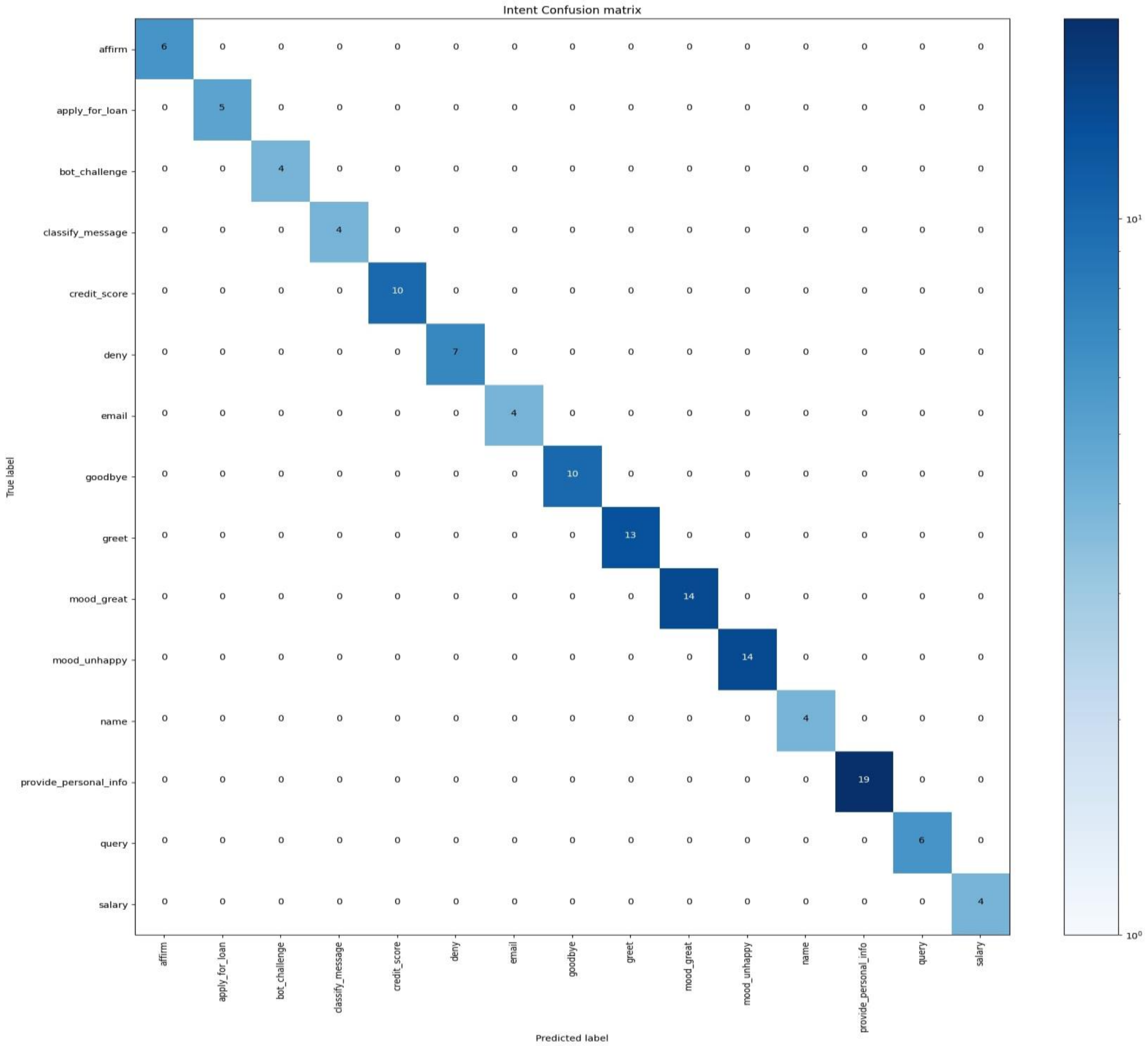


Figure 9.2: Intent Confusion matrix

Discussion on the Intent Confusion Matrix

This confusion matrix assesses how well your chatbot performs in categorizing intents. The rows indicate the actual labels, while the columns represent the labels predicted by the model. Below is a thorough examination of the given confusion matrix:

Key Observations

Diagonal Dominance:

The matrix exhibits a strong diagonal trend, signifying that the majority of intents are identified accurately. For instance:

affirm: Correctly identified 6 times.

greet: Correctly identified 13 times.

provide_personal_info: Correctly identified 19 times.

Zero Misclassifications:

All off-diagonal values are zero, which means there were no misclassifications during the testing phase. This indicates flawless intent classification performance. The brightness of the diagonal cells corresponds to the number of samples for each intent. Darker cells signify intents with a higher number of samples, such as provide_personal_info and mood_unhappy.

Performance Insights

High Accuracy:

The confusion matrix reveals perfect accuracy across all intents, which is an outstanding outcome.

Intent Generalization:

The model has successfully learned to differentiate between all intents, including those with fewer training examples, such as apply_for_loan, email, and salary.

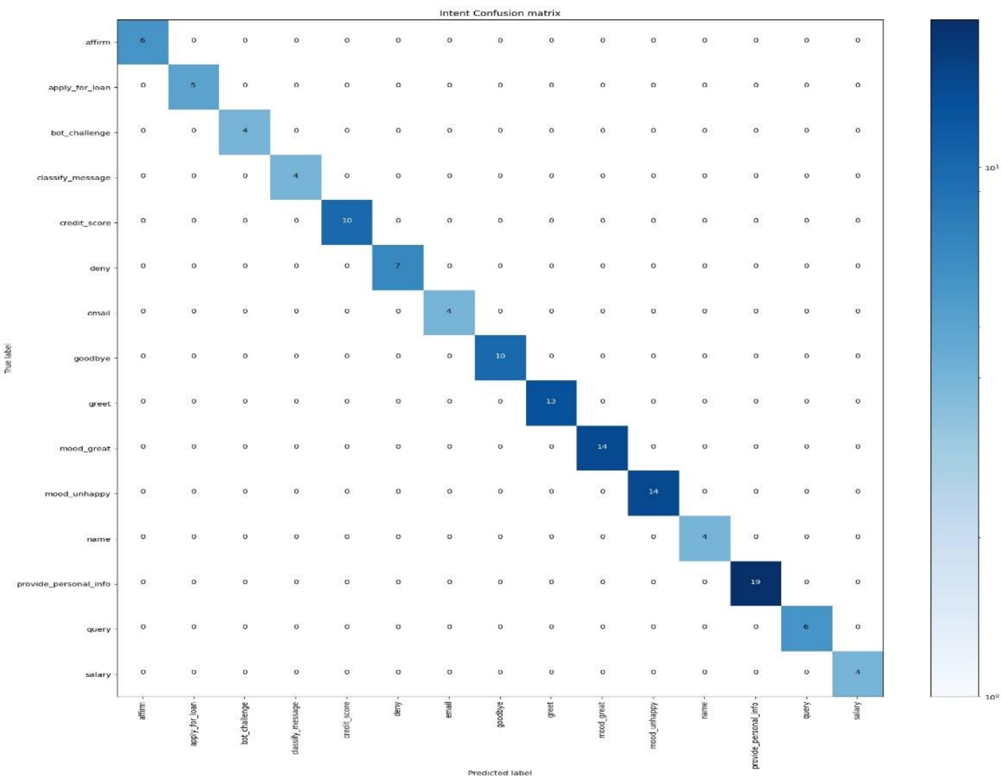


Figure 9.3:Entity Confusion matrix

Discussion on the Confusion Matrix Image

The confusion matrix presented illustrates how the DIETClassifier performs in entity recognition tasks. Here are some important insights and a thorough examination of the image:

Key Elements of the Confusion Matrix

Diagonal Dominance:

The matrix displays a prominent diagonal pattern, indicating that the classifier accurately identifies most entities within their respective categories. For instance:

dependents: Accurately classified on 10 occasions.

name: Accurately classified on 14 occasions.

no_entity: Accurately classified on 552 occasions.

Misclassifications:

The off-diagonal elements highlight instances where the classifier misidentified entities. Notable observations include:

The model confused no_entity with property_area in 2 instances and with score in 1 instance.

No significant misclassifications were found among other entities, reflecting an overall robust performance.

Darker shades along the diagonal indicate a higher number of correct identifications, while lighter shades off the diagonal signify fewer misclassifications.

Interpretation and Implications

High Accuracy for Dominant Entities:

The model excels notably with entities such as no_entity, name, and score, likely due to a larger volume of training examples available for these categories.

CHAPTER-10
CONCLUSION

In this project, we successfully created an AI-driven Customer Support Chatbot tailored for the banking sector using the RASA framework. This chatbot incorporates vital features such as responding to frequently asked questions (FAQs), assessing loan eligibility based on user input, and identifying spam or irrelevant messages to maintain effective communication. It aims to improve customer service by providing 24/7 assistance, automating common inquiries, and minimizing the need for human intervention, which enhances operational efficiency.

