LicenseBot : Empowering International Students with Driver’s License Assistance

# Abhishek Goud Thamatam, Esther Priyadarshini

Natural Language Processing

***Abstract-* Today, international students may experience challenges while obtaining a license to use vehicles in a new country of settlement. This is a spot where LicenseBot finds its place. It functions as a caring friend who will assist international students in Maryland in obtaining their driver’s licenses. The LicenseBot automatic solution is fueled by modern technologies like Dialogflow CX and Vertex AI, providing a personalized approach each time. LicenseBot was invented to simplify things, provide timely information, answer student's questions on the go, and consequently, provide students with solutions to issues they encounter. The paper will dwell on how LicenseBot was designed, how it works, and what makes it essential to highlight its potential to ease access to the necessary services and contribute to establishing a more inclusive academic environment for international students.**

***Keywords-* Chatbot, Dialogflow CX, Vertex AI, Google Cloud Platform, Natural Language Processing (NLP), Data Storage, Agent Builder, Cloud Computing, Machine Learning (ML), Large Language Models(LLMs), Intents**

# I. Introduction

The moment international students arrive in a destination country for the first time, they face the challenge of a new legal domain, in particular, the legal procedures and requirements for obtaining the ownership rights of their driving license. Another factor that contributes to this is the vast number of results that can be found on different sites, which makes it a puzzling situation for the students. Having some explanation about the procedures for international students, as well as overlooking relevant MVA websites, may be necessary as some information for international students is not clear.

To the task set before us, our project will design a tailored and streamlined solution that will find a good application in the summary of international students who intend to get a Maryland driver’s license. As a matter of fact, we have stemmed a chatbot that is meant to provide correct

and accessible information, which is being fed from verified sources of information that are reputable and from legal university websites.

This project was inspired by the observation of the challenges some international students go through in obtaining a driver’s license, as the procedures there are not easy. Online information brings along the issue of junk being available that a student needs to sift through to find precise guidance that applies in this situation.

The major concern that our project tackles is the absence of central and complete data regarding the process of getting a Maryland driver’s license for many foreign students in the country. The current materials, for example, the MVA site, usually have already given some general precautions on short-cutting routine things, but they don’t take into account international students’ particular problems and conditions at all. Omitting personalized data in processes can lead to confusion, aggravation and significant delays in receiving a license.

Our project, which will be given a life by the enthusiasm to make the whole process of student migration easy for international students, is aimed at creating a bridge between information sources that have a scattered layout and information sought specifically by this group. Through a search-able user interface and a verified information link, we aim to help international students with their country license stand-alone problems.

The main purpose of our project is to create a chatbot that becomes a reliable and easy way for international students to access information about acquiring a Maryland driver’s license. Specifically, our goals include:

* Obtain credible facts from trustworthy sources, including university legal websites, to avoid inaccuracy and ensure validity.
* Reducing the complexity through which people can access the right information by adopting a user-oriented interface and providing individual answers to the user’s questions.
* Supporting international students in their acquisition of the necessary knowledge and skills to efficiently go through the driver’s license application process, with the aim of avoiding unnecessary confusion.

## Description of the Chatbot Solution

Our chatbot solution is based on the use of modern technologies like Dialogflow CX, Vertex AI, and Google Cloud Platform to develop a user-friendly and effective service that provides information related to the Maryland DMV license application procedure. By means of data integration from different sources and employing natural language processing skills, the chatbot has the capacity to provide step-by-step guides and address the individual’s concerns.

In a nutshell, our endeavor addresses the problem of international students' having no access to the Maryland driver’s license. By designing a student-oriented chatbot solution, we look forward to equipping students with the re- required knowledge and resources to successfully cope with this critical life aspect in a new country.

# II. Literature Review

Chatbots are highly complex tools that are able to accommodate the key points of natural language communication as well as the perks of AI. From the first days of chatbots in the Sixties, the communicational approach has changed greatly from simple text-based interactions to advanced voice control systems such as Siri and Alexa, thus profoundly changing the way we humans meet with technology [2].

Initially, the bots started with the basic types like help, tech support, customer care, etc., and then, as things started to develop, the chatbots started to cater to different needs and choices. In this manner, rule-based chatbots perform specific tasks, and further natural language processing practiced by machine learning-based chatbots helps in understanding the queries posed by the users. Word-based and voice-based chatbots allow users to choose their communication mode [2]. Compared to that, retrieval-based and generative-based differ in the way they produce replies.



