VISWAJYOTHI
COLLEGE OF ENGINEERING & TECHNOLOGY
Approved by AICTE New Delhi & Affiliated to APJ Abdul Kalam Technological University
Vazhakulam P.O., Muvattupuzha
Ernakulam Dist., Kerala - 686 670
Tel: 0485 2262211 / 44
Email:vjcet@vjcet.org
www.vjcet.org
B.TECH PROGRAMMES ACCREDITED BY NATIONAL BOARD OF ACCREDITATION
NAAC

NexusAssist

GROUP 6

NIHAL THOMAS (VJC20AD035)
ROHAN RAJ (VJC20AD041)

PROJECT GUIDE:

MRS GEETHU GOPAN,
ASSISTANT PROFESSOR

ARTIFICIAL INTELLIGENCE AND
DATA SCIENCE

CONTENTS

INTRODUCTION

- AI powered solution.
- The project aims to deal with customer support section in a sector.
- It can handle general inquries and requests.
- This project require a prompt engineered model to deliver more reliable response and user friendly interface
- The project is divided into two phases.

PHASE 1

- The first phase of the project include the development of prompt engineered model to provide appropriate response and a user friendly interface for the project.

PHASE 2

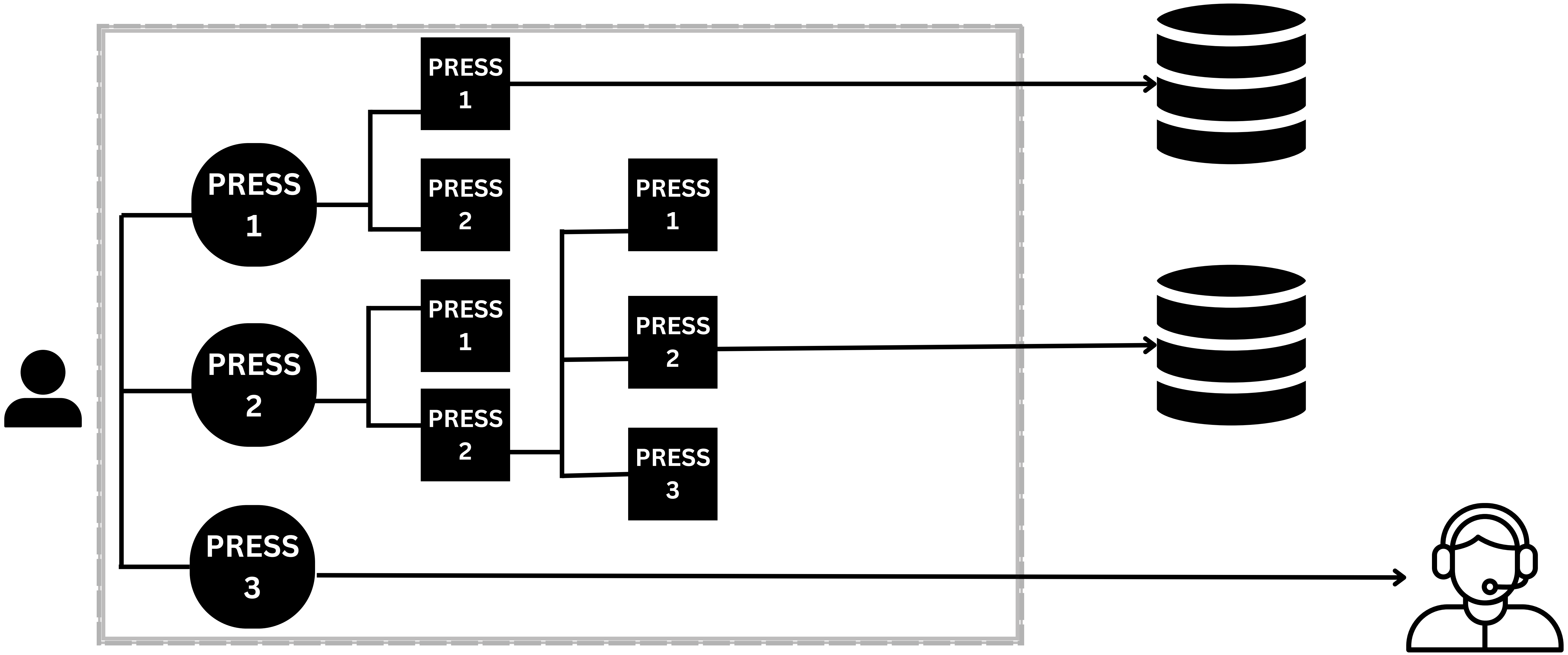
- Intergrate the chat bot with aws connect to deploy for voice communication through telephone and to retrieve required data from DynamoDB using Lambda fucntion.