By utilizing Natural Language Understanding (NLU) for intent classification and entity recognition, the system accurately processes user questions and delivers appropriate responses. The included spam detection function filters out unwanted messages, making the user experience more organized and streamlined. The chatbot's ability to evaluate loan eligibility further enriches the service by allowing users to make quick, self-service financial decisions.

Integrating RASA with web platforms guarantees that the chatbot is both accessible and scalable, allowing it to function across various devices and channels. Through this initiative, we have highlighted the role of conversational AI in revolutionizing customer support in banking, lowering operational costs, boosting user satisfaction, and enhancing overall service delivery.

In summary, this project establishes a strong groundwork for future advancements in AI-enabled customer support systems. Potential enhancements could involve the adoption of more sophisticated machine learning approaches, broadening the knowledge base, and connecting with additional banking services, thereby making the chatbot even more adaptable and advantageous for both customers and banking personnel.

REFERENCES

[1] T. Bocklisch, J. Faulkner, N. Pawlowski, and A. Nichol, “Rasa: Open-Source Language Understanding and Dialogue Management,” NIPS Workshop on Conversational AI, Dec. 2017

[2] S. Johnson, S. Sri Jayanthi, T. V. N. L. Harika Jhansi, K. Pavithra, S. Sowmya and M. u. Mahathi Devi, "Chat Finance: Tracking Wealth with a Conversational Bot," 2024 International Conference on Expert Clouds and Applications (ICOECA), Bengaluru, India, 2024, pp. 253-257, Doi: 10.1109/ICOECA62351.2024.00054.

[3] Frommert, Christian & Häfner, Anna & Friedrich, Julia & Zinke-Wehlmann, Christian. (2018). Using Chatbots to Assist Communication in Collaborative Networks: 19th IFIP WG 5.5 Working Conference on Virtual Enterprises, PRO-VE 2018, Cardiff, UK, September 17-19, 2018, Proceedings. 10.1007/978-3-319-99127-6_22.

[4] Jiao, Anran. (2020). An Intelligent Chatbot System Based on Entity Extraction Using RASA NLU and Neural Network. Journal of Physics: Conference Series. 1487. 012014. 10.1088/1742-6596/1487/1/012014.

[5] S. F. Suhel, V. K. Shukla, S. Vyas and V. P. Mishra, "Conversation to Automation in Banking Through Chatbot Using Artificial Machine Intelligence Language," 2020 8th International Conference on Reliability, Infocom Technologies and Optimization (Trends and Future Directions) (ICRITO), Noida, India, 2020, pp.611-618, Doi: 10.1109/ICRITO48877.2020.9197825.

[6] Fathima, Sasha & Student, Suhel & Shukla, Vinod & Vyas, Dr Sonali & Mishra, Ved P, “Conversation to Automation in Banking Through Chatbot Using Artificial Machine Intelligence Language”. 2020 International Conference on Recent Innovations in Technology and Engineering (ICRITO), Delhi , India, 2020, Doi: 10.1109/ICRITO48877.2020.9197825.

[7] The Rasa documentation. [Online]. Available: https://rasa.com/docs/rasa/2.x

[8] Singh, Netra & Singh, Devender. "Chatbots and Virtual Assistant in Indian Banks". Industrija 2019 Institute of Economic Sciences (IES), Belgrade, Serbia, Journal of Physics: 47. 75-101. 10.5937/industrija47-24578.

[9] Young T, Hazarika D, Poria S and Cambria E, "Recent Trends in Deep Learning-Based Natural Language Processing". 2017 *arXiv:1708.02709,* Computation and Language.

[10] A. Singh, K. Ramasubramanian, and S. Shivam, "Introduction to Microsoft Bot, Rasa, and Google Dialog Flow", In Building an Enterprise Chatbot (pp. 1-22), (2019) Springer Nature, Cham, Switzerland.

[11] R. S. Bird, J. Klein, and E. Loper, *Natural Language Processing with Python*, O'Reilly Media, 2009.

[12] J. Xu, Y. Wang, and L. Wang, "Enhancing RNN-based dialogue systems with pretrained contextual representations," *Proc 2021 Conf. Empirical Methods Natural Language Process*.

[13] S. D. Goh, T. Y. Soong, and S. K. Ho, "Spam detection with advanced feature selection and the influence of deep learning," *Proc. IEEE Conf. Cyber Intell. Security*, 2018.

[14] L. Wu, L. Dong, and W. Wei, "Understanding the impact of transformers in NLP: A comprehensive review," *Proc. 2020 Int. Joint Conf. Artif. Intell.*, 2020.

[15] S. Chaudhuri, A. S. Nair, and S. D. D. Thakur, "A comparative analysis of machine learning approaches for text classification," *Proc. 2021 Int. Conf. Machine Learning*, 2021.

[16] W. Y. S. Lee, S. P. Shandilya, and B. N. S. Bhadoria, "Spam email detection using machine learning algorithms," *J. Artif. Intell. Res.*, vol. 25, pp. 375–392, 2020.

[17] P. O. Lavrenko, V. S. Tsarev, and I. S. Goryunov, "Improvement of spam detection models in text mining systems," *Proc. Int. Conf. Artif. Intell. Comput. Sci.*, 2021.

[18] K. L. Dehghani, "Natural language processing for conversational AI: Recent advances," *IEEE Trans. Cogn. Dev. Syst.*, vol. 12, no. 3, pp. 343–359, 2020.

[19] R. S. Bird, J. Klein, and E. Loper, *Natural Language Processing with Python*, O'Reilly Media, 2009.

[20] J. Xu, Y. Wang, and L. Wang, "Enhancing RNN-based dialogue systems with pretrained contextual representations," *Proc. 2021 Conf. Empirical Methods Natural Language Process.*, 2021.

APPENDIX-A

PSUEDOCODE

Pseudo code for Customer Support Chatbot using Machine Learning

```
START

#Data Preparation
DEFINE datasets for spam detection, loan eligibility, and FAQs
LOAD labeled datasets
PREPROCESS data:
   REMOVE inconsistencies and null values
   ENCODE categorical variables
   STANDARDIZE text data

#Model Development
TRAIN DistilBERT for spam detection
TRAIN Random Forest for loan eligibility prediction
TRAIN DIETClassifier for FAQ intent matching

#RASA Integration
INITIALIZE RASA framework
DEFINE intents: loan_eligibility, faq_query, spam_check
DEFINE entities: income, credit_score
SETUP NLU pipeline with CountVectorsFeaturizer and DIETClassifier
IMPLEMENT custom actions for:
   Spam detection model
   Loan eligibility model
   FAQ response generation
CONFIGURE RulePolicy for dialogue management

#Deployment
SETUP deployment environment
HOST chatbot server
CONNECT user interface to chatbot

#Interaction Workflow
WHILE chatbot is running:
   RECEIVE user query
   PARSE user query for intent and entities
   IF intent is "loan_eligibility":
      CALL loan eligibility model
      RETURN loan eligibility result
   ELSE IF intent is "spam_check":
      CALL spam detection model
      RETURN spam detection result
   ELSE IF intent is "faq_query":
      CALL FAQ intent model
```

```
      RETURN matching FAQ response
   ELSE:
      RETURN default response
```

#Testing and Evaluation

```
FOR each function (spam detection, loan eligibility, FAQ matching):
   CALCULATE precision, recall, and F1-score
   LOG performance metrics
PERFORM manual testing for multi-turn conversations
```

#Continuous Monitoring

```
WHILE chatbot is live:
   TRACK performance metrics
   UPDATE models with new data
   IMPROVE conversation flows as needed
```

