Project Report

1. Introduction

1.1 Overview

To ensure that the project meets its completion criteria, following tasks need to be completed:

- A customer care dialog skill in Watson Assistant
- An enhanced Watson Discovery collection with Smart Document Understanding
- An IBM Cloud Functions web action that allows Watson Assistant to post queries to Watson Discovery
- A web application to integrate all the above parts
- Deploying of the web application in IBM cloud

1.2 Purpose

The project aims at providing a further assistance in customer handling by giving an additional functionality to a typical customer care chatbot for smart document understanding. So, in cases of queries regarding operations of a device, the Smart Document Understanding feature of Watson Discovery to return relevant sections of the owner's manual of the device asked about in the query.

2. Literature Survey

2.1 Existing Problem

The current chatbots avaible can answer to a customer’s query with a limited standpoint view. They are able to formally conenct and greet the customers and get details on the problems and requirements of the customer and then connect them to some staff or associate who can answer their queries. But it isn’t well learned to take on queries particular to functionality of some hardware that can directly be resolved with the look at owner’s manual.

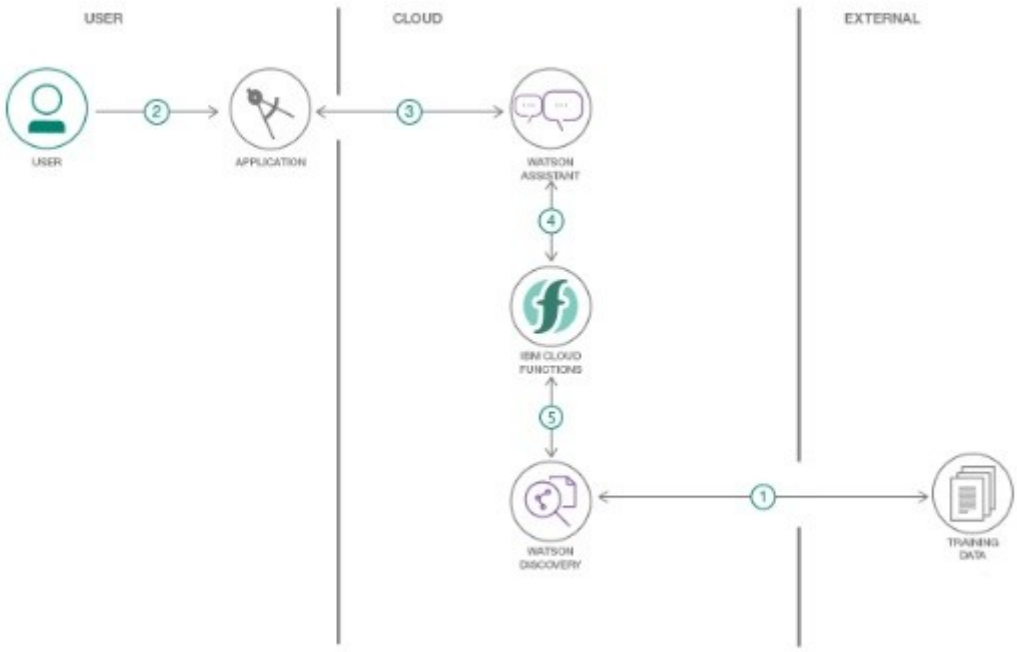
2.2 Proposed Solution

In this project, there will be another option. 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 owners manual. So now, instead of “Would you like to speak to a customer representative?” we can return relevant sections of the owners 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 owners manual is important and what is not. This will improve the answers returned from the queries.

3. Theoritical Analysis

3.1 Block Diagram



3.2 Hardware/Software designing

- Create IBM Cloud Services
- Configure Watson Discovery
- Create IBM Cloud Function Action
- Configure Watson Assistant
- Create Flow and configure node
- Deploy and run node red app

4. Experimental Investigations

Configure Watson Discovery

Launch the Watson Discovery tool and create a new data collection by selecting the Upload your own data option. Give the data collection a unique name. When prompted, select and upload the ecobee3_UserGuide.pdf file located in the data directory of your local repo.

The Ecobee is a popular residential thermostat that has a wifi interface and multiple configuration options.