PROBLEM STATEMENT

- **High Volume of Routine Inquiries asked on a daily basis**
- **Inconsistent Responses from customer care executives**
- **Limited Availability from the agents**

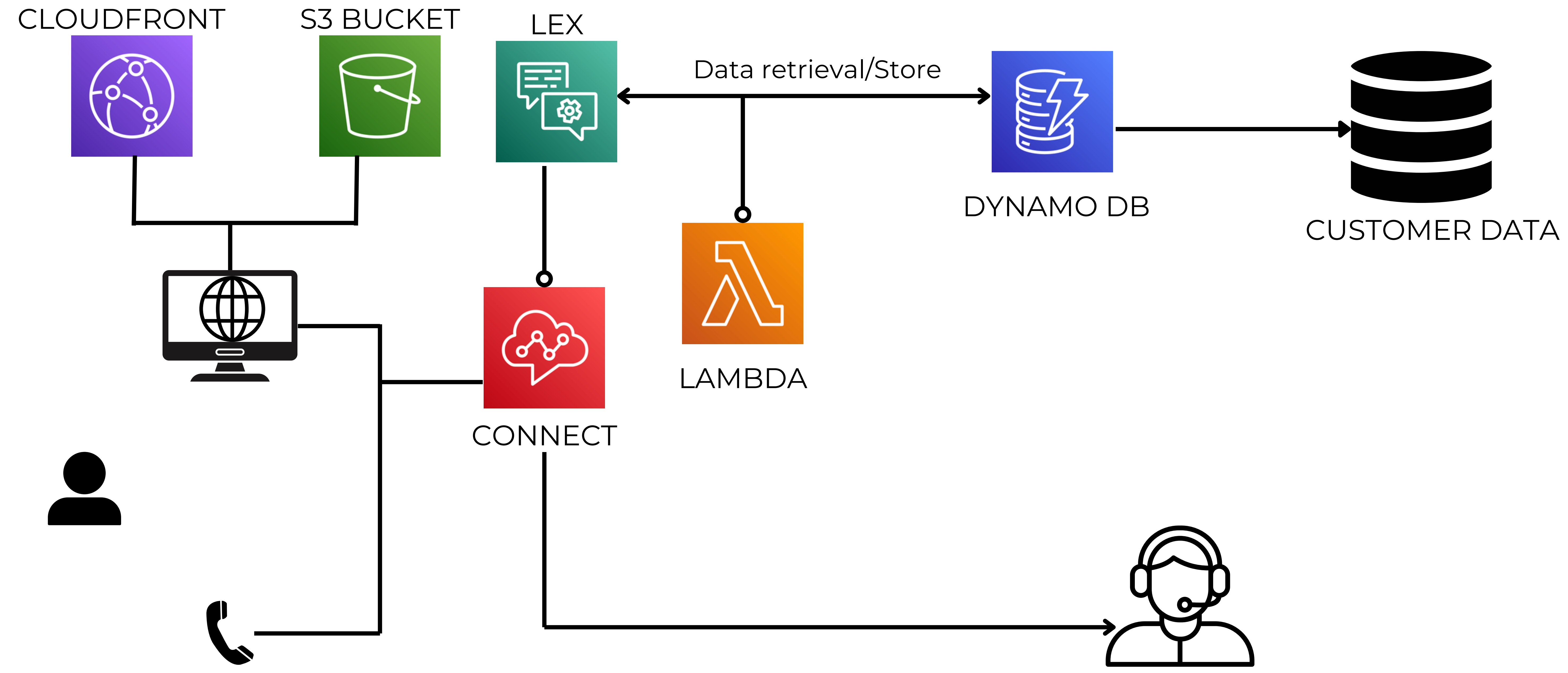
The project aims to optimize resource allocation and reduce operational costs by automating routine inquiries and reducing human agent workload, while also addressing the variability in human responses that can lead to inconsistent customer experiences and potentially inaccurate information. Additionally, it will provide 24/7 support through a voice-based system, ensuring customers have immediate assistance at all times, thus overcoming the limitation of traditional support hours.