STOP

APPENDIX-B

SCREENSHOTS

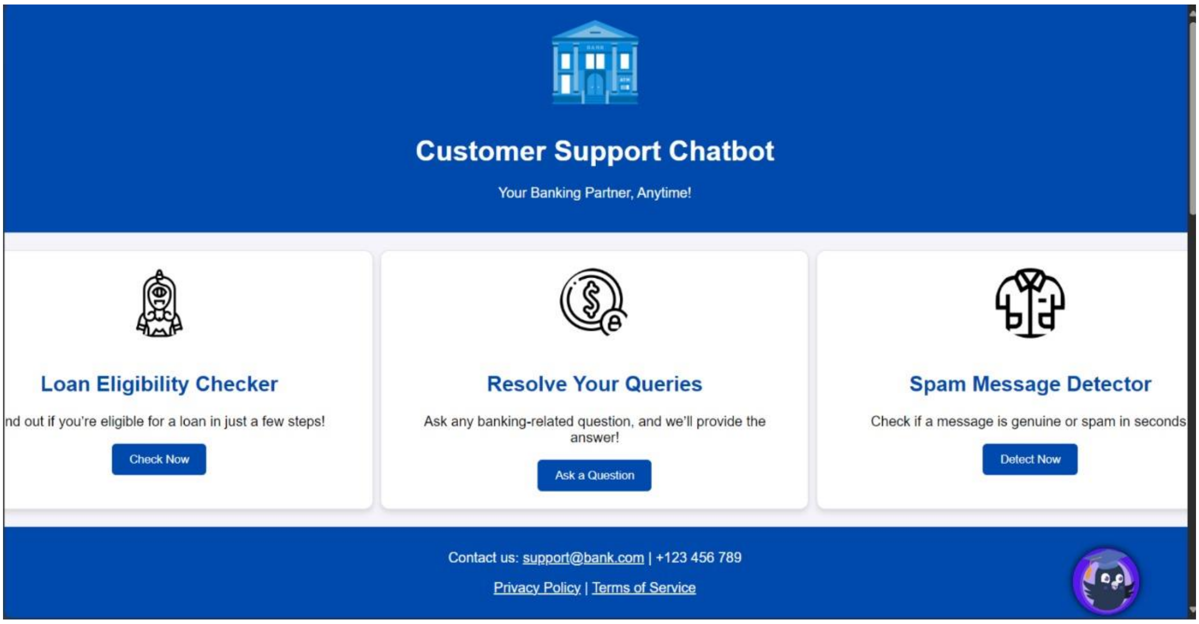


Figure 11.1: User interface

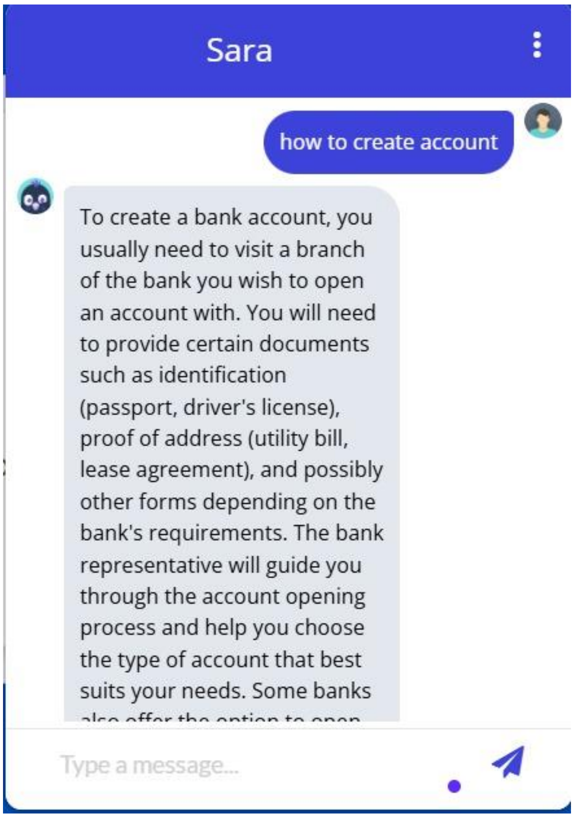


Figure 11.2: Query

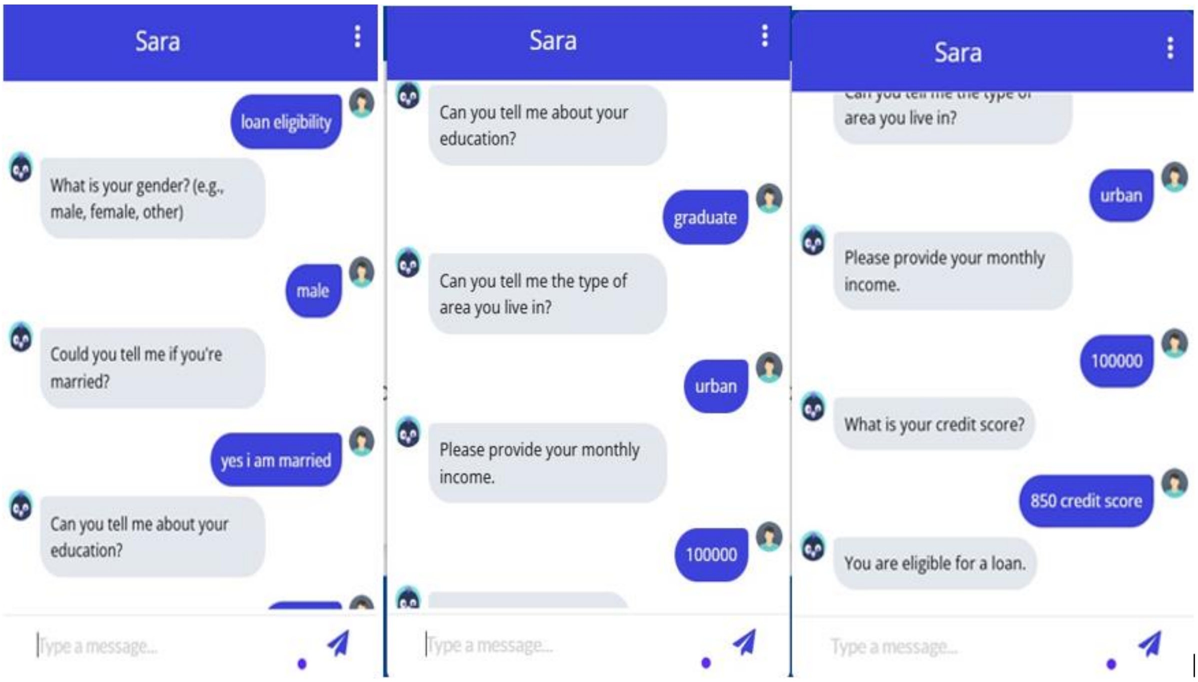


Figure 11.3: Loan eligibility

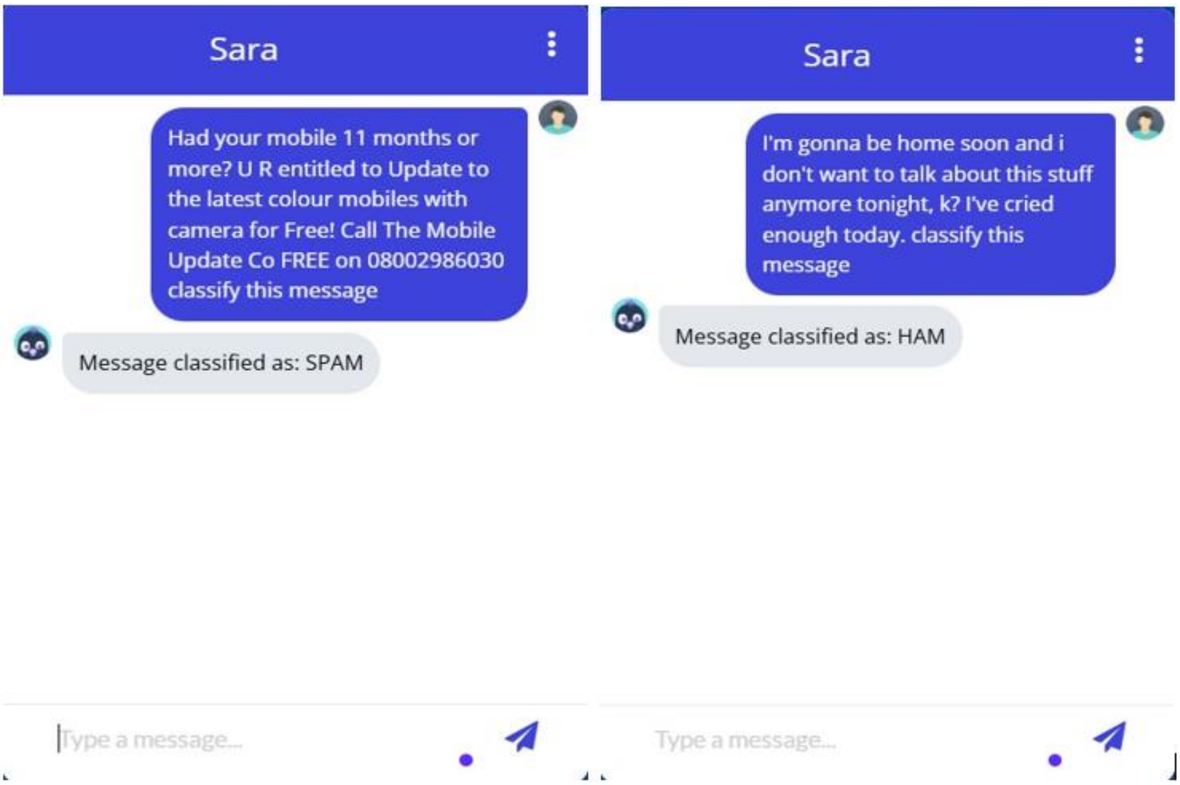


Figure 11.4: Spam classifier

APPENDIX-C

ENCLOSURES

Acceptance Certificate

JETIR553842 - Your Paper Accepted | Acceptance and Review Notification for your Paper | 7.95 impact factor

1 message

Editor JETIR <editor@jetir.org> Thu, Jan 16, 2025 at 10:50 AM

To: anjankumarposa@gmail.com

International Journal of Emerging Technologies and Innovative Research(JETIR)
An International Scholarly Open Access Journal, Peer-Reviewed, Refereed Journal Impact Factor 7.95 Calculate by Google Scholar and Semantic Scholar | AI-Powered Research Tool, Multidisciplinary, Monthly, Multilanguage Journal Indexing in All Major Database & Metadata, Citation Generator, Peer-Reviewed, Refereed, Indexed, automatic Citation Open Access Journal