Figure 1: A figure depicting the emergence of chatbots. NLP technology, which has become a major break-

through in the creation of chatbots has greatly contributed to the matter. Basic NLP tools like part-of-speech tagging, tokenization, stemming, lemmatization and Named-entity recognition form the powerful arsenal of chatbots to understand and process human language better. On the other hand, there are now solid chatbot development platforms like Dialogflow, Rasa, and Wit.ai that have made it quite easy to create and implement applications of chatbots of various types.

Lately, the prospect of implementing NLP and chatbots has been figuring out a way to improve the chat capabilities of these systems. Approaches like reinforcement learning and supervised fine-tuning have been adopted to improve and increase the quality of the chatbots’ responses. Projects that fall under ChatGPT, for instance, are from the line of the GPT series model and have a better capacity to respond to a user’s enquiries with a greater level of nuance and relevance to the context compared to the other chatbots.

The literature shows that chatbots can be applied in many fields of business and have the ability to make tasks seem easier and effortlessly straightforward through natural language. With the use of the NLP algorithms, the way forward is to improve the chatbots and employ a solid development platform as a base. It is possible to develop more intelligent and simple chat apps to meet the individual requirements of people across the domains.

# III. System Architecture

Our chatbot program architectural foundation has been employed to offer a strong and efficient platform through which international students who are interested in using the Maryland driving permit can ask for assistance. The system has been arranged as a whole and is composed of modules and components with close results to give users the desired experience.

## A. Overview of the Chatbot Architecture

The foundation of our chatbot architecture consists of Di- Dialogflow CX, Vertex AI, and Google Cloud Platform, all of which support natural language understanding, concurrent conversation flow, and data storage. These techs constitute what a digital assistant solution is made of, making it convenient for students to use it and enjoy personalized help.

## B. Components and Modules of the Chatbot System

1. Dialogflow CX: This part corresponds to the direct language understanding component of our chatbot system. It parses the users‘ query, finds the purpose, and recognizes the entities in order to make sure the user’s real intentions are understood properly [4].
2. Vertex AI: Vertex AI supplies the system with machine learning abilities that manage to improve performance and accentuate the intelligence of our chatbot [5]. Being able to do that, we can configure and use machine learning mods to get chatbots much better at understanding and responding to user messages.
3. Google Cloud Platform (GCP) ’s Bucket: GCP’s Bucket provides secure storage that enables us to effortlessly utilize the necessary data for the purpose of the chatbot. With it, we can store data securely and retrieve data of many different types, such as user-filtered history, program logs, and so on [6].
4. Agent Builder: Agent Builder from GCP acts as a platform, thanks to which we can process the creation and build-up of the dialogues with little effort. The tool provides a user-friendly interface under which we can define intents, entities, and conversational flows, which allows us to tailor the chatbot’s behavior to particular

aspects of the international student’s needs [7].

## B. Descriptions

* **Dialogflow CX** DialogFlow (CX) is an outstanding conversational AI platform that we use to build high-intelligence chatbots capable of coping with complicated conversational patterns. As it has, it offers strong capabilities for dialogue management, context handling, and interoperability with other systems [4].
* **Vertex AI** The Unified platform of Vertex AI is intended for building, training, and placing into operation ML models, and that’s all. It involves pre-built machine learning pipelines and automated model tuning functions, which measurably ease the workflow of creating a machine learning-based chatbot [5].
* **GCP’s Bucket** The Storage in Cloud Storage Bucket, provided by Google Cloud Platform, is a cloud-based scalable storage solution offering durability to our chatbot system with its immutable data storage. It provides us with a secure way to store and get back any data we need, and it facilitates the operation of our chatbot [6].
* **Agent Builder** Agent Builder is a graphical technology provided by GCP that assists in the manufacturing and composing of conversational agents. It provides an easy interface for the creation of intents, entities, and conversation flows so that we can execute them with fewer delays and flexibly tailor the chatbot’s behavior to suit the demands of international students [7].
* **Google Cloud Platform** Google Cloud Platform (GCP) is a set of cloud services that facilitate the development of applications and solutions at the user level, which is empowered by Google. It provides extensive infrastructure and platform services, including computing, storage and machine learning[6]. We use them mainly to build and deploy our chatbot system with scalability, reliability and security as our second priority.

In conclusion, our chatbot system design combines Dialogflow CX, Vertex AI, Virtual Storage Bucket, Agent Builder, and Google Cloud Platform to culminate in the creation of a wonderful and user-friendly tool that international students need to facilitate the process of obtaining a Mary- land driver’s license. These tech components cooperate perfectly to give the required individual advice and help one stay on the right track and cope with the process of licensing things in general.