FLOWCHART



TRADITIONAL IVR SYSTEM

FLOWCHART


CLOUDFRONT
S3 BUCKET
LEX
Data retrieval/Store
DYNAMO DB
CUSTOMER DATA
LAMBDA
CONNECT

METHODOLOGY

Amazon Lex Bot

It helps in understanding user intent and context by processing natural language input, allowing it to accurately identify what the user wants. It also helps to extract and fill slots from the user input to gather necessary information.

AWS Lambda

AWS Lambda function helps in extracting and storing data from DynamoDB. It helps to read data from a DynamoDB table, then process it and eventually update the table accordingly.

DynamoDB

DynamoDB, Amazon's NoSQL database service, can be used in a conversational IVR system to provide a fast, scalable, and reliable data storage solution for handling various types of user interactions.

Amazon Connect

Amazon Connect uses contact flows to define the behavior of an IVR system.

Contact flows are visual workflows that allows us to design and customize how our calls are routed, what prompts are played, and how data is collected

LITERATURE SURVEY

NO	NAME	METHODOLOGY	ADVANTAGES	DISADVANTAGES
1	Voice chatbot for hospitality[1]	Closed Domain Question Answering (cdQA) system with a Retriever and Reader components for answering questions from documents	• Efficient retrieval of relevant information • Integration with voice chatbot	• Domain-specific knowledge requirement • Limitations in understanding complex questions
2	Empirical analysis of the strengths and weaknesses of the PEFT techniques in LLM's[2]	Evaluate PEFT techniques on FLAN-T5-XL LLM, comparing their performance across datasets and tasks. Assess four fine-tuning methods: LoRA, $(IA)^3$, prompt tuning, and BitFit.	• Optimization and reduction of parameters. • Framework for choosing optimal fine-tuning techniques	• Slower convergence in low-data scenarios • Lack of clear optimal fine-tuning method

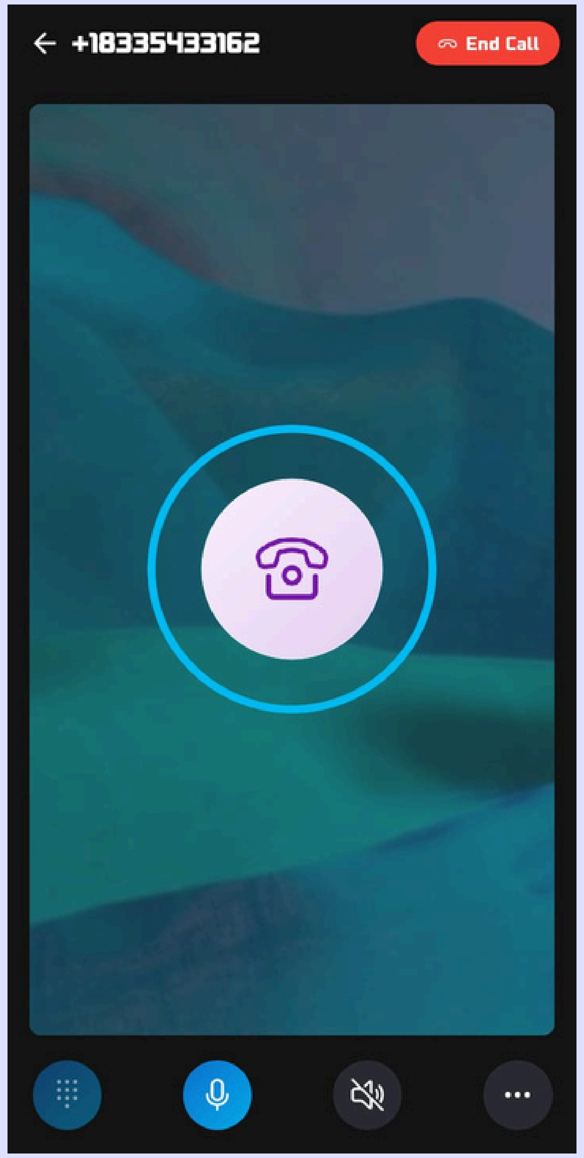
NO	NAME	METHODOLOGY	ADVANTAGES	DISADVANTAGES
3	Financial news analytics using fine tunes Llama 2 GPT model[3]	Llama 2 Large Language Model (LLM) fine-tuned with PEFT/LoRA for tasks such as financial news analysis, text summarization, and sentiment extraction from named entities.	• Fine-tuned LLM predicts responses for financial analytics. • Market perspective, summarization, sentiment extraction.	The accuracy of entity names in the JSON output may be problematic and require a more accurate and specified training dataset.
4	Idiolect: A Reconfigurable Voice Coding Assistant[4]	Idiolect handles inaccurate transcriptions by visually or verbally guiding users to choose actionable phrases, leveraging Tidyparse for error correction.	• Customizability • Natural language programming • Real-time speech recognition	• Recognition failure • Learning curve
5	Farmer’s Assistant using AI Voice Bot[5]	• Information Retrieval • Crops Suggestions • Fertilizer Suggestion	• Improved accessibility • Enhanced farming practices	• Weather Data Dependence • Language Limitations • Internet Dependency

