PROJECT REPORT

Intelligent Customer Help Desk with Smart Document Understanding

Category: Artificial Intelligence Developer Application
ID:SPS_APL_20200000571
Project ID: SPS_PRO_99
Internship at SmartInternz

Rayman Kaur
rayman1128.cse18@chitkara.edu.in

1. INTRODUCTION

1.1 My Working

I started my internship on 2nd May 2020. Github is the largest open source tool. The set-up and development environment has been created on github and the user name is Rayman174-A. Slack is an online tool used for formal chats and it makes the conversation simplified. All the messages can be searched which makes makes it easy to use. The account on slack has been made and the platform is very convinient. The zoho writer is like a word document but has many more features added to it. It saves the content automatically which makes is really feasible to users. I have made my account on IBM cloud.
The node-red-starter application has been developed partially till now.

What is node-red?

It is basically a programming tool for wiring together hardware devices, API's and online services. It is build on Node.js.

What is an API?

AN API takes threquest from a client, goes to the system and returns the response back to the client. Making a weather chatbot requires external sources which provide API so as to connect it with Watson Assistant. Other tools such as Microsoft azure, aws cloud can also be used for making bots.

Chat-Bots

A chatbot is a service powered by rules and sometimes artificial intelligence, that you interact via chat interference.
It was initiated by Alan Turing in 1950.
Test of machines ability to exhibit intelligent behaviour.

Main use cases of chatbots:-

- Virtual assistant
- E-commerce
- Customer service
- Call centre support
- Help desk

Working

1.When the user types his query in the chat window, the text is fed into NLU.

2.NLU extracts intents and entities.
3.Based on intent, the decision engine calls mapped functions and pass entities to it.
4.Query is processed using knowledge base.
5.Response is sent to client UI.

Types of chatbots

1. **Scripted chatbots:-** Predefined knowledge base and respond to particular questions.
2. **Self-learning chatbots:-** AI powered and more dynamic and intelligent scripted chatbots.

Why do we need ChatBots?

1.Assessible anytime
2.Handling capacity
3.Cost effective
4.Faster onboarding

Process of working with Chat-Bot

The anything-else dialog should be added at last. It's basically like you cannot write else condition before if condition. By using slots, you can prompt questions from chatbot.
Main thing is to connect the API to the bot.
Watson works 60% faster in all ways transforming the world.

IBM Watson Terms:-

Intents:- Goal or purpose of user's question
Entities:- Specific details about the question
Dialog:- Uses intents and entities to provide user's interaction

The chat bot can be integrated with the following:-

- Slack
- facebook messenger
- word press
- your own website

Steps for creating a webhook:-

Download files from mondlanglais/assistant-webhook-demo.
Go to functions-->Actions-->Create-->get flight file-->status-->Create
The file downloaded is get-flight-status.txt-->paste-->save
Endpoints-->Enable as web section-->Save
Copy URL--> Watson Assistant

Options-->Webhooks-->Paste URL and add .json at the end-->Dialog-->Check flight status-->$flightNumber-->Customize-->ON Webhook
The Assistant responds with:- $webhook_result_1/message.

Watson Discovery

1.Sequece:- It is a chain of actions, invoked in order, where the output of the one becomes input to the next.
2. Trigger:- A declaration that you want to react to a certain type of event.

1.2 Overview

We use the typical customer care chatbot experience but instead of relying on predefined responses, our dialog will provide a hook that can call out to other IBM Watson services for additional sources of information. In our case, it will be an owner's manual that has been uploaded into Watson Discovery.

1.3 Purpose

The purpose is to Enhance the customer helpdesks with Smart Document Understanding using webhooks in Watson Assistant.

2. LITERATURE SURVEY

2.1 Existing Problem

The typical customer care chatbot can answer simple questions, such as store locations and hours, directions, and maybe even making appointments. When a question falls outside of the scope of the pre-determined question set, the option is typically to tell the customer the question isn't valid or offer to speak to a real person.

2.2 Proposed solution

In this project, If the customer question is about the operation of a device, the application shall pass the question onto Watson Discovery Service, which has been pre-loaded with the device's owner's manual. So now, instead of "Would you like to speak to a customer representative?" we can return relevant sections of the owner's manual to help solve our customers' problems.

To take it a step further, the project shall use the Smart Document Understanding feature of Watson Discovery to train it on what text in the owner's manual is important and what is not. This will improve the answers returned from the queries.

3. THEORETICAL ANALYSIS

3.1 System Analysis

Steps required:-
1.Create an IBM cloud account.
2.Configure Watson Discovery
3.Create IBM Cloud Functions action
4.Configure Watson Assistant.
5.Build node-red flow to provide UI to the chatbot.
6.Deploy and run the application

4. EXPERIMENTAL INVESTIGATIONS

1. Create IBM Cloud Services

To create IBM Cloud, visit https://www.ibm.com/cloud
Sign Up/Login to the IBM cloud.
After logging in, you can see the IBM cloud dashboard.
Click on create resource.
We need to create the following two services:-

1. Watson Discovery
2. Watson Assistance

To create watson discovery service
Click on watson discovery--> select a region,plan-->Configure your service-->Create
The watson discovery service is created.

To create watson assistance service
Click on watson assistance-->select a region,plan-->Configure your service-->create
The watson assistance service is created.

All of the services will be found in resource list section.

Watson discovery
Launch watson discovery service and create a new data collection. Give it a new name.
Upload the documentecobee3_UserGuide.pdf.
This pdf needs to be processed under **smart document understanding.** Click the configure data button. The main goal is to annotate all the pages in the document so that discovery can learn what text is important and what all can be ignored.
Under the service credentials, you will keep a note of API key and the url.

