Project Report

Intelligent Customer Help Desk with Smart Document Understanding

Made By: Vishv Patel

Introduction

In this task, a chatbot is made which offers a total and simple approach to responding to various arrangements of inquiries posed by the clients. With the assistance of the Watson discovery channel, it can likewise respond to some run of the mill inquiries regarding the activity of a gadget since we have to take care of the proprietor's manual to the Watson discovery channel. The advantage of this sort of chatbot is that it is better than the commonplace chatbot which can address basic inquiries like location and hours. The chatbot is overhauled with the assistance of Watson discovery assortment which is manufacture utilizing smart document understanding.

Our main objective is

- To solve customer's queries as early as possible to save the time of the customer.
- We will use the IBM cloud function that allows Watson assistant to post queries to Watson discovery.
- The goal is to set up a remote connection between the customer and the company

Overview

The run of the mill client care chatbot can respond to straightforward inquiries, for example, store areas and hours, bearings, and perhaps making arrangements. When an inquiry falls outside of the extent of the pre-decided inquiry set, the alternative is regularly to tell the client the inquiry isn't substantial or offer to address a genuine individual. In this task, a chatbot is made which offers a total and simple approach to respond to various arrangements of inquiries posed by the clients. With the assistance of Watson discovery channel, it can likewise address some ordinary inquiries concerning the activity of a gadget since we have to take care of the proprietor's manual to the Watson discovery channel. To make it a stride further, the task will utilize the Smart Document Understanding component of Watson Discovery to prepare it on what text in the proprietors manual is significant and what isn't. This will improve the appropriate responses came back from the inquiries.

Purpose

The aim of the project is to create an Intelligent Customer Help Desk Smart Document Understanding feature us with a typical customer care chatbot that can answer simple questions, such as store locations and hours, directions, and maybe

even making appointments. If a question falls outside of the scope of the pre-determined question set, then tell the customer that the question is not valid or offer to speak to a real person. This problem statement provides a way to create an Smart Help Desk that can be used at all the sectors in a different ways which can help the country to Digitalize. Considering an eclectic combination of all the features I will be able to make Intelligent Customer Help Desk using Artificial Intelligence.

Literature Survey

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.

Proposed Solution

For achieving above purpose, the chatbot should be trained and get familiar with the product manual so that it can answer the complex queries also. So, the solution is that, 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 Human customer representative?" we can return relevant queries of the owner's manual to help solve our customer's problems.