NO	NAME	METHODOLOGY	ADVANTAGES	DISADVANTAGES
6	End-to-End Speech to Intent Prediction[6]	XGBoost, BERT, RNNs, lattice modeling, KenLM, pretrained transformers, ASR integration, data augmentation	• Higher accuracy • Simplified pipeline	• Availability of pre-training data • Dataset limitations
7	Generative Pre-trained Transformer for Biomedical Text Generation and Mining[7]	Covering biomedical text tasks like relation extraction, question answering, and document classification.	• Domain-specific • Improved performance • Fluent text generation	• Lack of comparison with other domain-specific models • Limited evaluation on other biomedical NLP tasks
8	Building an Effective Email Spam Classification Model with spaCy[8]	Naive Bayes, Decision Tree C45, and Multilayer Perceptron algorithms in Python to train and classify spam emails.	• High accuracy rate • Natural language processing • Multiple machine learning algorithms	• Limited dataset • Lack of comparison with other models

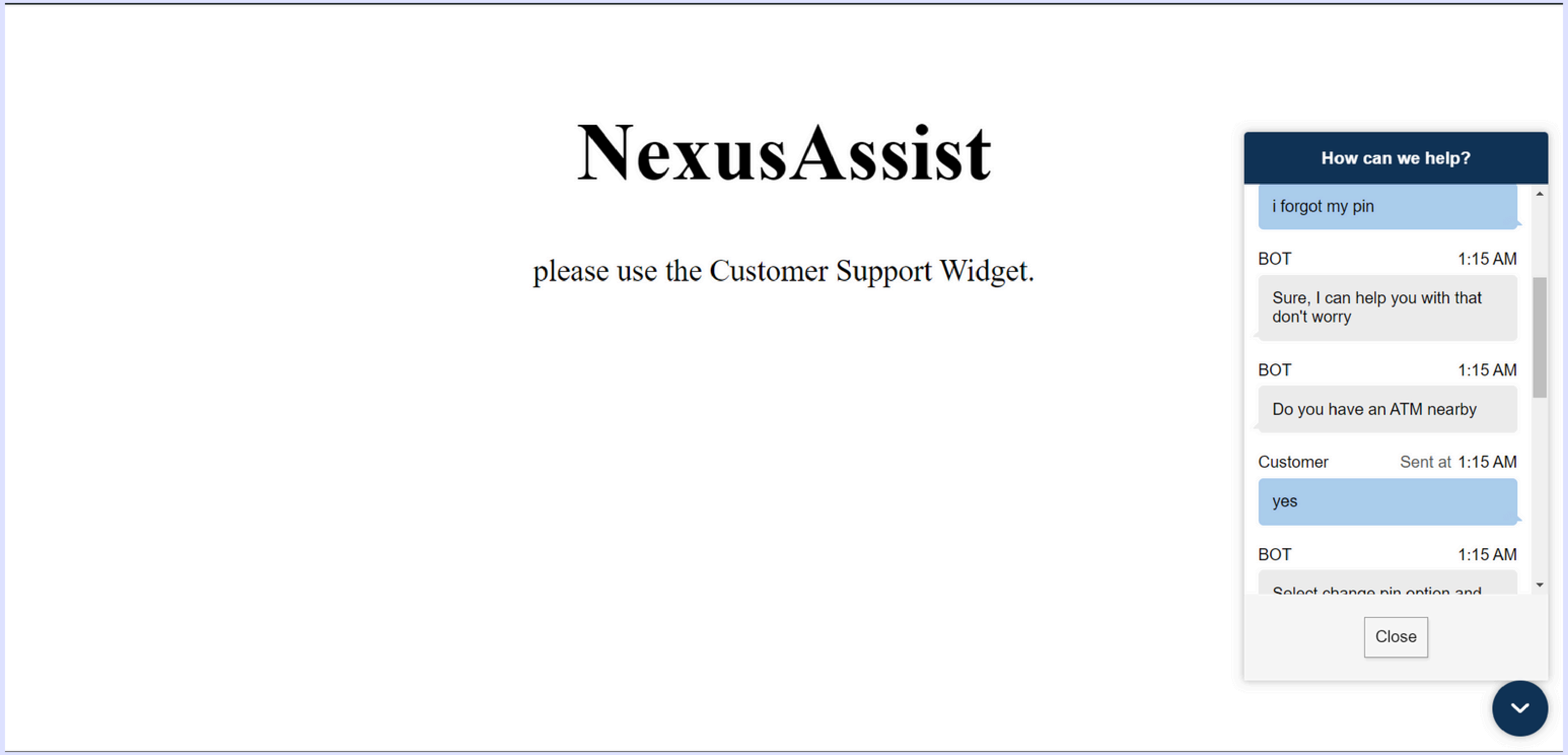
NO	NAME	METHODOLOGY	ADVANTAGES	DISADVANTAGES
9	Evaluation of Conversational Agents: Understanding Culture, Context and Environment in Emotion Detection[9]	• Convolutional Neural Network (CNN) for image and audio data. • Employing an Audio-Frame Mean Expression and addressing bias through data balancing, pre-processing, and augmentation.	• Real-time detection • Consideration of cultural diversity • Improved accuracy	• Possible misclassification • Need for further validation • Possible misclassification
10	On Adapting the DIET Architecture and the Rasa Conversational Toolkit for the Sentiment Analysis Task[10]	Sentiment analysis, including lexicon-based methods, corpus-based methods, machine learning approaches, and hybrid approaches	• Rasa and DIET enable easy sentiment analysis for non-experts. • Simplifies design and democratizes sentiment analysis technology.	• Computational cost increases with sentence length • Other existing methods might outperform proposed ones.