Cloud Functions

Check for functions tab in catalog-->Start creating-->Actions-->Paste the json code in the code section-->Parameter tab-->Enter url,environment_id,collection_id,iam_apikey.

Now go the code section again and click invoke. Click on the endpoints and enable the web action

Watson Assistant

Launch the watson assistant-->create skill-->Dialog skill-->next.
Use sample skill-->select customer care sample skill-->create an intent and type in the questions.
Add the node below What can I do node.
Name the node as Ask product information and assign it to the new intent which you just created.

The webhookl communicates with an IBM Cloud Functions web action, which is connected to the Watson Discovery service.
Enable the webhook for IBM Cloud functions which you created-->Apply.

The dialog node should have a Return variable [1] set automatically to $webhook_result_1. This is the variable name you can use to access the result from the Discovery service query.
You will also need to pass in the users question via the parameter input [2]. The key needs to be set to the value: "<?input.text?>" .

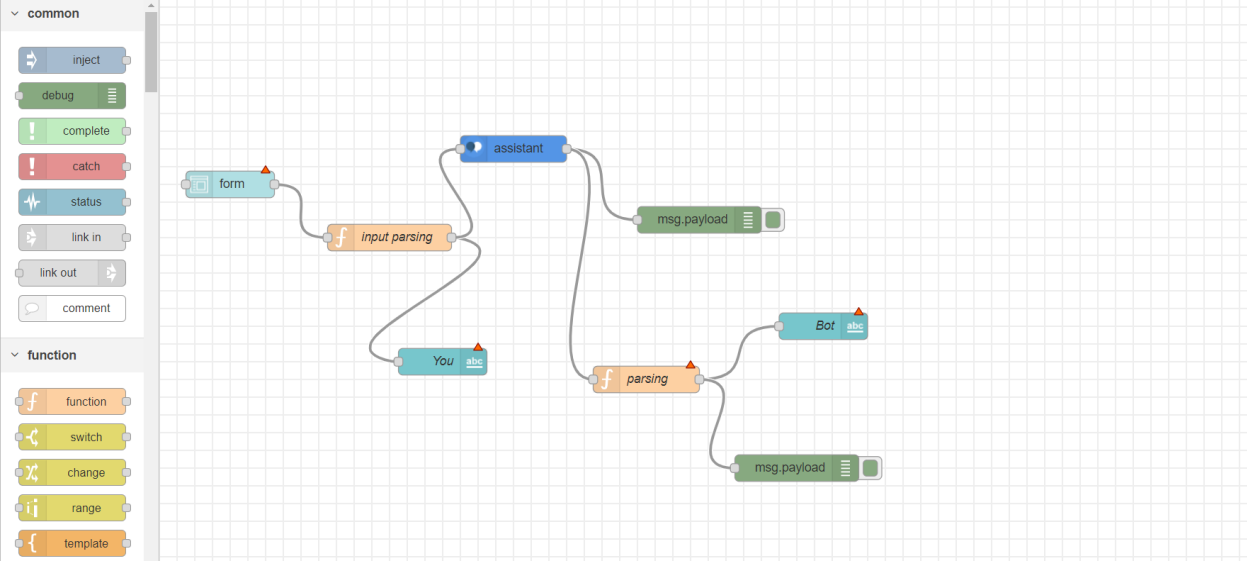
Building a node-RED flow to integrate all services

Click on node-RED app and start creating it. Select the region as Chennai or whatever is closer to your place and create. You will need to create an IBM Cloud API key to allow the deployment process to access your resources.
Now deploy the app.
Now open the node-RED app and reduce the memory per instance to 128Mb.
Click on visit app URL on the top left of the page.
Click on next and at the end , the main application will load.

Configure Nodes and Build a Web Dashboard in Node-RED

To add nodes to integrate assistant, click the dropbox at the top right of page.
Click onManage Palette.
Go to install tab and search for node-red-dashboard.
Now deploy and run the application.

The node-red flow for the smart customer care chatbot is as follows:-



First, Add a Form Node. Connect it with a function node and name it input parsing. To that add a Text Node and name it You. To the input parsing node, add assistant node.Enter the API Key, Service Endpoint (URL) and Workspace ID (Skill ID) from Step 4 and click Done. To the assistant node add debug node. Add another function node to assistant node and name it parsing. Add text node to parsing node and name it Bot. Add another debug node to parsing node.

The chatbot link:-
https://node-red-rcozs.eu-gb.mybluemix.net/ui/#!/0?socketid=lpb4Fr8g-6sCQHY5AAA0

ADVANTAGES & DISADVANTAGES

Advantages: • Faster Customer Service • Increased Customer Satisfaction • Lower Labour Costs • Variety of Uses • Data collection • 24-7 availability • Multiple Customer Handling

Disadvantages: • Limited Responses for Customers • Customers Could Become Frustrated • Maintenance • They aren't human • Time-Consuming

8. APPLICATIONS

A Product or Software Company Customer Help Desk

9. CONCLUSION

An Intelligent Customer Helpdesk with Smart Document Understanding is made using various IBM Services like IBM Watson Discovery, IBM Watson and IBM Cloud Function.

10. FUTURE SCOPE

A More Human Friendly Chatbot, or a personalized Chatbot is to be expected.
All the source code for this project has been provided in the github repository:-
https://github.com/Rayman174-A/Chatbot