Acceptance Notification and Review Result of Your Paper - JETIR553842 | JETIR (ISSN:2349-5162) | www.jetir.org | editor@jetir.org Your Email id: anjankumarposa@gmail.com Track Your Paper Link Track Your Paper https://www.jetir.org/trackauthorhome.php?a_rid=553842

Dear Posa Anjan Kumar,

Your manuscript with Registration ID: JETIR553842 has been **Accepted** for publication in the Journal of Emerging Technologies and Innovative Research (www.jetir.org). Track Your Paper Link Track Your Paper https://www.jetir.org/trackauthorhome.php?a_rid=553842 Your Review Report is as follows:

An International Scholarly Open Access Journal, Peer-Reviewed, Refereed Journal Impact Factor 7.95 UGC Approved Journal No 63975(19)
Review Results:

Registration ID	JETIR553842
Email ID	anjankumarposa@gmail.com
Paper Title	Customer Support Chatbot with Machine Learning for Banking System: Spam Detection and Customer Assistance
Review Status	Accepted
Impact Factor & Licence:	Open Access, Peer-Reviewed, Refereed, Indexing,ISSN Approved,DOI and Creative Common Approved & 7.97 Calculated by Google Scholar
Unique Contents	88 %

© 2025 JETIR January 2025, Volume 12, Issue 1 www.jetir.org (ISSN-2349-5162)

JETIR.ORG ISSN: 2349-5162 | ESTD Year : 2014 | Monthly Issue
JOURNAL OF EMERGING TECHNOLOGIES AND INNOVATIVE RESEARCH (JETIR)
An International Scholarly Open Access, Peer-reviewed, Refereed Journal

Customer Support Chatbot with Machine Learning for Banking System: Spam Detection and Customer Assistance

Dr. Manjula H M
Associate Professor, Department of Computer Science,
Presidency University, Bengaluru -560 064, Karnataka, India.

Manda Kuswanth[1], Swathi M[2], Anusha A[3], Posa Anjan Kumar[4]
Presidency University, Bengaluru, Karnataka, India.

Abstract - Conversational AI technologies are transforming customer interactions across all industries by delivering automated, real time, instant support. The focus of this project is to create a chatbot for the banking industry to serve as a users' center to the questions; to evaluate a potential borrower's ability to secure a loan; and to filter out any spam. Specific machine learning tasks are exploited through the use of machine learning models targeted on particular tasks using RASA framework for conversation management and natural language understanding (NLU). The chatbot features include, precise intent detection, multi turn conversation handling, and a modular design that makes it easy for you to expand it in future. Meaningful results of evaluation show that these chatbots are ultimately performing very well with intent recognition, spam filtering, and loan eligibility assessment, suggesting that these chatbots will effectively contribute to improve user experience and operate with greater efficiency. This work sets the stage for future work from exploring the opportunities for voice activated conversational AI in other industries like banking.

***Key words*: Chatbot, Conversational AI, Banking Sector, NLU, Machine Learning, RASA Framework, User Experience, Loan Eligibility, Intent Recognition, Spam Filtering.**

I.INTRODUCTION

Conversational AI has made huge strides ahead and its impact on customer service has been especially radical in heavily regulated sectors like banking where positive feedback, security, and growth depend on it. The primary interface to customers is chatbots, or conversational agents [18], which 24/7 support customers while also lowering the operational costs. Natural Language Processing combined with machine learning are used in these banking chatbots to respond to a variety of customer queries from basic account queries to much more complex financial queries. These systems achieve personalized and contextual experiences by combining effective intent detection, entity identification as well as conversation management, resulting in higher customer satisfaction and loyalty.[1]

In this paper, I talk about the development and deployment of a banking chatbot built on top of the Rasa framework [2], which is an open-source NLP framework [16] that is highly modular and flexible. Advanced ML processes support the intent classification, entity recognition, and conversation management. For both intent recognition and entity extraction, the DIETClassifier[1] is used in the designed chatbot and Rasa CORE [2] is used to enable dynamic, context sensitive multi turn conversations.

This chatbot also takes care of common queries and supports account management, and uses machine learning classifiers for spam detection [16] to remain able to filter out the irrelevant or malicious messages [26]. Additionally, the system provides loan eligibility [30] assessments based on information a user elects to submit, including income, credit ratings, etc. Unlike most rule based systems that tend to get pretty rigid as

© 2025 JETIR January 2025, Volume 12, Issue 1 www.jetir.org (ISSN-2349-5162)

the tasks get more complex, this chatbot leverages ML to make it flexible and scalable for a number of different banking tasks.

II. REVIEW OF LITERATURE

Thomas et al. [1] examine how Rasa can be used to help in building smart chatbots to support service in various industry. Rasa Natural Language Understanding (NLU) pipeline excels at interpreting complex user intents and has machine learning models for tailored interactions, something that they highlight. The flexibility of Rasa behind custom actions and entities makes it a powerful tool for supporting automating customer support, as their research shows.

Zhang et al. [3] investigate the use of intent recognition models in Rasa to enhance the customer service at financial institutions. Their work builds upon using pre-trained models such as BERT and DistilBERT to classify user intents in order to increase the accuracy of responses to questions about services offered by a bank. The authors also suggest that, rather than learning via random initialization, fine tuning transformer models for considered tasks is beneficial towards providing relevant answers with as few mistakes as possible.

Gupta et al. [5] authors work on integrating a loan eligibility classification system into a chatbot. Using decision trees and machine learning algorithms, they show how a Rasa based chatbot can gather user data and predict a loan eligibility. The use of custom actions in Rasa to realize rule based logic for financial applications is exactly what their research supports: Real time decision making.

Patel et al. [6] analyze how better multi turn conversations can be established by Rasa's dialogue management policy. Next, they describe the use of RulePolicy and MemoizationPolicy to persist context in the case of long conversations to maintain a smooth conversational experience. How the chatbot is made to track its users history and adapt responses on the basis of past conversations is explained by the authors; this allows their chatbot to handle complex banking or customer service queries.

Williams et al. [7] study how machine learning helps in identifying the intent and extracting the entities of customers for use in chatbots. They use Rasa's DIETClassifier to show how it performs very well at matching customer intent, and extracting key entities like dates, amounts and product names. Another interesting feature of the study is Rasa's NLU components' ability to learn and improve with industry specific datasets for more specialized customer queries.

Lee et al. [8] compare Rasa's NLU pipeline against other chatbot frameworks for sentiment analysis. A research showcases how pre trained models such as DistilBERT can be integrated into Rasa's pipeline to improve the chatbot's response on customer queries with a more empathetic approach by detecting the sentiment. They talk about their exploration towards fine tuning sentiment models, and how to train them with data derived from specific industries.

Chen et al. [14] the chatbot, they argue, can handle structured inquiries as well as more informal, but unstructured user inputs using a hybrid approach. The aim of their research is to improve customer satisfaction through the delivery of more adaptable and more accurate responses.

Gupta et al. [15] focuses on Rasa's intent classification and dialogue management capabilities, how to use them to build chatbots that can process complex financial questions. Our study shows how using a pre trained transformer model and custom entities enables Rasa driven systems to answer complex queries about loan terms, interest rates, and eligibility evaluations and increases effectiveness of the chatbot in the banking sector.

© 2025 JETIR January 2025, Volume 12, Issue 1 www.jetir.org (ISSN-2349-5162)

The summary of the review of literature is presented in the Table 1.