OUTPUT

PHONE



WEB



NexusAssist

please use the Customer Support Widget.

How can we help?

i forgot my pin

BOT 1:15 AM

Sure, I can help you with that don't worry

BOT 1:15 AM

Do you have an ATM nearby

Customer Sent at 1:15 AM

yes

BOT 1:15 AM

Select change pin option and

LIMITATIONS

Complex Queries/Requests

While conversational IVR systems can handle simple queries well, complex or nuanced questions may pose challenges for them. They might struggle to understand and process more sophisticated or multi-layered requests.

Contextual Understanding

They may struggle to retain information from previous interactions and might not be able to recall past conversation history effectively

Technical Issues

Network issues, background noise, or poor audio quality can affect the performance of speech recognition and conversational IVR system.

FUTURE SCOPE

Complex Queries/Requests

conversational IVR systems would handle complex queries and requests through advancements in AI technologies such as natural language processing (NLP), machine learning, and deep learning.

Adaptive Learning

Conversational IVR systems will continuously learn and adapt based on user interactions and feedback, enabling them to improve their performance over time.

Increased Personalization

Future conversational IVR systems will leverage customer data and machine learning to provide more personalized experiences, such as remembering past interactions and preferences, and offering customized solutions based on individual needs.

CONCLUSION

In conclusion, NEXUS, our conversational IVR system tailored for the banking sector, represents a significant advancement in customer service and engagement. Designed to handle simple customer queries with ease and efficiency with reduced support from human agents.

PAPER PUBLICATION STATUS

The paper has been accepted for publication in **IJRPR**, which will be published in current issue. The Unique ID of paper is **IJRPR-80980**.