# IV. Design

In developing and executing the chatbot project for foreign nationals to make them comfortable with a series of procedures carried out to get a Maryland driver’s license, some points should be taken into account to guarantee the efficiency and convenience of the instrument.

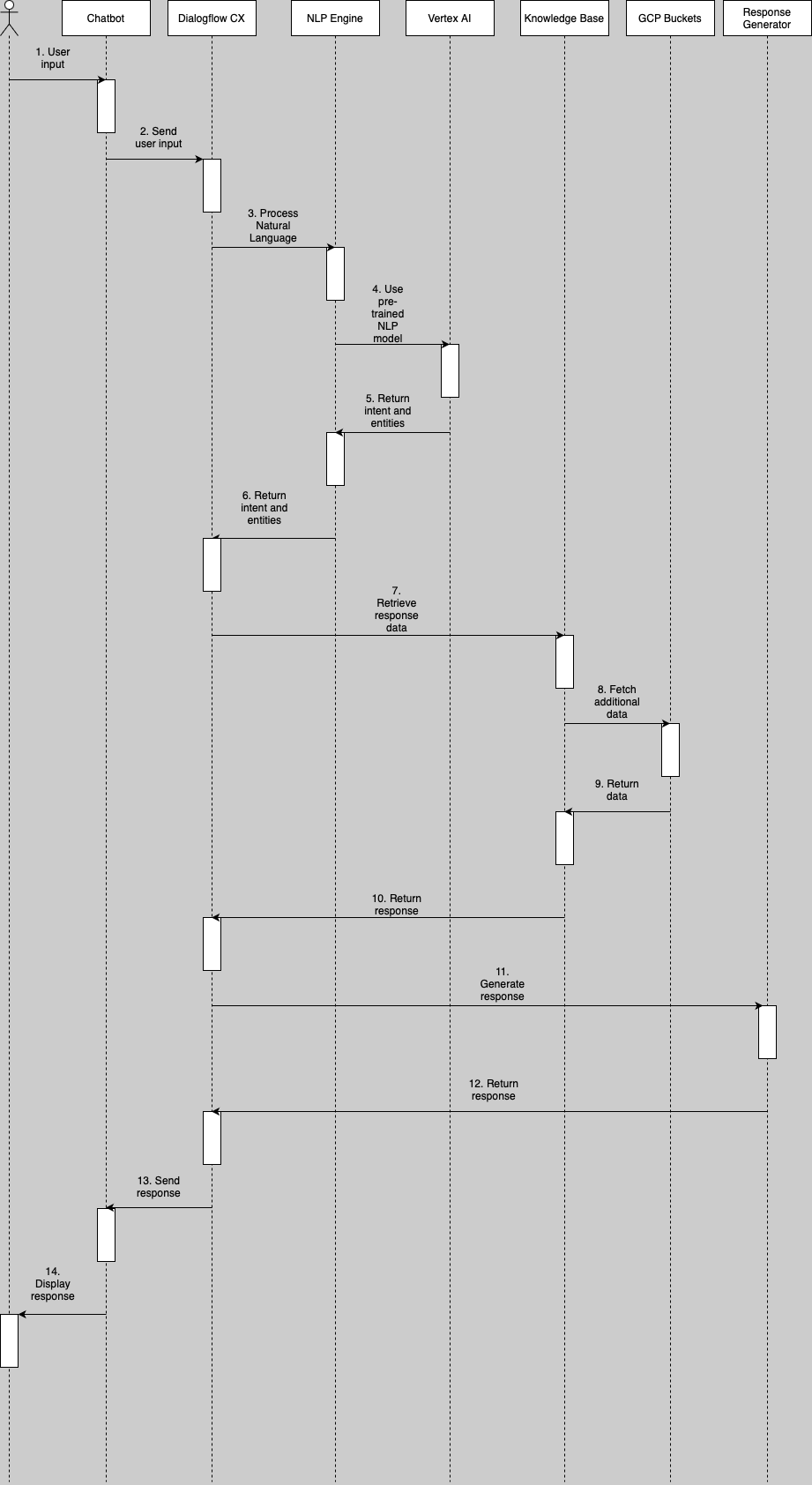


Figure 2: A sequence diagram depicting the interaction between different entities of the chatbot.