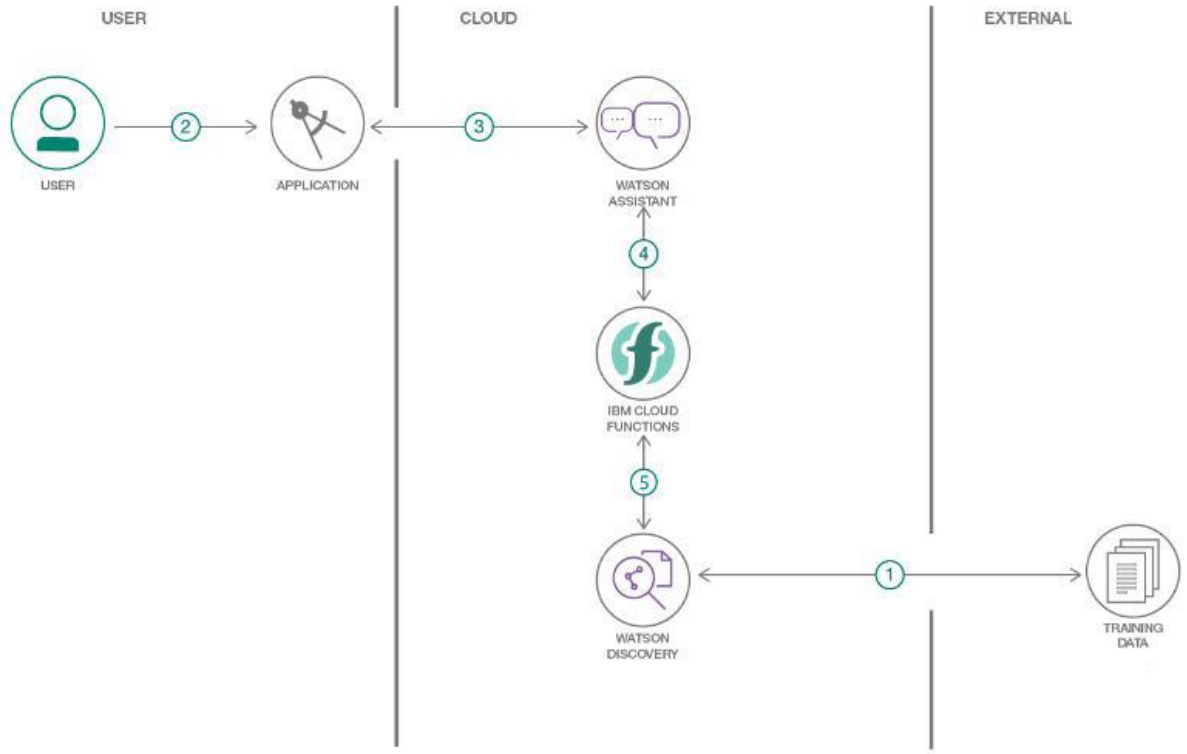
Theoretical Analysis

Chatbots : Chatbot is made up of two words "Chat" representing conversation and "Bot" representing a robot. Hence a chatbot is enabling conversations with a robot.

Types of Chatbots:

1) Scripted Chatbot
2) Intelligent chatbot
3) Application chatbot

Block diagram



- The product manual is annotated utilizing Watson Discovery SDU
- The client connects with the backend server through the application UI. The frontend application UI is a chatbot that connects with the client in a discussion.
- Exchange between the client and backend server is composed utilizing a Watson Assistant discourse expertise.
- In the event that the client asks an item activity inquiry, a hunt question is passed to a predefined IBM Cloud Functions activity.
- The Cloud Functions activity will question the Watson Discovery administration and return the outcomes.

Hardware/Software Designing

1) Create IBM Cloud services
2) Configure Watson Discovery
3) Create IBM Cloud Functions action
4) Configure Watson Assistant
5) Create flow and configure node
6) Deploy and run Node Red web-app