REFERENCES

[1] Zhang, J., Chen, S., Liu, J. and He, J., 2023. Composing parameter-efficient modules with arithmetic operations. arXiv preprint arXiv:2306.14870.
Vancouver

[2] Pu, G., Jain, A., Yin, J. and Kaplan, R., 2023. Empirical Analysis of the Strengths and Weaknesses of PEFT Techniques for LLMs. arXiv preprint arXiv:2304.14999.

[3] Pavlyshenko, B.M., 2023. Financial News Analytics Using Fine-Tuned Llama 2 GPT Model. arXiv preprint arXiv:2308.13032.

[4]Considine, B., Albion, N. and Si, X., 2023. Idiolect: A Reconfigurable Voice Coding Assistant. arXiv preprint arXiv:2305.03089.

REFERENCES

[5] Anekar, D.R., Suryavanshi, S., Auti, D., Lokhande, P. and Deshmukh, A., Farmer’s Assistant using AI Voice Bot.

[6] Goyal, A., Singh, A. and Garera, N., 2022. End-to-End Speech to Intent Prediction to improve E-commerce Customer Support Voicebot in Hindi and English. arXiv preprint arXiv:2211.07710.

[7] Luo, R., Sun, L., Xia, Y., Qin, T., Zhang, S., Poon, H. and Liu, T.Y., 2022. BioGPT: generative pre-trained transformer for biomedical text generation and mining. Briefings in Bioinformatics, 23(6), p.bbac409.

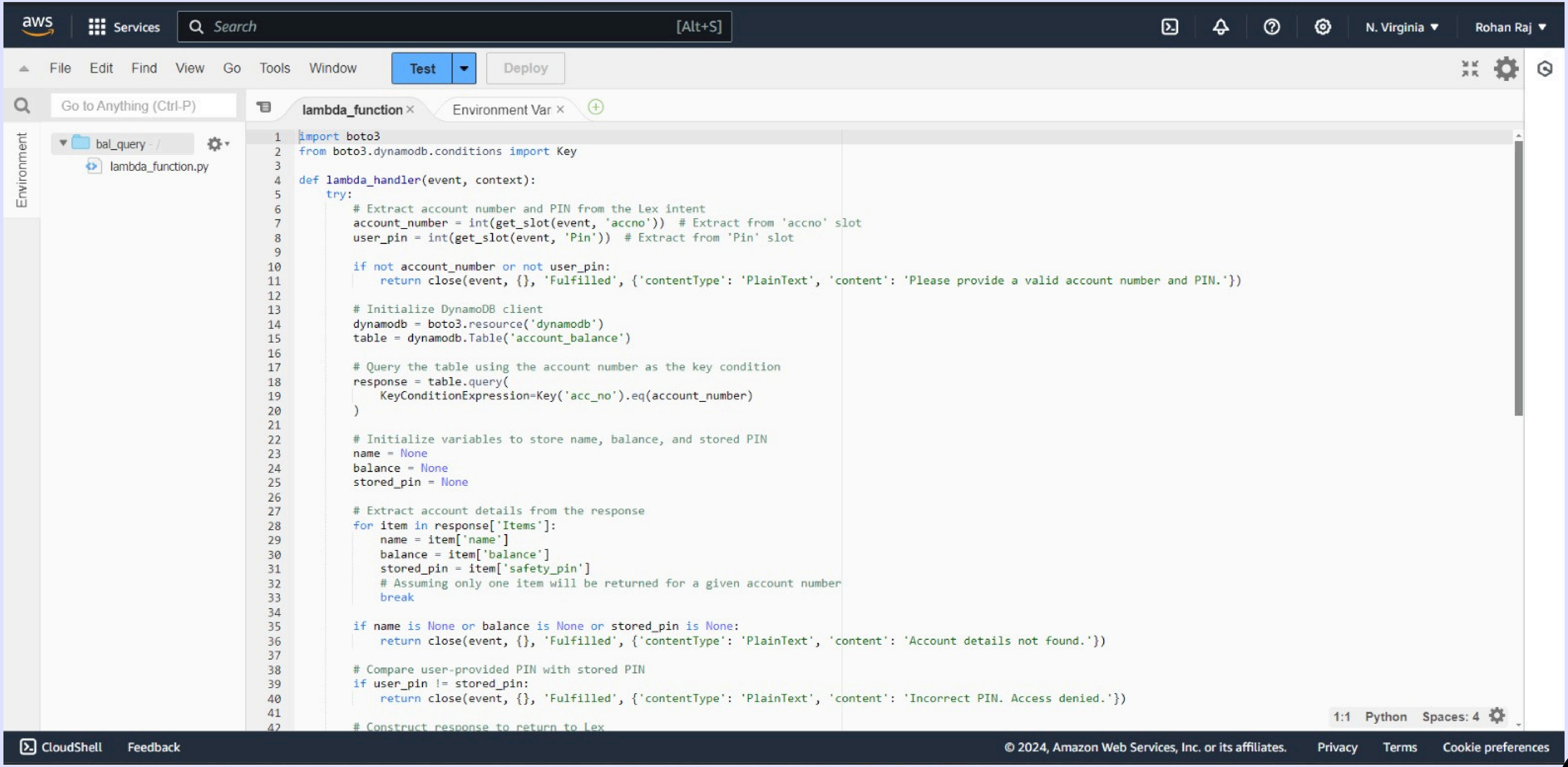
[8] Taghandiki, K., 2023. Building an effective email spam classification model with spacy. arXiv preprint arXiv:2303.08792.