## A. Design Considerations for the Chatbot Solution

* **User-Friendly Interface** Chatbot design interface was made not complex and easy to use, targeting foreign students who may be unaware of technological processes.
* **Personalized Assistance** Chatbots have the ability to be the right assistance for all users as they understand what queries they’ve submitted and guide them through all the licensing situations that they are in.
* **Integration of Multiple Technologies** We used Di- Dialogflow CX, Vertex AI, and Agent Builder together so that the system could be run by NLP Dialogflow, machine learning capabilities of vertex AI, and the conversation flow could be managed by Agent Builder.

# V. Data

We extracted our data from several known sources to

ensure that the information provided by the model is precise and sufficient. For data collection, I used the Maryland Motor Vehicle Administration (MVA) handbook and the MVA website, which was rich in rules and layout for attaining a driver’s license in Maryland [10]. Besides this, the data from the international students’ services websites in Maryland was also popularized on the main website to address specific queries and concerns regarding driver’s license regulations. Diverse sources of information, if used appropriately, will help our chatbot deliver the most complete, up-to-date and tailor-made information to all those foreign students who would like to get a driver’s license in Maryland [11].

# VI. Implementation

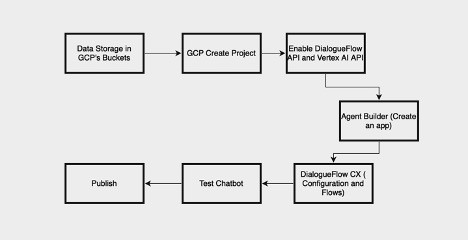


Figure 3: Flowchart depicting the steps involved in creating the chatbot.

## A. GCP Data Storage

The first step in loading data to a Google Cloud Storage (GCS) bucket is signing in to the Google Cloud Console to use your Google account. Once in, access the Cloud Storage menu from the left navigation bar and click on it. If you haven’t been a bucket owner, press ”Create Bucket”; other- wise, choose your present bucket. What comes next? Inside the container, click ”Upload Files” and specify the files that you want to upload for your local machine. Watch the following to make sure that Google Cloud Console is working fine. Check the file in the bucket and confirm once it is uploaded. The data is put up, safely kept there, and can be retrieved by a computer program or physically. Set up the privileges and permission levels in accordance with your requirements by tracking activity and usage using Cloud Console functions [6]. Verify the feasibility and appropriateness of the proposed business idea with consideration of potential risks and challenges. You’re guaranteed this process will help you to have an organized system where you can access your data for your research whenever you need it and save you plenty of time.

## B. GCP Project

First, go ahead and visit the Google Cloud console and sign in using your Google account credentials to create a new GCP project [7]. Click on the project selector drop-down menu on the top right of the screen and open the window

with the project selection options by simply clicking on it. On this tab, you will click on the ”New Project” button to begin the project establishment path. Start a project using a name, an (optional) project, and an organization ID. Click ’Create’ to continue. Wait for your Google Cloud Platform to be created. Please note these could take a few minutes. Lo and behold, after some time, you’ll be led to the project dashboard, where you’ll be able to use a number of tools and materials for your project. Optionally, you can set up billing for your project by selecting the ’billing’ menu and then moving through the prompts. With your GCP now up and running, you can progress to implementing and managing services like Compute Engine, Cloud Storage, and Big- Query to accomplish your project objectives.

## C. Enabling APIs

To enable Dialogflow and Vertex AI APIs in your Google Cloud Platform (GCP) project, first log in to your GCP account and open the GCP Console. From there, navigate to the ”APIs Services” section and click on the ”Enable APIs and Services” button. In the search bar, type ”Dialogflow API” and select it from the list, then click ”Enable” to enable the Dialogflow API. Repeat the same process for the Vertex AI API. Next, authentication can be set up by creating a new service account under the ”Credentials” section. Assign appropriate roles to the service account and generate a new key in JSON format. Download the JSON key file and securely store it for authentication. With the APIs enabled and authenticated, you can now use them in your project to build and deploy your chatbot and leverage AI capabilities provided by Ver tex AI.

## D. Agent Builder

To enable the Dialogflow and Vertex AI APIs from the Google Cloud Platform (GCP), manage your Google Cloud Platform (GCP) project; first, log in to your GCP account and open the GCP Console. You must be in the ”APIs Services” section to do so. Click the ”Enable APIs and Services” button to continue. Search for ”Dialogflow API” in the search bar, and click the name provided to enable it. After that, click the ”Enable” button above. Let’s practice the same inside Vertex AI API. Follow that by going to the ” Credentials” menu, where you will need to create a new service account. In this step, let’s give the service account a suitable role and get a key in JSON format. The JSON key file will be downloaded and should be securely stored for authentication. With APIs of your choice enabled and authenticated, you could then have it integrated into your project and de-ploy a chatbot of your own in addition to using AI technologies available from Vertex AI.