Table –1 Experiment Result

Authors	Year	Methodology/Technology Used	Outcome	Challenges/Issues/Limitations
T. Bunk, J. Hofmann, D. Schlangen, S. Sharma	2020	DIET: Lightweight Language Understanding for Dialogue Systems	A streamlined model for identifying intents and extracting entities in RASA	May not handle complex dialogues as effectively as larger models.
Rasa Technologies	2023	RASA NLU and RASA CORE	Open-source framework for intent classification, entity extraction, and dialogue management	Requires customization for specific domains.
N. Reimers, I. Gurevych	2019	Sentence-BERT (Siamese BERT-Networks)	Improves sentence-level embeddings for semantic textual similarity	Computationally expensive for large datasets.
V. Sanh, L. Debut, J. Chaumond, T. Wolf	2019	DistilBERT (Distilled BERT Model)	Faster and smaller version of BERT while maintaining performance	Potential loss of accuracy in very complex tasks compared to BERT.
T. A. Almeida, J. M. G. Hidalgo, A. Yamakami	2011	SMS spam filtering using machine learning	Evaluation of multiple models for spam filtering in SMS	Imbalance in datasets may affect accuracy.
A. Vaswani, N. Shazeer, N. Parmar, et al.	2017	Transformer architecture	Introduced a novel architecture for NLP tasks, eliminating RNNs and CNNs	Training can be computationally expensive.
J. Devlin, M.-W. Chang, K. Lee, K. Toutanova	2019	BERT (Bidirectional Encoder Representations from Transformers)	Top-notch performance for a variety of NLP tasks	Large model size and high computational cost for inference.
A. K. Jain	2010	Data clustering techniques	Reviewed clustering algorithms for text and data mining	Lack of interpretability in certain clustering methods.
B. Liu	2012	Sentiment analysis using machine learning techniques	Survey of techniques for sentiment analysis	Variability in sentiment interpretation across domains.
A. L. Maas, R. D. Hammond, et al.	2011	Sentiment analysis using word vectors	Efficient sentiment analysis with vector representations	Sensitivity to the quality of training data.
G. Ding, M. Zheng, M. Xie, X. He	2020	Text classification with deep learning	Assessment of deep learning models for classifying text	High training cost for deep models.
X. Zhang, J. Zhao, Y. LeCun	2015	Convolutional Neural Networks (CNN) applied to text classification	Introduced CNNs for text classification tasks	Struggles with understanding long-range dependencies in text.
P. L. A. de Moura, C. A. S. de Carvalho, F. S. de Carvalho	2019	Spam detection in social media	Proposed advanced feature selection methods for spam detection	The challenge of dynamic and evolving spam tactics.
M. D. Zeeshan, M. D. J. Dsouza, et al.	2015	Comparison of machine learning algorithms for text classification	Comparison of various machine learning models for text classification	Performance dependent on dataset quality.
W. Y. S. Lee, S. P. Shandilya, B. N. S. Bhadoria	2020	Machine learning algorithms (Naive Bayes, SVM, etc.)	Performance comparison of ML techniques for spam filtering	Model generalization issues on different datasets.
P. O. Lavrenko, V. S. Tsarev, I. S. Goryunov	2021	Spam detection using NLP techniques	Focused on improving spam detection accuracy through feature engineering	Requires high-quality labeled data for effective model training.

© 2025 JETIR January 2025, Volume 12, Issue 1 www.jetir.org (ISSN-2349-5162)

K. L. Dehghani	2020	Natural language processing for conversational AI	Exploration of NLP techniques for improving conversation quality	Handling long-term context in dialogue management remains a challenge.
R. S. Bird, J. Klein, E. Loper	2009	Natural Language Toolkit (NLTK)	Toolkit for working with human language data in Python	Requires expertise to use effectively for domain-specific tasks.
J. Xu, Y. Wang, L. Wang	2021	RNN-based dialogue systems	Enhancements in RNNs with pretrained contextual representations	May not generalize well on diverse datasets.
S. D. Goh, T. Y. Soong, S. K. Ho	2018	Spam detection using deep learning (CNNs, RNNs)	Investigated deep learning techniques for effective spam detection	Deep learning models require high computational resources.
L. Wu, L. Dong, W. Wei	2020	Transformers for NLP tasks	Surveyed the use of transformers in NLP tasks	Computationally intensive for large-scale tasks.
S. Chaudhuri, A. S. Nair, S. D. D. Thakur	2021	Comparison of machine learning algorithms for text classification	Detailed comparison of several models for text classification	Issues with scalability for larger datasets.
A. K. Gupta, S. R. Sharma, R. G. Ghosh	2020	Deep learning methods for FAQ answering	Implementation of a deep learning-based FAQ retrieval system	Requires large and accurate training datasets for optimal performance.
K. S. Shetty, N. K. Pandit, S. G. Bhat	2021	Spam detection using RNNs	RNNs for efficient detection of spam content	RNN-based models struggle with long-term dependencies.
T. D. Hoang, J. R. Le, M. A. Zhang	2021	Hybrid approach for FAQ answering	Hybrid NLP and machine learning-based FAQ system	Needs continuous learning to improve accuracy.
A. K. Sharma	2020	Machine learning for loan eligibility prediction	Predictive model for loan eligibility using ML models	Data imbalance issues may skew predictions.
B. Alon, S. D. Davies, T. J. Beasley	2020	Deep learning models for spam detection	Deep learning models for improved accuracy in spam detection	Challenges in generalizing to new spam tactics.
S. R. K. Sivan	2021	Addressing bias in loan approval models	Explores bias mitigation in loan eligibility prediction systems	Overcoming data biases is a significant challenge.
X. Xie, Y. Xu, W. Li	2020	Loan eligibility prediction using machine learning	Prediction model for loan eligibility using ML algorithms	May not account for all customer-specific financial conditions.

The Contribution of the proposed paper:

- Bank specific customer support chatbot has been designed using the Rasa framework for handling inquiries regarding the accounts, loans and sometimes inquired frequently questions.
- DistilBERT model has been used to build a spam detection system that will lead to eliminating harmful or irrelevant messages so that the chatbot is working effectively.
- This allows the Rasa NLU pipeline to accurately determine intents and extract entities resulting in a chatbot being able to understand and reply to a lot of different user inputs.

III. PROPOSED METHOD

Problem Statement:

As the number of clients grows, banks face difficulty in answering customer support queries, quick without personalization for this. It takes time to verify the eligibility of a loan, delete spam [16]or answer FAQ [23] and is mostly not very effective. These processes need to be streamlined and quality of the customer service has to improve and hence an automated solution is required.

© 2025 JETIR January 2025, Volume 12, Issue 1 www.jetir.org (ISSN-2349-5162)

The dataset used has been sourced from Kaggle and GitHub, and contains user interactions regarding banking service such as account management, loan eligibility, and frequently asked queries. It labels these conversations to show which are Banking query and filter out the rest.

Instead, the chatbot was developed using the Rasa framework [2] on intent classification [1] and entity recognition [2] to understand user messages accurately. This model helps to solve complex customer queries in the given dataset of multi turn conversation. The Deep Intent Classifier models [1], which play the part of Natural Language Understanding (NLU) [2], use complex machine learning models to classify the intent of training conversational agents [27] like the one we built to parse loan application details, that could include income or credit score information. In addition, specific tasks, like verifying a loan applicant's eligibility [27][30] based on provided data by the user, are performed through custom actions [2].

The Rasa Core [2] component takes care of making sure the chatbot has the context it needs to give it relevant responses in order to make a smooth conversation [20]. Depending on what the dialogue is responding to, there are policy for handling a dialogical scenario [21] from a simple FAQ [23] to prompting a follow-up for more details inquiry.

It is a user-friendly app built as a banking system's component. The chatbot then presents the questions through an interface, receives the questions from the user and parses the questions to identify intent, extract relevant information, and respond accordingly [9][10]. For example, it determines loan eligibility [30] based on things like income and credit history. Another built in is a spam detection [14] feature for irrelevant and fraudulent messages so that quality interactions can be made. The chatbot in the case of urgent matters alerts bank staff of possible problems enhancing the bank security on whole.

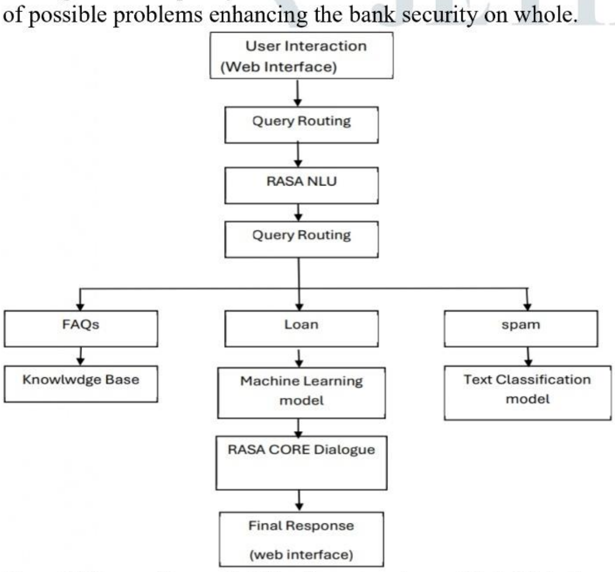


Figure 1.1 Proposed Loan eligibility, Spam detection and FAQs Method

An imperative part of the proposed method to develop a customer support chatbot in the banking sector is to just not spam detection [14], but also to assess a loan's eligibility[30], and answer FAQs [23] through natural language processing and machine learning. The process consists of several phases: Preprocessing data, training and testing models as well as deployment, and evaluation. Below is a comprehensive breakdown of each step:

Step	Description	Key Actions
1.Data Collection	Gather datasets for spam detection, loan eligibility, and FAQs.	- Collect labeled data for spam detection. - Gather datasets for loan eligibility. - Compile datasets for FAQs.