Experimental Investigation

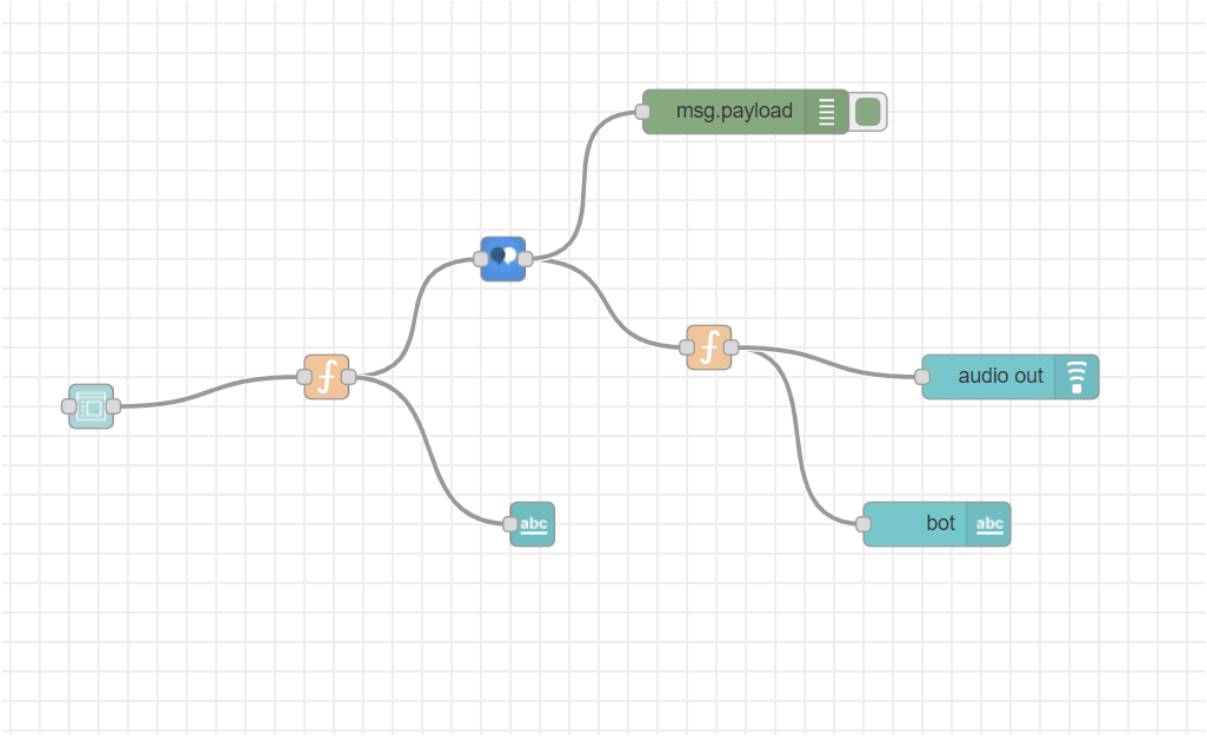
Configure Watson Discovery

- Upload the required document and annotate all the text with different field like title, subtitle, text, footer etc.
- Index the document by Title, subtitle and text field and divide the document with the subtitle fields.
- Run the test queries to see if the SDU works properly.

Setting up IBM Cloud Function

- IBM cloud function will be used to make queries against our Discovery collection.
- The function will be given some input and it will pass this input to the Watson discovery and fetch the output through our chatbot.

Flow Chart



Result

At the end of the project we developed the chatbot which can recognize the queries which are related to the device operation. The model created i.e. a chatbot

would be able to identify any operational question posted by the user and using IBM Watson discovery will redirect the user to the section of the owner's manual where the answer to the question lies.

Advantages / Disadvantages

Advantages

- Improved Customer Service.
- Always-Available Customer Support
- Proactive Customer Interaction
- Increased Customer Engagement
- Reduce manpower.
- Cost efficient.
- No need to divert calls to Customer care Representative for simple queries.

Disadvantages

- Answer to the chatbots can be inappropriate Giving same answer to many question sometimes.
- Chatbots cannot identify the sentiments and intentions of the customers because of some reasons like Network or Technical issues.

Applications

The chatbot can be deployed on almost any website or application to resolve the customer’s queries and also the bot can be deployed on many social media platforms like Facebook, Slack, Telegram etc. They make life easier for customers, are available 24/7, save time (no more long waits to talk to a customer care representative) and they are easy to use.

Conclusion

By Completing the given task, we have developed an intelligent information retrieval system by combining multiple IBM cloud services like Watson Assistant, Watson Discovery and Cloud Function. And we learned that Chatbots can reach out to a large audience on messaging apps and be more effective than humans.

Future Scope

Accuracy of smart document understanding can be increased by using some additional training and involvement of some websites.
And can also be done using Watson Assistant learn skill.

- With use of text-to-speech and speech-to-text service the chatbot can be accessible handsfree.
- Adding Natural Language Processing in the Bot to understand the User
- Statements.
- Voice Capabilities of the Bot.
- Voice Recognition with Bot.

