E-Car Renting with chat bot using Artificial Intelligence

A. Salman Khan[1], G. Satheeshkumar[2], Dr. T. Guhan[3]

[1,2] 4th year, department of Computer Science Engineering, Sri Ramakrishna Engineering College

[3] Asst. Professor, department of Computer Science Engineering, Sri Ramakrishna Engineering College
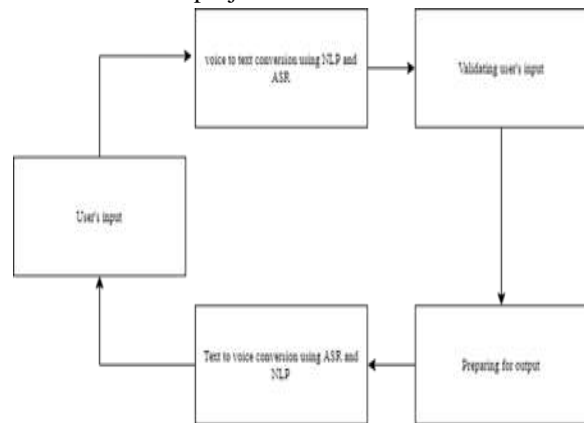
Abstract- **This paper mainly describes a chatbot that can be used to help the car rental company for enabling the renting of cars. It has a user-friendly interface that helps the user to communicate with our bot and rent them for a specific time. It will ask the customers about some important information like the starting and ending date of their journey and also about the car model based upon the number of travelers. The model of rental cars will be provided while chatting with our bot. Based on their requirement, they shall be able to make bookings. The bot shall also check for the availability of the car. If there is no availability of cars, our intelligent bot will automatically suggest another car model. It can segregate the information about the number of bookings in the current month or the last 6 months or the last year. It also provides another feature that the admin can manually chat with their customers in case of any technical difficulties.**

Index terms- **Amazon Lex, Amazon Resource Name, Amazon Web Services, Artificial Intelligence, Lambda function**

I.INTRODUCTION

Artificial intelligence chatbot is a technology that provides interaction between humans and machines using natural language. In General, the main task of a chatbot is like a normal search engine. But it will only produce one output. If the user affords any input, it will make a new search for that specified input. Nothing is related to the previous output. This project mainly focuses on enabling chatbots to act as a tool or guide for car renting. We have established a relationship for previous input. In the chatbot context, this functionality will increase the capability of chatbot's input processing. This makes its implementation likable to a lot of businesses that may not have the manpower or financial resources to keep employees working around the clock. The working of chatbot can be classified into two ways. One option is providing a set of guidelines. And another way is using Machine Learning. If the chatbot works by

providing a set of guidelines, then its conversation will be very small. It will respond to a set of requests. There are some possibilities that the bot might misunderstand and it leads to improper communication. But we have implemented our bot with Machine Learning. The chances of malfunctioning are very low in this case. The Basic workflow of our project is shown below:

II. RELATED WORK

A. AN IPHONE'S VOICE ASSISTANT -SIRI

Siri is Apple's built-in voice-controlled personal assistant. It was introduced as an IOS application by Siri, Inc., and later acquired by Apple on April 28, 2010. The idea is that you talk to the assistant as you would do to a friend and it aims to help you get things done, whether that be making a dinner reservation or sending a message. It uses Artificial Intelligence and Natural Language Processing, and it comprises three components - a conversational interface, personal context awareness, and service delegation. The conversational interface is a straight word-for-word voice recognition but deciphering the meaning is all down to statistics and machine learning. This is where the personal context-awareness system comes in. The service delegation system provides access to all the iPhone's built-in

apps and their inner working which helps the assistant to perform the desired task.

B. GOOGLE ASSISTANT FOR ANDROID MOBILE

Google's version of Apple's Siri is Google Assistant. It is one of the most advanced and dynamic of the assistants out there. Rather than providing links to websites, it is designed to have conversations with you to complete tasks. Unlike Siri, Google Assistant can handle a wider range of questions because Google Assistant taps into Google's web-wide search results every time you search, making it more comprehensive.

C. A WEB APPLICATION FOR CAR RENTING PURPOSES

An online car rental system is an application to automate the car rental business. This system is designed specifically for medium and small-scale businesses. For the past couple of years, the low cost and availability of resources made this rental business successful. But recently the demand reached its peak as there was difficulty in finding trained resources and satisfying customer needs. These factors lead to the concept of self-driven cars where customers can rent vehicles by the hour, day, or week. With the rise of internet usage, everyone is finding their needs at the top of search engine page results. So, by taking the self-driven car rental service online, Customers will be able to use this service with ease. Thus, making the business sustain

III. ISSUES IN RELATED WORKS

- Google Assistant and Siri will do some basic commands like "Set an alarm", "Sing a song". If the user provides an irrelevant phrase or command, it will directly pass to the default browser.
- If the user provides single input, it will provide a single output. It will not answer for nested queries. This is considered to be a lack of communication.
- In a normal car renting website, there will be a form that would be present on the very first page. The user has to manually fill that form to rent.