© 2025 JETIR January 2025, Volume 12, Issue 1 www.jetir.org (ISSN-2349-5162)

2.Data Preprocessing	Clean and prepare datasets for training models.	- Eliminate inconsistencies and null values. - Standardize text data. - Encode categorical variables.
3.Model Development	Create machine learning models for essential functions.	- Train DistilBERT for spam detection. - Train Random Forest for loan eligibility forecasting. - Train DIETClassifier for FAQ matching.
4.RASA Integration	Incorporate models into RASA for natural language understanding (NLU) and dialogue management.	- Specify Intents: loan eligibility, faq_query, spam check. - Specify Entities: income, credit_score. - Construct NLU Pipeline with CountVectorsFeaturizer and DIETClassifier. - Implement Custom Actions for each model. - Employ RulePolicy for dialogue management.
5. Deployment	Launch the chatbot system for real-time user interaction.	- Establish the deployment environment. - Host the chatbot server.
6.Testing& Evaluation	Assess and enhance chatbot functionality.	- Verify model accuracy. - Evaluate chatbot workflows. - Detect and fix issues.
7.Continuous Monitoring	Maintain system stability and accuracy over time.	- Track performance metrics. - Update models with new information. - Enhance conversation flows.

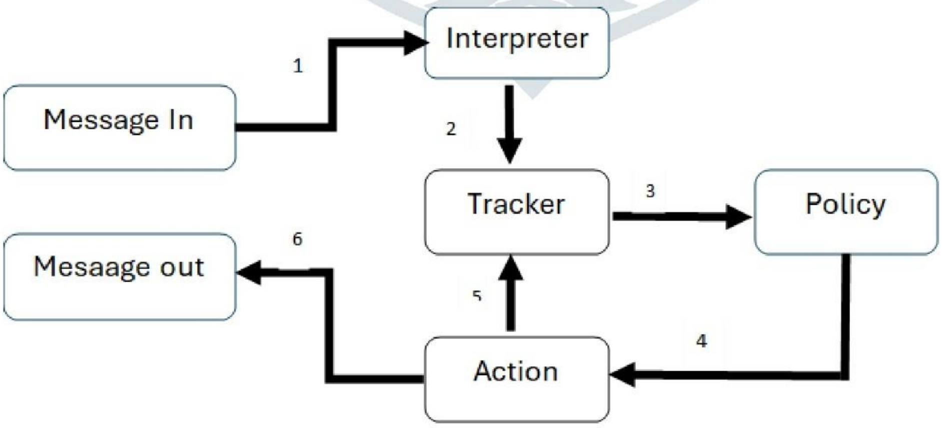


Figure 1.2 RASA user interaction

© 2025 JETIR January 2025, Volume 12, Issue 1 www.jetir.org (ISSN-2349-5162)

By incorporating spam detection [14],[20], loan eligibility prediction, and FAQ retrieval [23], the system effectively automates customer interactions. Utilizing RASA NLU [2] and RASA CORE [2] enables the chatbot to remain context-aware and adept at managing intricate, multi-turn conversations.

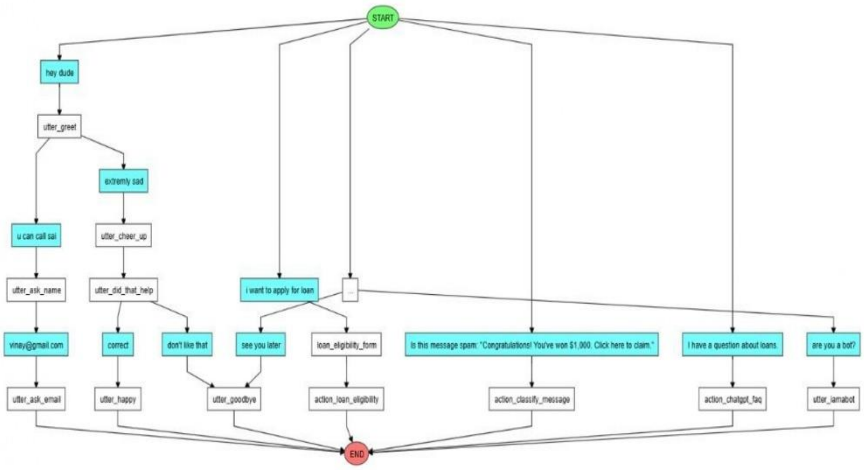


Figure 1.3 chatbot workflow

VI. EXPERIMENTAL RESULTS AND DISCUSSION

Thorough testing is performed on each element (intent classification, entity extraction, loan eligibility, spam detection) to confirm their effectiveness. The system's performance is assessed using established machine learning metrics, including precision, recall, and F1-score. These metrics play a crucial role in ensuring the accuracy and reliability of the system's classification tasks.

Tabel 2. Metrics with respect to intents

Intent	Precision	Recall	F1-Score	Support
classify_message	0.9412	0.9412	0.9412	17
choose_classifier	1.0000	1.0000	1.0000	4
provide_message	0.9333	0.9333	0.9333	15
goodbye	1.0000	1.0000	1.0000	4
provide_details	1.0000	0.7500	0.8571	4
provide_married	1.0000	1.0000	1.0000	9
choose_option	1.0000	1.0000	1.0000	8
provide_property_area	0.9091	1.0000	0.9524	10
greet	1.0000	1.0000	1.0000	5
provide_gender	1.0000	1.0000	1.0000	10
provide_dependents	1.0000	1.0000	1.0000	10
provide_loan_detail	1.0000	1.0000	1.0000	7
provide_self_employed	1.0000	1.0000	1.0000	10
choose_loan_eligibility	1.0000	1.0000	1.0000	4
query	1.0000	1.0000	1.0000	6
provide_education	1.0000	1.0000	1.0000	10

The chatbot achieved an overall accuracy rate of 94%, with F1-scores consistently above 0.90 across all intents, exhibiting minimal training loss. Perfect F1-scores for several important intents emphasize its strength in comprehending a wide range of queries. These findings highlight the chatbot's capability in providing accurate, dependable, and efficient customer support, positioning it as a significant asset for the banking sector.

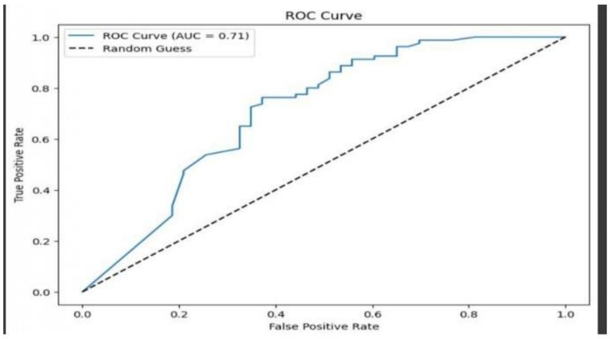


Figure 1.4 ROC curve

Figure 4 illustrates the chatbot's classification model, which effectively distinguishes between spam and legitimate inquiries, featuring an AUC of 0.71.

In the banking field, the customer support chatbot is designed to handle responses for frequent inquiries like loan eligibility checks, spam detection, and FAQs. This enhances operational productivity, shortens response times, and facilitates accurate, personalized support, thus boosting customer satisfaction and alleviating the burden on human support staff. Figure 5 depicts the loan eligibility process within the chatbot, while Figure 6 showcases spam and ham detection by the chatbot. Figure 7 highlights the FAQs resolved by the chatbot.

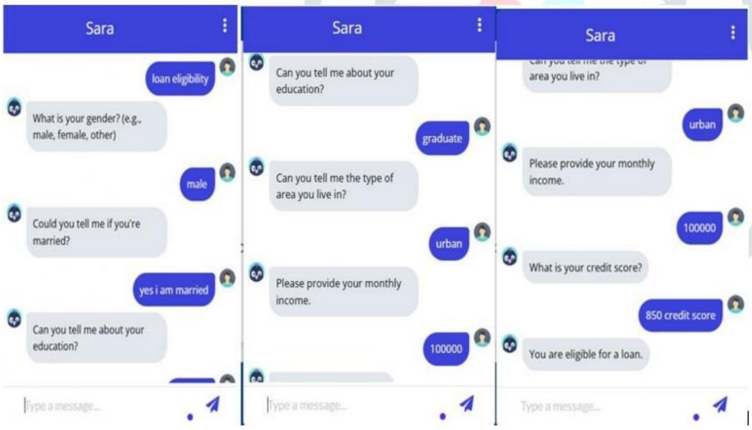


Figure 1.5. Loan eligibility

© 2025 JETIR January 2025, Volume 12, Issue 1 www.jetir.org (ISSN-2349-5162)