Bibliography

Source Code


```
[
    {
        "id": "c9d759bf.756d88",
        "type": "tab",
        "label": "Watson Assisstant",
        "disabled": false,
        "info": ""
    },
    {
        "id": "5c561c2d.59b754",
        "type": "watson-conversation-v1",
        "z": "c9d759bf.756d88",
        "name": "Assisstant",
        "workspaceid": "f6ea5498-fdf5-4bcd-9566-f627c06a53fe",
        "multiuser": false,
        "context": false,
        "empty-payload": false,
        "service-endpoint": "https://api.eu-
gb.assistant.watson.cloud.ibm.com/instances/70a61e3c-c548-465e-9cab-
8e38bac4e52e",
        "timeout": "",
        "optout-learning": false,
        "x": 375,
        "y": 180,
        "wires": [
            [
                "f6898694.723348",
                "b30b28bd.d30798"
            ]
        ],
        "l": false
    },
    {
        "id": "a82f3762.1f7d38",
        "type": "function",
```

```
    "z": "c9d759bf.756d88",
    "name": "F1",
    "func": "msg.payload = msg.payload.text; \nreturn msg;",
    "outputs": 1,
    "noerr": 0,
    "x": 255,
    "y": 260,
    "wires": [
        [
            "5c561c2d.59b754",
            "749d912d.9911"
        ]
    ],
    "l": false
},
{
    "id": "1492b1c2.c32e7e",
    "type": "ui_form",
    "z": "c9d759bf.756d88",
    "name": "",
    "label": "form",
    "group": "e97ce86.7658f18",
    "order": 1,
    "width": 0,
    "height": 0,
    "options": [
        {
            "label": "Enter the input",
            "value": "text",
            "type": "text",
            "required": true,
            "rows": null
        }
    ],
    "formValue": {
        "text": ""
    },
    "payload": "",
    "submit": "submit",
    "cancel": "cancel",
    "topic": "",
    "x": 95,
    "y": 280,
    "wires": [
        [
            "a82f3762.1f7d38"
        ]
    ],
    "l": false
},
{
    "id": "f6898694.723348",
    "type": "function",
    "z": "c9d759bf.756d88",
    "name": "F2",
```

```
        "func": "var message =
\"\";\n\nif(msg.payload.intents[0].intent ==
\"device_operation\"){\n    for(var i in
msg.payload.context.webhook_result_1.passages){\n        message =
message +
msg.payload.context.webhook_result_1.passages[i].passage_text +
\"\\n\";\n    }\n\n    msg.payload =
message;\n}\n\nelse\nmsg.payload=msg.payload.output.text[0];\n\nretu
rn msg;",
        "outputs": 1,
        "noerr": 0,
        "x": 515,
        "y": 240,
        "wires": [
            [
                "9955c636.e9cb68",
                "5c9f34f5.a0326c"
            ]
        ],
        "l": false
    },
    {
        "id": "749d912d.9911",
        "type": "ui_text",
        "z": "c9d759bf.756d88",
        "group": "e97ce86.7658f18",
        "order": 3,
        "width": 0,
        "height": 0,
        "name": "",
        "label": "You",
        "format": "{{msg.payload}}",
        "layout": "row-spread",
        "x": 395,
        "y": 360,
        "wires": [],
        "l": false
    },
    {
        "id": "9955c636.e9cb68",
        "type": "ui_text",
        "z": "c9d759bf.756d88",
        "group": "e97ce86.7658f18",
        "order": 6,
        "width": 0,
        "height": 0,
        "name": "",
        "label": "bot",
        "format": "{{msg.payload}}",
        "layout": "col-center",
        "x": 670,
        "y": 360,
        "wires": []
    },
    {
        "id": "b30b28bd.d30798",
```

```
        "type": "debug",
        "z": "c9d759bf.756d88",
        "name": "",
        "active": true,
        "tosidebar": true,
        "console": false,
        "tostatus": false,
        "complete": "false",
        "x": 540,
        "y": 80,
        "wires": []
    },
    {
        "id": "5c9f34f5.a0326c",
        "type": "ui_audio",
        "z": "c9d759bf.756d88",
        "name": "",
        "group": "e97ce86.7658f18",
        "voice": "",
        "always": "",
        "x": 720,
        "y": 260,
        "wires": []
    },
    {
        "id": "e97ce86.7658f18",
        "type": "ui_group",
        "z": "",
        "name": "CHATBOT",
        "tab": "449af108.6f704",
        "order": 1,
        "disp": true,
        "width": 13,
        "collapse": false
    },
    {
        "id": "449af108.6f704",
        "type": "ui_tab",
        "z": "",
        "name": "CUSTOMERCARE",
        "icon": "",
        "disabled": false,
        "hidden": false
    }
]
```


References

[1] https://www.ibm.com/cloud/get-started

[2] https://developer.ibm.com/tutorials/how-to-create-a-node-red-starter-application/

[3] https://github.com/watson-developer-cloud/node-red-labs

[4] https://www.ibm.com/watson/products-services

[5] https://developer.ibm.com/components/watson-assistant/series/learning-path-watson-assistant

[6] https://developer.ibm.com/articles/introduction-watson-discovery/
[7] https://cloud.ibm.com/docs/services/discovery?topic=discovery-getting-started
[8] https://medium.com/@rimaibrahim/node-red-watson-discovery-chatbot-telegram-ce616ddcd0d9
[9] https://developer.ibm.com/technologies/web-development/articles/ws-restful
[10] https://cloud.ibm.com/docs/openwhisk?topic=cloud-functions-getting-started