Before applying SDU to our document, lets do some simple queries on the data so that we can compare it to results found after applying SDU.

Click the Build your own query [1] button.

Enter queries related to the operation of the thermostat and view the results. As you will see, the results are not very useful, and in some cases, not even related to the question.

Annotate with SDU

Now let's apply SDU to our document to see if we can generate some better query responses.

From the Discovery collection panel, click the Configure data button (located in the top right corner) to start the SDU process.
Here is the layout of the Identify fields tab of the SDU annotation panel:

The goal is to annotate all of the pages in the document so Discovery can learn what text is important, and what text can be ignored.

As you go though the annotations one page at a time, Discovery is learning and should start automatically updating the upcoming pages. Once you get to a page that is already correctly annotated, you can stop, or simply click Submit [5] to acknowledge it is correct. The more pages you annotate, the better the model will be trained.

For this specific owner's manual, at a minimum, it is suggested to mark the following:

- The main title page as title
- The table of contents (shown in the first few pages) as table_of_contents
- All headers and sub-headers (typed in light green text) as a subtitle
- All page numbers as footers
- All warranty and licensing infomation (located in the last few pages) as a footer
- All other text should be marked as text.

Once you click the Apply changes to collection button [6], you will be asked to reload the document. Choose the same owner's manual.pdf document as before.

Now, as a result of splitting the document apart, your collection will look very different.

Create IBM Cloud Functions

ow let's create the web action that will make queries against our Discovery collection.

Start the IBM Cloud Functions service by selecting Create Resource from the IBM Cloud dashboard.
In the code editor window [2], cut and paste in the code from the disco-action.js file found in the actions directory of your local repo. The code is pretty straight-forward - it simply connects to the Discovery service, makes a query against the collection, then returns the response.

Click the checkbox for Enable as Web Action [2]. This will generate a public endpoint URL [3].

Take note of the URL value [3], as this will be needed by Watson Assistant in a future step.

To verify you have entered the correct Discovery parameters, execute the provied curl command [4]. If it fails, re-check your parameter values.

Configure Watson Assistant

Launch the Watson Assistant tool and create a new dialog skill. Select the Use sample skill option as your starting point.

This dialog skill contains all of the nodes needed to have a typical call center conversation with a user.

Add new intent

The default customer care dialog does not have a way to deal with any questions involving outside resources, so we will need to add this.

Create a new intent that can detect when the user is asking about operating the Ecobee thermostat.
From the Customer Care Sample Skill panel, select the Intents tab.
Click the Create intent button.

Name the intent #Product_Information, and at a minimum, enter the following example questions to be associated with it.

Create new dialog node

Now we need to add a node to handle our intent. Click on the Dialog [1] tab, then click on the drop down menu for the Small Talk node [2], and select the Add node below [3] option.

Enable webhook from Assistant

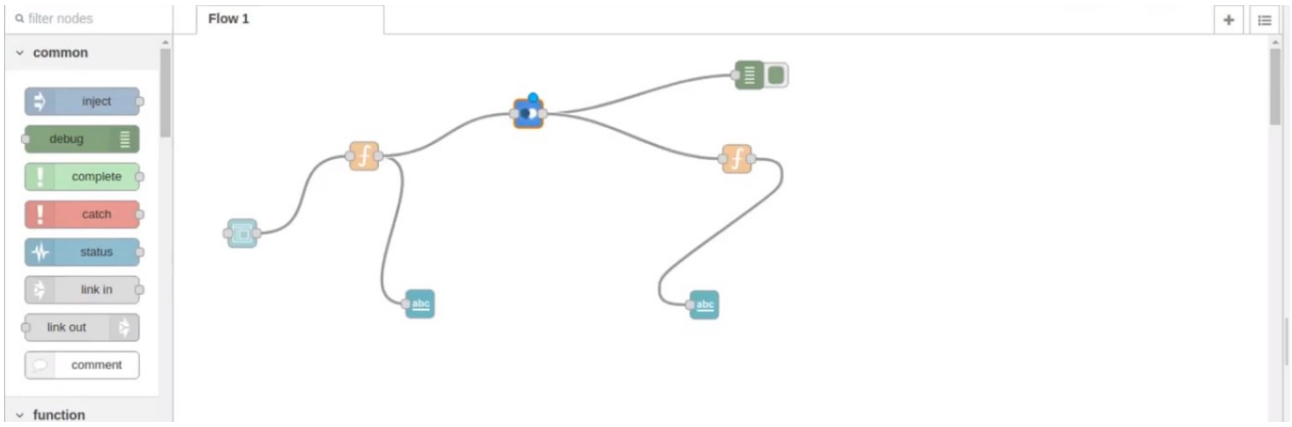
Set up access to our WebHook for the IBM Cloud Functions action you created in Step #4.