## E. Dialogflow CX Setup

Initiating a new application on the Agent Builder happens to be quite an easy system that takes only three development processes. Users log on to the site and then go to the ”Create New App” area, where they can choose what app they want to build, which can be a chatbot, real-time voice assistant or interactive agent. When users decide on the app type they

want to create, they proceed to tailor some settings, such as opting for the Name, description, language settings, and interaction model [4]. This process also includes entity identification, intent recognition, and dialogue flow management, which are used to solve user questions and responses and integrate with separate systems or APIs if required. To begin with, users need to enter data or import the data that plays the necessary role in helping the app to perform effectively. Data include schemas, sample data, and connections to external data sources if needed. With these personalized apps in hand, users can achieve their targeted objectives by leveraging the type, configuration, and data to address their specialized requirements in the Agent Builder.

## F. Testing Chatbot

To ace the chatbot’s Dialogflow CX testing with the help of test cases, use these particular steps. First of all, determine certain situations or user input scenarios for testing – address common queries, edge cases and complex user journeys. Create a test case each for separation scenario, detailing scenario, and cognitively; the chatbot must be designed to generate a specific response. Ensure that the Dialogflow CX agent is turned on and configured to achieve the desired results. Execute these test cases by interacting with an entity like a user would by logging and writing things down, comparing them with the expected results. Slew up any anomalies found and re-engineer or debug as necessary. Do regression testing in the mentioned area and implement changes, upgrades, or additions if needed. Lastly, the test’s outcomes, comprising the problems, solutions, and feedback, will be noted down, which will improve the future arcs of test/development and bring forth more and more enhanced chatbot performance and user experience.

## G. Publish

When we are done testing the chatbot, we move to the publishing process, wherein a configuration option gives us a choice to integrate either authorized or unauthorized public APIs (Application Programming Interface) and select User Interfaces (UI) styles. After having a notify the API option to your website through the addition of the API code, the system generates an HTML code that can be embedded directly into the website in one click. This helps the chatbot code to robotize chatting and ensure a smooth interaction with users.

Integrating our website with the Dialogflow chatbot will be done by getting the code HTML, which will be plugged directly into the HTML code of the website. This integration will be a pathway to track the activation of the chatbot on the website, which in turn will make it possible for the users to interact with it in real time. We might also end up using JavaScript to make the website more ahead-of-the-game and dynamic functionality-wise. We act this out by including the HTML code supplied to Dialogflow to ensure the chatbot’s operation functions flawlessly and gives its users a smooth and straightforward experience. This method lets us flatten the deployment process for the chatbot, ensuring finetune work across different web platforms.

# VI. Functionality and Features

The driver’s license application chatbot we developed for international students has a lot of characteristics and possibilities included to make the whole process easy and accessible to the users who will be getting the necessary help from it.

## A. Overview of the Chatbot Features and Capabilities

* **Information Retrieval** Chatbot has access to such sources as university websites and official documentation from which it can retrieve useful and confirmed information to successfully respond to user queries.
* **Step-by-Step Guidance** Each user is walked through the licensing process to clearly understand the reasons for this, such as scheduling appointments and documentation submissions.
* **Personalized Assistance** The chatbot adjusts its responses and provides guidance to the individual user’s needs, which might include information concerning their visa status, the country they came from, and their knowledge of former driving.

## B. User Interaction Flow and Dialog Management

* **Initiating Conversations** The platform starts a dialogue with learners by asking questions or letting them express their intentions about how they would like to have a Maryland driver’s license.
* **Dialog Management** The chatbot navigates the dialogues adaptively, grabbing information about the purpose of the visit and such when required, giving selective tips and responding to the visitors in a friendly way.

## C. Handling of Natural Language Processing (NLP) for User Queries

* **Understanding User Queries The chatbot was built on a strong base of NLP and AI techniques that help it un**derstand the user queries in context, forming intents and capturing entities to discover what the user is doing and intending to do [8].
* **Contextual Understanding** The chatbot has the ability to both keep track of the conversation context and appropriately associate previous interactions with future ones. This way, it delivers an experience that is seamless and personalized [9].