REFERENCES

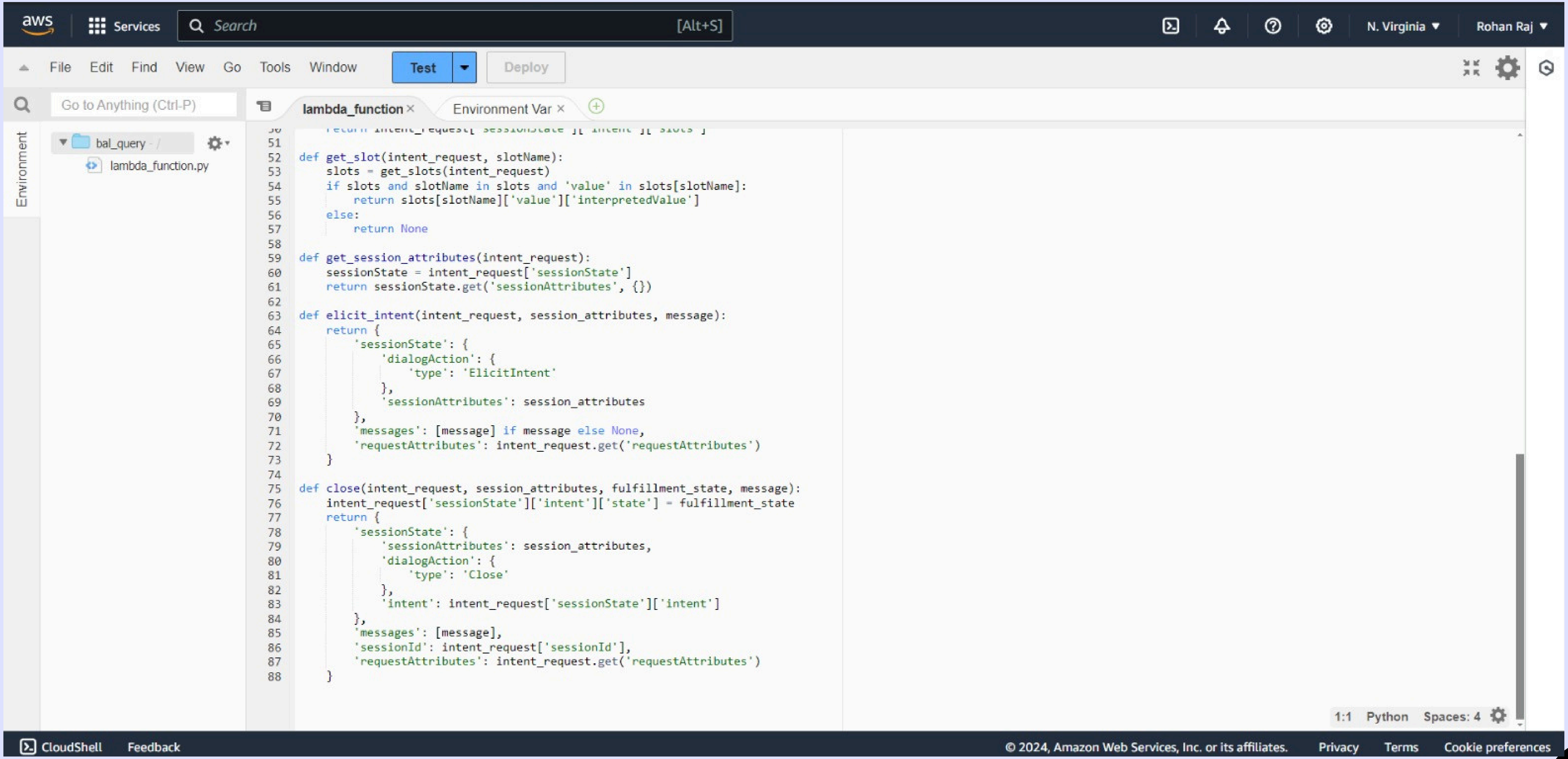
[9] Teye, M.T., Missah, Y.M., Ahene, E. and Frimpong, T., 2022. Evaluation of conversational agents: understanding culture, context and environment in emotion detection. IEEE Access, 10, pp.24976-24984.