Select the Options tab [1]:
Enter the public URL endpoint for your action

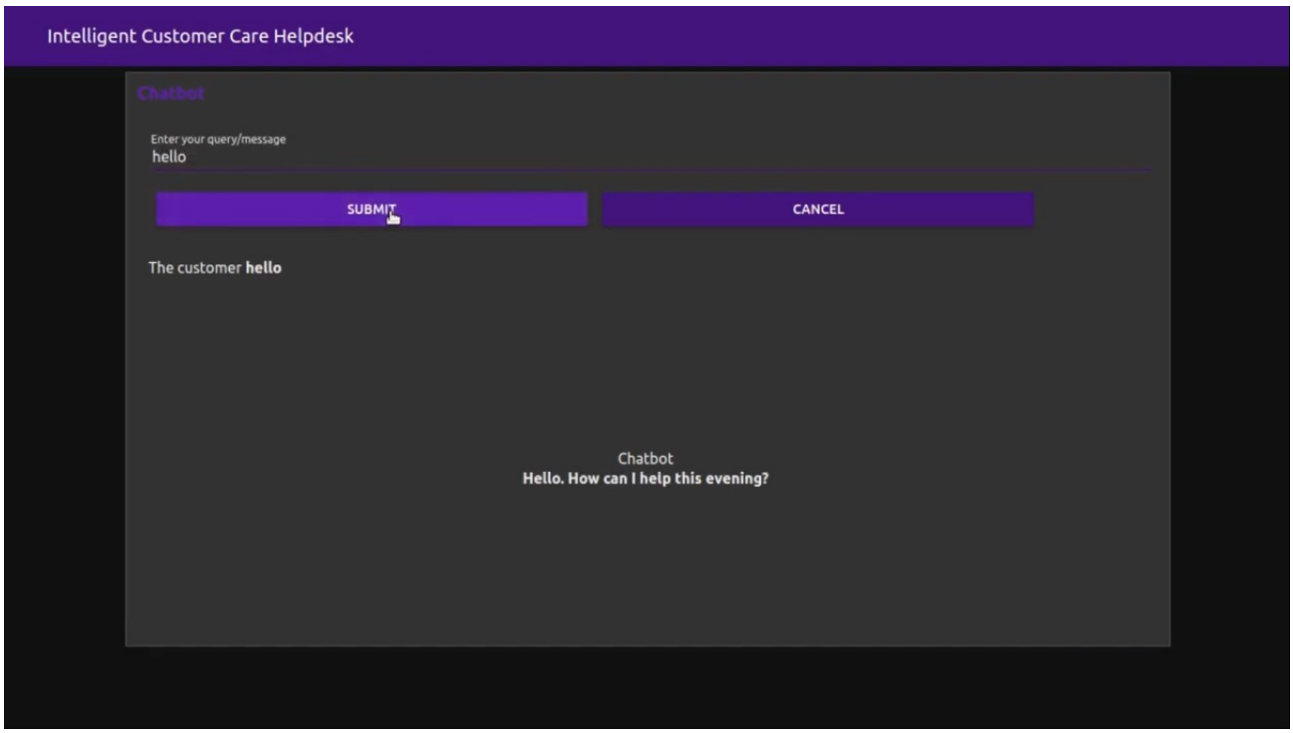
Return to the Dialog tab, and click on the Ask about product node. From the details panel for the node, click on Customize, and enable Webhooks for this node Click Apply.