## D. Demonstration of the Chatbot in Action

* **Interactive Demo The users** may use the chatbot using a user-friendly interface by either texting or giving voice commands so as to familiarize themselves with it and see the action of getting a Maryland driver’s license.
* **Real-Time Assistance** Chatbot shows its ability to provide delighted users with instant answers and help, and it does so when the individuals need the answers to their questions and clarification of their concerns.

# VII. Evaluation and Results

## A. Evaluation Metrics and Methodology

To monitor whether the chatbot we created for foreigners who are seeking a Maryland DMV license is successful, as well as whether it is easy to use, we applied multiple metrics and methods of evaluation [14].

* **User Satisfaction Surveys** We carried out surveys with the purpose of getting opinions from users regarding their overall satisfaction with the chatbot and specifically on feature specifics, such as ease of use, helpfulness, and clarity of the replies.
* **Accuracy of Responses** The precision was evaluated by comparing the chatbot’s planning based on authentic information via official sources in order to support individuals in having correct information throughout the licensing period.
* **Response Time** The chatbot was secure in its rapid response, and the users got prompt assistance and guidance during the entire conversation session.

## B. Analysis of Performance Metrics

* **Accuracy, The speed of the chatbot’s reply,** was one part of the result of our testing, which involved determining the time within which a query can be responded to and the provision of timely help.
* **Response Time** The chatbot maintained a rapid response time, ensuring that users received prompt assistance and guidance throughout their interactions.We evaluated the chatbot’s response time to user queries, aiming to provide timely assistance and minimize delays in communication.