[10] Arevalillo-Herráez, M., Arnau-González, P. and Ramzan, N., 2022. On adapting the DIET architecture and the Rasa conversational toolkit for the sentiment analysis task. IEEE Access, 10, pp.107477-107487.

THANK YOU



```python
import boto3
from boto3.dynamodb.conditions import Key

def lambda_handler(event, context):
    try:
        # Extract account number and PIN from the Lex intent
        account_number = int(get_slot(event, 'accno'))  # Extract from 'accno' slot
        user_pin = int(get_slot(event, 'Pin'))  # Extract from 'Pin' slot

        if not account_number or not user_pin:
            return close(event, {}, 'Fulfilled', {'contentType': 'PlainText', 'content': 'Please provide a valid account number and PIN.'})

        # Initialize DynamoDB client
        dynamodb = boto3.resource('dynamodb')
        table = dynamodb.Table('account_balance')

        # Query the table using the account number as the key condition
        response = table.query(
            KeyConditionExpression=Key('acc_no').eq(account_number)
        )

        # Initialize variables to store name, balance, and stored PIN
        name = None
        balance = None
        stored_pin = None

        # Extract account details from the response
        for item in response['Items']:
            name = item['name']
            balance = item['balance']
            stored_pin = item['safety_pin']
            # Assuming only one item will be returned for a given account number
            break

        if name is None or balance is None or stored_pin is None:
            return close(event, {}, 'Fulfilled', {'contentType': 'PlainText', 'content': 'Account details not found.'})

        # Compare user-provided PIN with stored PIN
        if user_pin != stored_pin:
            return close(event, {}, 'Fulfilled', {'contentType': 'PlainText', 'content': 'Incorrect PIN. Access denied.'})

        # Construct response to return to Lex
```



```python
    return intent_request['sessionState']['intent']['slots']

def get_slot(intent_request, slotName):
    slots = get_slots(intent_request)
    if slots and slotName in slots and 'value' in slots[slotName]:
        return slots[slotName]['value']['interpretedValue']
    else:
        return None

def get_session_attributes(intent_request):
    sessionState = intent_request['sessionState']
    return sessionState.get('sessionAttributes', {})

def elicit_intent(intent_request, session_attributes, message):
    return {
        'sessionState': {
            'dialogAction': {
                'type': 'ElicitIntent'
            },
            'sessionAttributes': session_attributes
        },
        'messages': [message] if message else None,
        'requestAttributes': intent_request.get('requestAttributes')
    }

def close(intent_request, session_attributes, fulfillment_state, message):
    intent_request['sessionState']['intent']['state'] = fulfillment_state
    return {
        'sessionState': {
            'sessionAttributes': session_attributes,
            'dialogAction': {
                'type': 'Close'
            },
            'intent': intent_request['sessionState']['intent']
        },
        'messages': [message],
        'sessionId': intent_request['sessionId'],
        'requestAttributes': intent_request.get('requestAttributes')
    }
```