IV. PROPOSED METHODOLOGY

We have proposed a methodology where a Customized Artificial Intelligence chatbot will play a vital role in Car renting purposes. We have trained our chatbot using Amazon Web Services that can be integrated in any kind of Web platform. If the user provides any wrong input or any irrelevant input our chatbot will not proceed to the next question. It will show the user the same question for as many times as required by the company. This function will let the user know that the input provided by him/her is wrong.

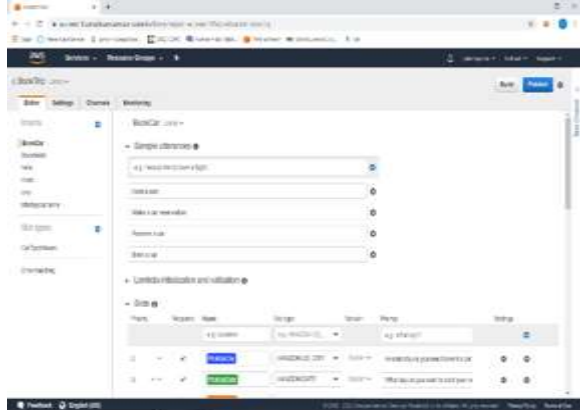## V. MERITS IN OUR PROPOSED WORK

- This chatbot will tend to attend the customers 24x7 to book a car.
- This will also require a smaller number of laborers. A single admin will be more enough to manage the car renting objective.
- Voice chat communication between the user and chatbot will be more effective. So that the user doesn't have to worry about filling a form.

VI. MODULES

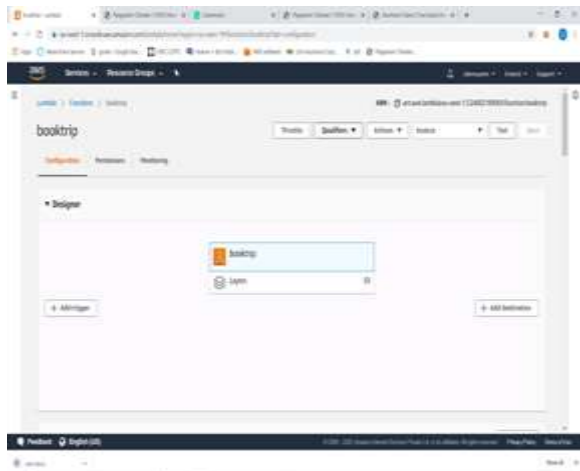### A. TRAINING OUR CHATBOT USING AMAZON LEX

Amazon Lex is one of the services that are provided by Amazon Web Services and it is one of the finest platforms to train and deploy our customized chatbot. The first step to train our bot is Creating Intent. An Intent is something that the user intends to do or find out. After creating Intents, we have to focus on two main things called, "Sample Utterances" and "Response." Amazon Lex also has a feature called Natural Language Understanding. For example, if the user tells, "what's your name" instead of "what is your name", it will be able to understand that the user is asking its name. It will not throw any error. And Intent can be further distinguished by slot name, slot types, and prompt. The slot name is just like a variable name. The slot type is like the type of variable. Prompt is the message that the user is going to get to make sure they send up the information that we want. Using these options, we have trained our

bot to acquire some information such as starting date of journey, Ending date of journey, type of cars, etc.,

B. USING LAMBDA FUNCTION SERVICE FOR VALIDATION

Lambda function is also another service that is provided by Amazon Web Services. This function can be used as a code hook for our trained chatbot. The main function of this lambda is to initialize, validate, and also for configuring our intents. Unless we use any lambda function, it will provide the intent information directly to the customers. Without Lambda our bot's performance will be very low and the main drawback is that it will not validate any slots and their types. We could configure our trained bot with Lambda Function using Amazon Resource Name. Amazon Resource Name is an ID that is unique and is given to each developer who has an AWS account. The programming language that was used here is Python 2.7. We have also implemented another feature called sentimental analysis. This would be very much useful for the company to increase its market rate.
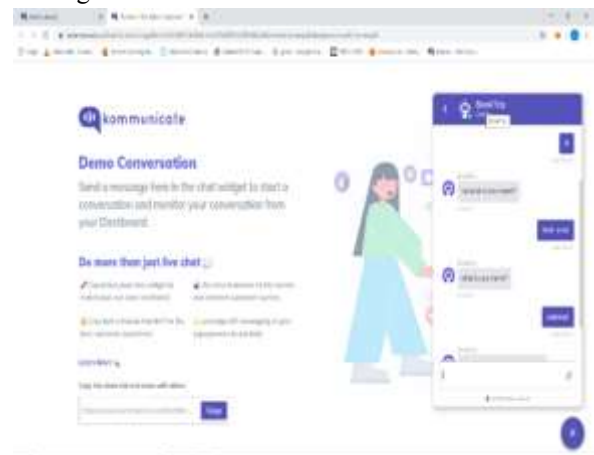