## C. Results

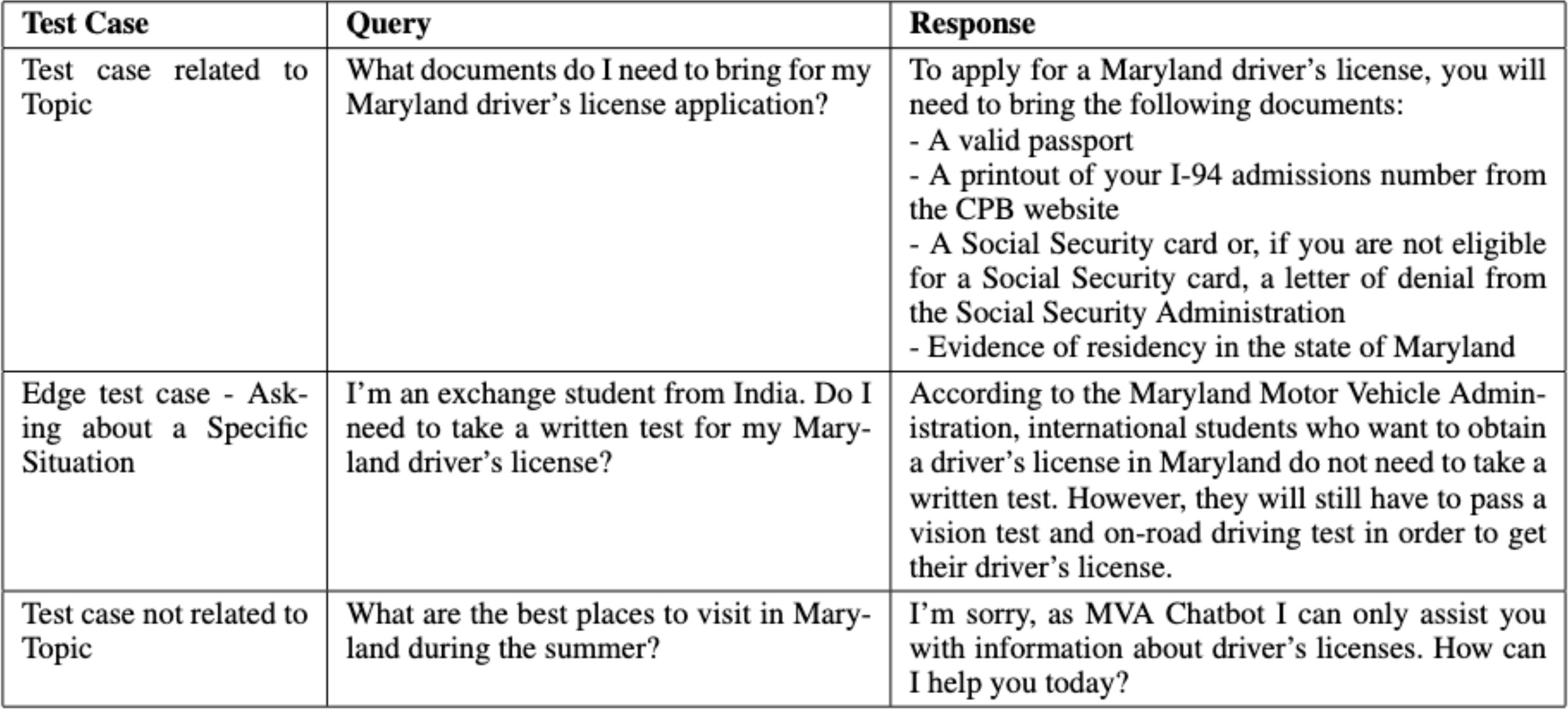


Figure 4: Table showing the generated responses for different test scenarios.

# VIII. Discussion

## A. Interpretation of Results and Findings

The performance of our developed chatbot was confirmed by the results from our test query, thus showing that the bot was able to address correctly and in detail issues faced by

users who wanted to get a Maryland driver’s license. We went through our scenarios selectively and repeated tests from which we were able to provide personalized inspirations and disambiguate complicated ones to users. The fact that our chatbot gets the checks gives credibility to it as a reliable source for international students seeking assistance with licensing.

## B. Comparison with Existing Chatbot Solutions

Unlike other chatbot solutions, ours stands out through its robotic focus on the individual needs of only users. Talking to our chatbot will not be like a too simple chatbot for whom giving just an ordinary answer is enough. It will be powered by more advanced natural language understanding techniques and machine learning models capable of much more meaningful responses. Through this process, the LLM approach is connected to the concept of hands-on LLM, and this way, the user can also have a more charming and effective time with the device.

We bring a strong element to the chatbot development field; we show that personal assistance and contextual guidance can be incorporated, enabling users to have a better experience. Chatbot, in our case, demonstrates the high role of empathy. It does that by being attentive to the unique needs of every international student and tailoring the set responses to those needs. In addition, we believe in incorporating the latest technologies like Dialogflow and VertexAI into our chatbot; this shows the importance of using state-of-the-art tools to improve the chatbot’s ability and user experience level.

## C. Discussion on Strengths and Limitations

**Strengths**

* **Scope Expansion** The additional feature of the chatbot is its ability to cover more than the issue of applying for a driving license. In this regard, not requiring a great deal of refinement, it could give information and guidance with respect to fields like vehicle registration, road rules and driving regulations.
* **Accessibility** The multimedia nature of the chatbot makes it possible for the users to access it in different ways, such as on both web and mobile platforms, to relieve themselves from problems whenever they appear.

**Limitations**

* **Scope Limitation** Although our bot provides fully-fledged assistance in getting a driver’s license in Mary- land state only, its possibilities are, at the current stage, confined to this concern. Increasing the scope of its services to encompass additional topics, such as traffic and due process issues, will make it more practical and user-friendly.
* **Language Limitation** The chatbot covers only English as the current language, which may present a wall for those who wish support in their native tongues. Implementing multilingual assistance is worth trying, as it may overcome that limitation and facilitate a more expanded user base for better accessibility.

# IX. Conclusion

## A. Summary of the Project and its Achievements

All in all, the chatbot successfully deals with the students’ questions and wrongly understood information regarding the licensing process in Maryland. With the help of current natural language processing expertise and machine learning techniques, the chatbot helps the users go through each step of the licensing process and serve their specific needs or circumstances. It becomes available through different platforms, which enable it to respond without delay and in context. This eventually makes it not only efficient but also easy to use.

## B. Contributions to the Field of Chatbot Development

We bring a strong element to the chatbot development field; we show that personal assistance and contextual guidance can be incorporated, enabling users to have a better experience. Chatbot, in our case, demonstrates the high role of empathy. It does that by being attentive to the unique needs of every international student and tailoring the set responses to those needs. In addition, we believe in incorporating the latest technologies like Dialogflow and VertexAI into our chatbot; this shows the importance of using state-of-the-art tools to improve the chatbot’s ability and user experience level.