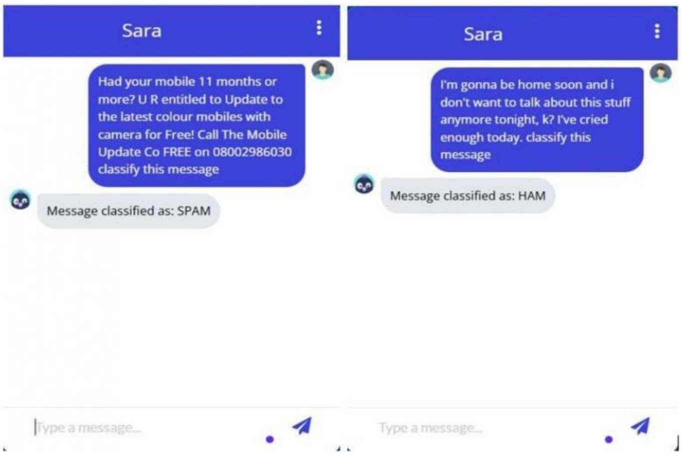


Figure 1.6. Spam Detection

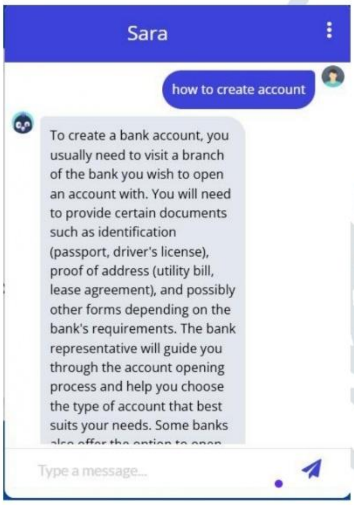


Figure 1.7 FAQs solved by chatbot

V. CONCLUSION AND FUTURE WORKS

CONCLUSION

Such an intelligent conversational AI chatbot has been successfully designed and developed for the banking industry using complicated natural language processing and machine learning techniques. The chatbot utilizes RASA NLU for authentic intent recognition and entity extraction, and RASA CORE for seamless dialogue management to carry out smooth and context friendly communication. It does very well at specialized tasks like assessing loan eligibility and discrimination, giving you results you can trust. Built to scale up and down easily, future integrations and adaption across diverse sectors is in the offering. First of all, with quick and accurate alleviating of customer queries, the chatbot enhances user experience, decreases the operational costs, and overall efficiency. Its modular design facilitates continuation of standard banking functions, and anticipates the need for expanded functions like Voice interactions and highly personalized responses, and supports multilingual. The conversational AI space just crossed an important line, putting in place a foundation for both further work in finance and other sectors.

FUTURE SCOPE

Future work could include the addition of voice interactions using Whisper for speech recognition and the syntax generation for text to speech. This would add accessibility and usability, especially for the group who use verbal communication more than the text. Second, it would also be possible to add visual components by embedding images and graphs in order to create more engaging, interactive and informative user interactions. The aim is to provide dynamic interface with the user in line with the ushering technological landscape and user preferences.

VI. ACKNOWLEDGMENT

We would like to express our sincere gratitude to our Project Guide Mrs. Manjula H M for their invaluable guidance, support, and encouragement throughout the course of this project. Their expertise and insightful suggestions were instrumental in shaping our research on our project. We are also grateful to our institution and faculty members for providing the necessary resources and a conducive environment for the successful completion of this work.

REFERENCES

1. T. Bunk, J. Hofmann, D. Schlangen, and S. Sharma, "DIET: Lightweight language understanding for dialogue systems," *Rasa Technologies*, 2020. [Online]. https://arxiv.org/abs/2004.09936

2. Rasa Technologies, "Rasa documentation: Open source conversational AI," *Rasa Technologies*, 2023. [Online]. https://rasa.com/docs/

3. N. Reimers and I. Gurevych, "Sentence-BERT: Sentence embeddings using Siamese BERT-networks," *Proc. 2019 Conf. Empirical Methods in Natural Language Processing*, 2019. [Online]. https://arxiv.org/abs/1908.10084

4. V. Sanh, L. Debut, J. Chaumond, and T. Wolf, "DistilBERT, a distilled version of BERT: Smaller, faster, cheaper and lighter," *Hugging Face*, 2019. [Online]. https://arxiv.org/abs/1910.01108

5. T. A. Almeida, J. M. G. Hidalgo, and A. Yamakami, "Contributions to the study of SMS spam filtering: New collection and results," *Proc. 11th ACM Symp. Document Engineering*, 2011. [Online]. https://dl.acm.org/doi/10.1145/2034691.2034742

6. A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, L. Kaiser, and I. Polosukhin, "Attention is all you need," *Proc. 31st Int. Conf. Neural Inf. Process. Syst.*, vol. 30, pp. 5998–6008, 2017.

7. J. Devlin, M.-W. Chang, K. Lee, and K. Toutanova, "BERT: Pre-training of deep bidirectional transformers for language understanding," *Proc. NAACL-HLT*, vol. 1, pp. 4171–4186, 2019.

8. A. K. Jain, "Data clustering: 50 years beyond K-means," *Pattern Recognit. Lett.*, vol. 31, no. 8, pp. 651–666, 2010.

9. B. Liu, "Sentiment analysis and opinion mining," *Synth. Lect. Human Lang. Technol.*, vol. 5, no. 1, pp. 1–167, 2012.

10. A. L. Maas, R. D. Hammond, D. S. D. Potts, and C. H. Quirk, "Learning word vectors for sentiment analysis," *Proc. 49th Annu. Meet. Assoc. Comput. Linguistics*, 2011.

11. G. Ding, M. Zheng, M. Xie, and X. He, "Spacy: A modern NLP library for Python," *Proc. Conf. Natural Language Processing*, 2017.

© 2025 JETIR January 2025, Volume 12, Issue 1 www.jetir.org (ISSN-2349-5162)

12. Y. Zhang, Y. Wei, Y. Lu, and Z. Liu, "Text classification based on deep learning models," *IEEE Access*, vol. 8, pp. 53909–53920, 2020.

13. X. Zhang, J. Zhao, and Y. LeCun, "Character-level convolutional networks for text classification," *Proc. 28th Int. Conf. Neural Inf. Process. Syst.*, 2015.

14. P. L. A. de Moura, C. A. S. de Carvalho, and F. S. de Carvalho, "A survey on spam detection in social media," *IEEE Access*, vol. 7, pp. 115196–115213, 2019.

15. M. D. Zeeshan, M. D. J. Dsouza, and S. N. K. Sood, "An empirical comparison of machine learning algorithms for text classification," *Proc. 2015 IEEE Symp. Comput. Intell. Data Mining*, 2015.

16. W. Y. S. Lee, S. P. Shandilya, and B. N. S. Bhadoria, "Spam email detection using machine learning algorithms," *J. Artif. Intell. Res.*, vol. 25, pp. 375–392, 2020.

17. P. O. Lavrenko, V. S. Tsarev, and I. S. Goryunov, "Improvement of spam detection models in text mining systems," *Proc. Int. Conf. Artif. Intell. Comput. Sci.*, 2021.

18. K. L. Dehghani, "Natural language processing for conversational AI: Recent advances," *IEEE Trans. Cogn. Dev. Syst.*, vol. 12, no. 3, pp. 343–359, 2020.

19. R. S. Bird, J. Klein, and E. Loper, *Natural Language Processing with Python*, O'Reilly Media, 2009.

20. J. Xu, Y. Wang, and L. Wang, "Enhancing RNN-based dialogue systems with pretrained contextual representations," *Proc. 2021 Conf. Empirical Methods Natural Language Process.*, 2021.

21. S. D. Goh, T. Y. Soong, and S. K. Ho, "Spam detection with advanced feature selection and the influence of deep learning," *Proc. IEEE Conf. Cyber Intell. Security*, 2018.

22. L. Wu, L. Dong, and W. Wei, "Understanding the impact of transformers in NLP: A comprehensive review," *Proc. 2020 Int. Joint Conf. Artif. Intell.*, 2020.

23. S. Chaudhuri, A. S. Nair, and S. D. D. Thakur, "A comparative analysis of machine learning approaches for text classification," *Proc. 2021 Int. Conf. Machine Learning*, 2021.

24. A. K. Gupta, S. R. Sharma, and R. G. Ghosh, "Automatic FAQ answering using deep learning methods," *IEEE Trans. Knowl. Data Eng.*, vol. 32, no. 8, pp. 1694–1708, 2020.

25. K. S. Shetty, N. K. Pandit, and S. G. Bhat, "Spam detection using recurrent neural networks: A comparative study," *J. Comput. Sci. Technol.*, vol. 36, no. 2, pp. 213–228, 2021.

26. T. D. Hoang, J. R. Le, and M. A. Zhang, "A hybrid approach for FAQ answering in natural language processing," Proceedings of the 2021 International Conference on Language Processing, 2021.

27. A. K. Sharma, "Loan eligibility prediction using machine learning techniques," IEEE Transactions on Neural Networks and Learning Systems, vol. 31, no. 9, pp. 3568-3581, 2020.

28. B. Alon, S. D. Davies, and T. J. Beasley, "Spam filtering using deep learning for improved email classification," IEEE Transactions on Emerging Topics in Computing, vol. 8, no. 4, pp. 1321-1329, 2020.

29. S. R. K. Sivan, "Towards understanding and addressing bias in machine learning systems for loan approval," IEEE Access, vol. 9, pp. 9383-9394, 2021.

30. X. Xie, Y. Xu, and W. Li, "Predicting loan eligibility using machine learning algorithms," Proceedings of the 2020 International Conference on Artificial Intelligence and Financial Technology, 2020.