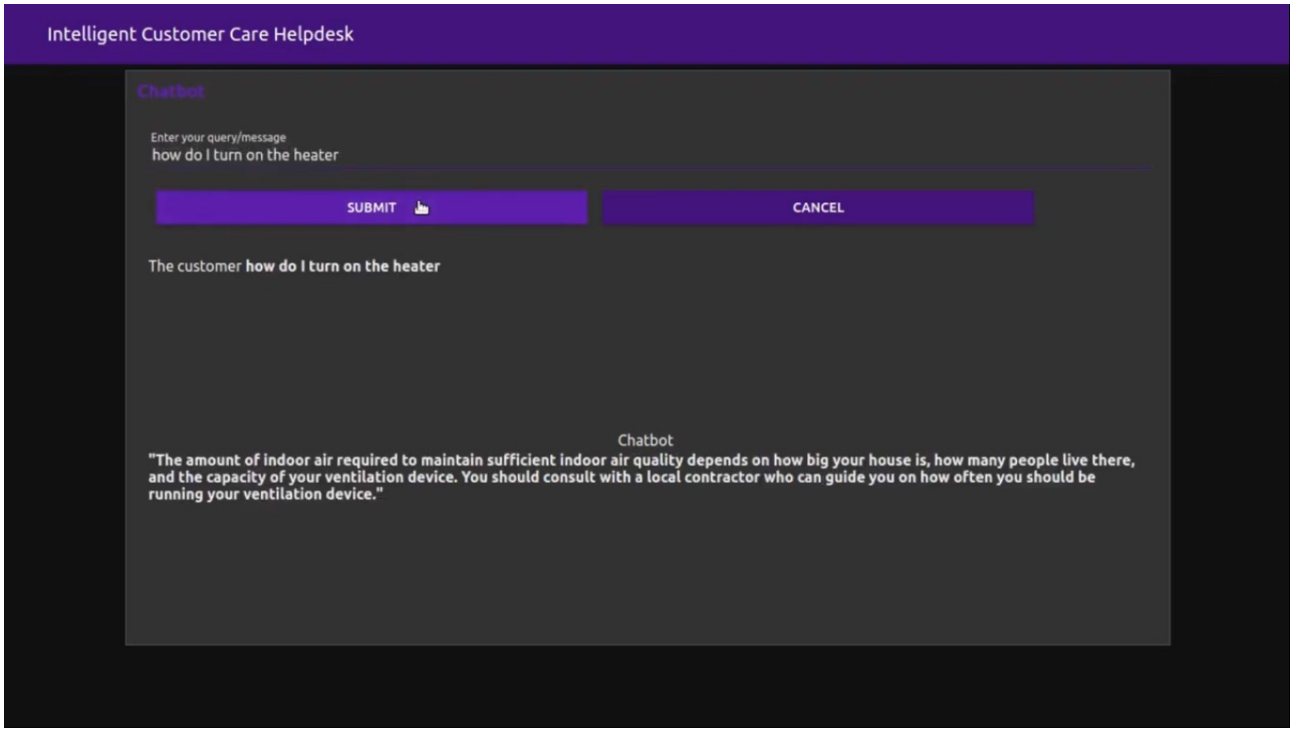
The dialog node should have a Return variable [1] set automatically to $webhook_result_1. This is the variable name you can use to access the result from the Discovery service query.
You will also need to pass in the users question via the parameter input.

5. Flowchart



6. Result





7. Advantages and Disadvantages

Advantages

- Reduced Manual Effort
- Faster Resolution of the customer’s query
- Self-reliant

- Cost Effective

Disadvantages

- Improvement in training of the chatbots needed
- Risk reduction teckniques needed to be implemented
- In case the customer’s query is not resolved then it should proceed to the manual handling of the problem.

8. Applications

- It can be deployed on BPO firms.
- Can be used in sales and marketing firms and assistance.

9. Conclusion

Its outcome will be a working chatbot that can handle customer queries and able to use Watson Discovery to help with queries regarding functioning of devices.

10. Future Scope

- Imporved UI
- Continuous chat and interaction mode can be made
- Support for cross-platforms and APIs
- Speech based query and Voice based query resolution
- Using EdgeAI Toolkits for IoT devices real time monitoring.

11. Bibliography

- https://github.com/ibm/watson-discovery-sdu-with-assistant
- https://www.ibm.com/in-en/cloud

Appendix

The required source code is uploaded on github profile in the way of files with .json extension.