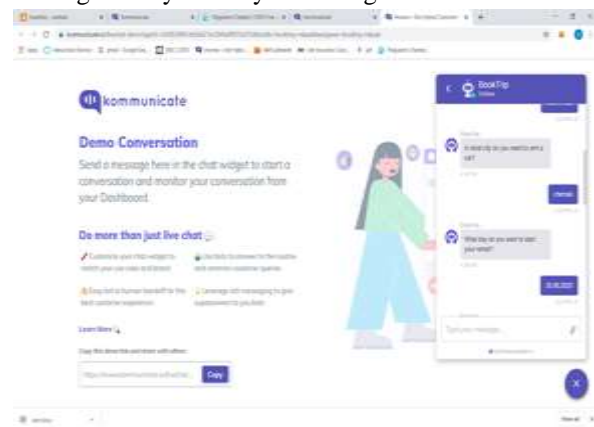## C. TESTING OUR BOT IN A WEB PLATFORM

After configuring our bot with the lambda function, we could directly deploy in any web application. We have used an online platform called Kommunicate.io. We can test one or more at the same time. If we need only need any specific intent that should work, it will provide Resolve function. This is a platform where we could deploy a bot for testing. We just need a Gmail id to work on this platform. For integrating our bot, we need some information like Access key ID, Secret Access Key, Region, Our bot name in Lex platform and Bot alias name. We shall have all the information once we sign up in the AWS management console. Once after integrating our bot, it is ready to use. If the user enters their information for booking, it will be stored as a .json file in the Amazon Lex console. Thus, it would act as a database also.
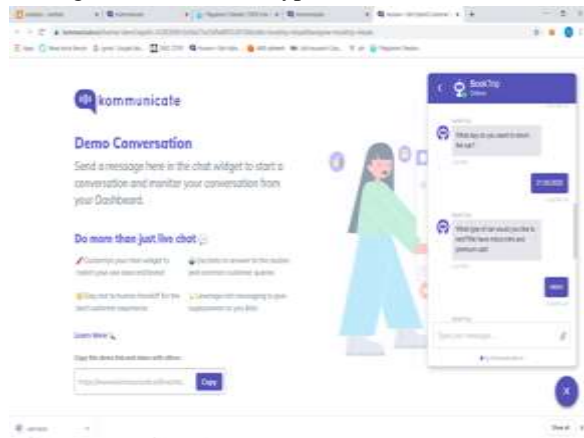
VI. RESULTS

Asking the customer's name:

Asking the city and day of renting:

Asking about return and type of car:

Providing the price of car and conformation:

VII. FUTURE WORK

1 Customizing different bot for different fields.
2 Providing an acknowledgment after booking via sending an email to the user.
3 Notification can be provided if there is any discount for booking.

VIII. CONCLUSION

Being a chatbot, it will attract more users to the company. It will also require very few of the precious time of the users to book a car. This project will also help all those peoples around the globe who require taxi/cab for their use with the help of a bot. Chatbots attend to consumers at all times of the day and week. It's not limited by time or any physical location. This will provide an experience as chatting with an actual human.

REFERENCES

[1] G. Daniel, J. Cabot, L. Deruelle, and M. Derras, ''Multi-platform chatbot modeling and deployment with the jarvis framework,'' in Advanced Information Systems Engineering (Lecture Notes in Computer Science), vol. 11483, P. Giorgini and B. Weber, Eds. Rome, Italy: Springer, Jun. 2019, pp. 177–193, doi: 10.1007/978-3-030-21290-2_12.

[2] O. Diaz and F. M. Villoria, ''Generating blogs out of product catalogues: An MDE approach,'' J. Syst. Softw., vol. 83, no. 10, pp. 1970–1982, Oct. 2010, doi: 10.1016/j.jss.2010.05.075.

[3] D. Falessi, N. Juristo, C. Wohlin, B. Turhan, J. Münch, A. Jedlitschka, and M. Oivo, ''Empirical software engineering experts on the use of students and professionals in experiments,'' Empir Softw. Eng, vol. 23, no. 1, pp. 452–489, Feb. 2018, doi: 10.1007/s10664-017-9523-3.

[4] L. C. Klopfenstein, S. Delpriori, S. Malatini, and A. Bogliolo, ''The rise of bots: A survey of conversational interfaces, patterns, and paradigms,'' in Proc. Conf. Designing Interact. Syst. (DIS), 2017, pp. 555–565.

[5] https://link.springer.com/chapter/10.1007/978-3-319- 91662-0_9