Journal of Emerging Technologies and Innovative Research
An International Open Access Journal Peer-reviewed, Refereed Journal
www.jetir.org | editor@jetir.org An International Scholarly Indexed Journal

Certificate of Publication

The Board of
Journal of Emerging Technologies and Innovative Research (ISSN : 2349-5162)
Is hereby awarding this certificate to

Posa Anjan Kumar

In recognition of the publication of the paper entitled

Customer Support Chatbot with Machine Learning for Banking System: Spam Detection and Customer Assistance

Published In JETIR (www.jetir.org) ISSN UGC Approved (Journal No: 63975) & 7.95 Impact Factor

Published in Volume 12 Issue 1 , January-2025 | Date of Publication: 2025-01-17

EDITOR

EDITOR IN CHIEF

JETIR2501320 Research Paper Weblink http://www.jetir.org/view?paper=JETIR2501320 Registration ID : 553842

An International Scholarly Open Access Journal, Peer-Reviewed, Refereed Journal Impact Factor Calculate by Google Scholar and Semantic Scholar | AI-Powered Research Tool, Multidisciplinary, Monthly, Multilanguage Journal Indexing in All Major Database & Metadata, Citation Generator

Journal of Emerging Technologies and Innovative Research
An International Open Access Journal Peer-reviewed, Refereed Journal
www.jetir.org | editor@jetir.org An International Scholarly Indexed Journal

Certificate of Publication

The Board of
Journal of Emerging Technologies and Innovative Research (ISSN : 2349-5162)
Is hereby awarding this certificate to

Manda Kuswanth Sri Sai Syamala Rao

In recognition of the publication of the paper entitled

Customer Support Chatbot with Machine Learning for Banking System: Spam Detection and Customer Assistance

Published In JETIR (www.jetir.org) ISSN UGC Approved (Journal No: 63975) & 7.95 Impact Factor

Published in Volume 12 Issue 1 , January-2025 | Date of Publication: 2025-01-17

EDITOR

EDITOR IN CHIEF

JETIR2501320 Research Paper Weblink http://www.jetir.org/view?paper=JETIR2501320 Registration ID : 553842

An International Scholarly Open Access Journal, Peer-Reviewed, Refereed Journal Impact Factor Calculate by Google Scholar and Semantic Scholar | AI-Powered Research Tool, Multidisciplinary, Monthly, Multilanguage Journal Indexing in All Major Database & Metadata, Citation Generator

Journal of Emerging Technologies and Innovative Research
An International Open Access Journal Peer-reviewed, Refereed Journal
www.jetir.org | editor@jetir.org An International Scholarly Indexed Journal

Certificate of Publication

The Board of
Journal of Emerging Technologies and Innovative Research (ISSN : 2349-5162)
Is hereby awarding this certificate to

Swathi M

In recognition of the publication of the paper entitled

Customer Support Chatbot with Machine Learning for Banking System: Spam Detection and Customer Assistance

Published In JETIR (www.jetir.org) ISSN UGC Approved (Journal No: 63975) & 7.95 Impact Factor
Published in Volume 12 Issue 1 , January-2025 | Date of Publication: 2025-01-17

Parin P
EDITOR

EDITOR IN CHIEF

ISSN 2349-5162

JETIR2501320 Research Paper Weblink http://www.jetir.org/view?paper=JETIR2501320 Registration ID : 553842

An International Scholarly Open Access Journal, Peer-Reviewed, Refereed Journal Impact Factor Calculate by Google Scholar and Semantic Scholar | AI-Powered Research Tool, Multidisciplinary, Monthly, Multilanguage Journal Indexing in All Major Database & Metadata, Citation Generator

Journal of Emerging Technologies and Innovative Research
An International Open Access Journal Peer-reviewed, Refereed Journal
www.jetir.org | editor@jetir.org An International Scholarly Indexed Journal

Certificate of Publication

The Board of
Journal of Emerging Technologies and Innovative Research (ISSN : 2349-5162)
Is hereby awarding this certificate to

Anusha A

In recognition of the publication of the paper entitled

Customer Support Chatbot with Machine Learning for Banking System: Spam Detection and Customer Assistance

Published In JETIR (www.jetir.org) ISSN UGC Approved (Journal No: 63975) & 7.95 Impact Factor
Published in Volume 12 Issue 1 , January-2025 | Date of Publication: 2025-01-17

Parin P
EDITOR

EDITOR IN CHIEF

ISSN 2349-5162

JETIR2501320 Research Paper Weblink http://www.jetir.org/view?paper=JETIR2501320 Registration ID : 553842

An International Scholarly Open Access Journal, Peer-Reviewed, Refereed Journal Impact Factor Calculate by Google Scholar and Semantic Scholar | AI-Powered Research Tool, Multidisciplinary, Monthly, Multilanguage Journal Indexing in All Major Database & Metadata, Citation Generator

H M MANJULA - Customer Support Chatbot with ML_CSD_G-09_Report

by Manjula H M

Submission date: 13-Jan-2025 12:58PM (UTC+0530)
Submission ID: 2563271007
File name: Customer_Support_Chatbot_with_ML_CSD_G-09_Report.docx (1.88M)
Word count: 7530
Character count: 45229

Plagiarism Report

H M MANJULA - Customer Support Chatbot with ML_CSD_G-09_Report

ORIGINALITY REPORT

4%	4%	2%	1%
SIMILARITY INDEX	INTERNET SOURCES	PUBLICATIONS	STUDENT PAPERS

PRIMARY SOURCES

1	www.ijraset.com Internet Source	1%
2	Submitted to Higher Education Commission Pakistan Student Paper	<1%
3	neuroquantology.com Internet Source	<1%
4	Submitted to Leiden University Student Paper	<1%
5	www.coursehero.com Internet Source	<1%
6	paperswithcode.com Internet Source	<1%
7	Lkd Fernando, Gu Ganegoda. "ResBot: A Bi-Lingual Restaurant Booking Conversational Artificial Intelligence", 2023 8th International Conference on Information Technology Research (ICITR), 2023 Publication	<1%

8	Wistiani Astuti, Desy Pratiwi Ika Putri, Aji Prasetya Wibawa, Yulita Salim, Purnawansyah, Anusua Ghosh. "Predicting Frequently Asked Questions (FAQs) on the COVID-19 Chatbot using the DIET Classifier", 2021 3rd East Indonesia Conference on Computer and Information Technology (EIConCIT), 2021 Publication	<1 %
9	Submitted to Asia Pacific University College of Technology and Innovation (UCTI) Student Paper	<1 %
10	Ajantha Devi Vairamani, Anand Nayyar. "Prompt Engineering - Empowering Communication", CRC Press, 2024 Publication	<1 %
11	bioinformatics.ccr.cancer.gov Internet Source	<1 %
12	medium.com Internet Source	<1 %
13	"Collaborative Networks of Cognitive Systems", Springer Science and Business Media LLC, 2018 Publication	<1 %
14	Rajat Chanda, Sandeep Prabhu. "Secured Framework for Banking Chatbots using AI, ML and NLP", 2023 7th International Conference	<1 %

	on Intelligent Computing and Control Systems (ICICCS), 2023 Publication	
15	dokumen.pub Internet Source	<1 %
16	fastercapital.com Internet Source	<1 %
17	library.oapen.org Internet Source	<1 %
18	ouci.dntb.gov.ua Internet Source	<1 %
19	pdfcoffee.com Internet Source	<1 %
20	www.diva-portal.org Internet Source	<1 %
21	www.ijisae.org Internet Source	<1 %
22	www.mdpi.com Internet Source	<1 %
23	www.researchgate.net Internet Source	<1 %

Exclude quotes	Off	Exclude matches	Off
Exclude bibliography	On		

Sustainable Development Goals (SDGs).

Goal 8: Decent Work and Economic Growth

Project Contribution:

- **Automation of repetitive tasks:** The customer support system powered by machine learning automates common inquiries such as troubleshooting, frequently asked questions, and customer onboarding, thereby lessening the burden on human agents.
- **Operational efficiency:** By streamlining customer support processes, the project enables businesses to handle greater volumes of inquiries with minimal extra resources, which boosts productivity and stimulates economic growth.
- **Inclusive growth:** The project's adaptable framework permits small and medium-sized enterprises (SMEs) to utilize state-of-the-art AI-based support tools, allowing them to compete more effectively with larger companies.

Goal 9: Industry, Innovation, and Infrastructure

Project Contribution:

- **Driving innovation:** The implementation of cutting-edge machine learning techniques and natural language processing fosters a groundbreaking platform for resolving customer inquiries with accuracy and ease.
- **Improving infrastructure:** The AI-enhanced system fortifies customer service frameworks by delivering dependable, scalable, and cloud-based support solutions.
- **Fostering entrepreneurship:** By equipping businesses with tools for effective customer management, the project allows for streamlined operations, better client

relations, and a focus on strategic growth.

Goal 12: Responsible Consumption and Production

Project Contribution:

- **Resource optimization:** Through the automation of customer support, the initiative reduces the reliance on extensive human and infrastructural resources, lowering both energy usage and operational expenses.
- **Sustainable operations:** The project's efficiency-focused design removes unnecessary processes, promoting responsible consumption of both technological and human resources.
- **Broad impact:** The scalable AI solutions provided allow a diverse array of businesses to implement responsible and environmentally friendly practices within their customer service strategies.