## C. Future Directions and Potential Areas of Research

Echoing this, some domains of chatbot technology that may make sense to explore in the future are given below. These include:

* **Enhanced Scope** Following this, we intend to develop the chatbot further to cover some of the services and information related to transportation and legal matters like vehicle registration, way of the road and traffic regulations, along with driving safety tips.
* **Multilingual Support** Enabling multilingualism would make possible a wider range of users, comprising foreigners, those who speak a non-English language, and international students who may come from different and varied linguistic backgrounds, to have barrier-free access to the chatbot assistance.
* **Integration with Government Services** Applying the feasibility of integration and incorporating government databases and services might pave the way for the two-way information exchange so that the user can complete appointments and deliver related documentation using the chatbot interface [13]. However, the implementation of web crawling systems with the ability to get data from live or updated websites like government sites may serve the purpose as the users can get the most accurate and up-to-date data whenever they need.

# References

1. Serban, I. V., Lowe, R., Henderson, P., Charlin, L., & Pineau, J. (2018). A survey of available corpora for building data-driven dialogue systems: The jour- nal version. *Dialogue & Discourse*, 9(1), 1–49. https:

//doi.org/10.5087/dad.2018.101

1. Akasaki, S., & Kaji, N. (2017). Chat detection in an intelligent assistant: Combining task-oriented and non- task-oriented spoken dialogue systems. *Proceedings of the 55th Annual Meeting of the Association for Com- putational Linguistics (Volume 1: Long Papers)*. https:

//doi.org/10.18653/v1/p17-1120

1. Caldarini, G., Jaf, S., & McGarry, K. (2022). A liter- ature survey of recent advances in Chatbots. *Informa- tion*, 13(1), 41. https://doi.org/10.3390/info13010041
2. Google. (n.d.). Dialogflow CX documentation. Google Cloud. https://cloud.google.com/dialogflow/cx/docs
3. Google. (n.d.-b). Vertex AI with Gemini 1.5 Pro.

Google Cloud. https://cloud.google.com/vertex-ai

1. Google. (n.d.-a). Cloud storage documentation.

Google Cloud. https://cloud.google.com/storage/docs

1. Google Cloud Platform. (n.d.). Google Cloud Docu- mentation. https://cloud.google.com/docs
2. Nithyanandam, S. D., Kasinathan, S., Radhakrishnan, D., & Jebapandian, J. (2021). NLP for Chatbot Appli- cation. In *Advances in Computational Intelligence and Robotics* (pp. 142–168). https://doi.org/10.4018/978- 1-7998-7728-8.ch008
3. Chauhan, D., Singh, C., Rawat, R., & Dhawan,

M. (2024). Evaluating the performance of con- versational AI Tools. In *Conversational Artificial Intelligence* (pp. 385–409). https://doi.org/10.1002/ 9781394200801.ch24

1. Maryland Department of Transportation Mo- tor Vehicle Administration. (n.d.). https:

//mva.maryland.gov/drivers/Pages/international- applicants.aspx#internationalstudents

1. Maryland Global. (n.d.). Getting a driver’s license. https://marylandglobal.umd.edu/global-learning- all/international-students-scholars/life-umd/getting- drivers-license
2. Cahn, J. (2017). Chatbot - Architecture, Design and Development. [https://www.academia.edu/download/](http://www.academia.edu/download/) 57035006/CHATBOT thesis final
3. Lokman, A. S., & Ameedeen, M. A. (2018). Mod- ern Chatbot Systems: A technical review. *Proceed- ings of the Future Technologies Conference (FTC) 2018*, 1012–1023. https://doi.org/10.1007/978-3-030-

02683-7 75

1. Poser, M., Ku¨stermann, G. C., Tavanapour, N., & Bit- tner, E. A. (2022). Design and evaluation of a conversa- tional agent for facilitating idea generation in organiza- tional innovation processes. *Information Systems Fron- tiers*, 24(3), 771–796. https://doi.org/10.1007/s10796-

022-10265-6

1. Følstad, A., Araujo, T., Law, E. L.-C., Brandtzaeg, P.

B., Papadopoulos, S., Reis, L., Baez, M., Laban, G., McAllister, P., Ischen, C., Wald, R., Catania, F., Meyer von Wolff, R., Hobert, S., & Luger, E. (2021). Future Directions for Chatbot Research: An interdisciplinary research agenda. *Computing*, 103(12), 2915–2942. https://doi.org/10.1007/s00607-021-01016-7

1. Janssen, A., Rodr´ıguez Cardona, D., Passlick, J., & Breitner, M. H. (2022). How to make Chatbots produc- tive – a user-oriented implementation framework. *In- ternational Journal of Human-Computer Studies*, 168, 102921. https://doi.org/10.1016/j.ijhcs.2022.102921