PROJECT REPORT

Project Name: INTELLIGENT CUSTOMER HELP DESK WITH SMART DOCUMENT UNDERSTANDING

Internship Project at SmartInternz

Category: Artificial Intelligence

Name: SRISHTI KALRA

Email Id: srishti.1007.sk@gmail.com

1. INTRODUCTION

1.1 Project Description:

The typical customer care chat bot 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.

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.

- **Project Skill Requirements**: Python, IBM Cloud, IBM Watson
- **Functional Requirements**: IBM Cloud
- **Technical Requirements**: AI, ML, WATSON AI, PYTHON
- **Software Requirements**: Watson assistant, Watson discovery, Node-RED, Cloud Function Actions

1.2 Scope of Work:

- Create a customer care dialog skill in Watson Assistant
- Use Smart Document Understanding to build an enhanced Watson Discovery collection
- Create an IBM Cloud Functions web action that allows Watson Assistant to post queries to Watson Discovery
- Build a web application with integration to all these services & deploy the same on IBM Cloud Platform

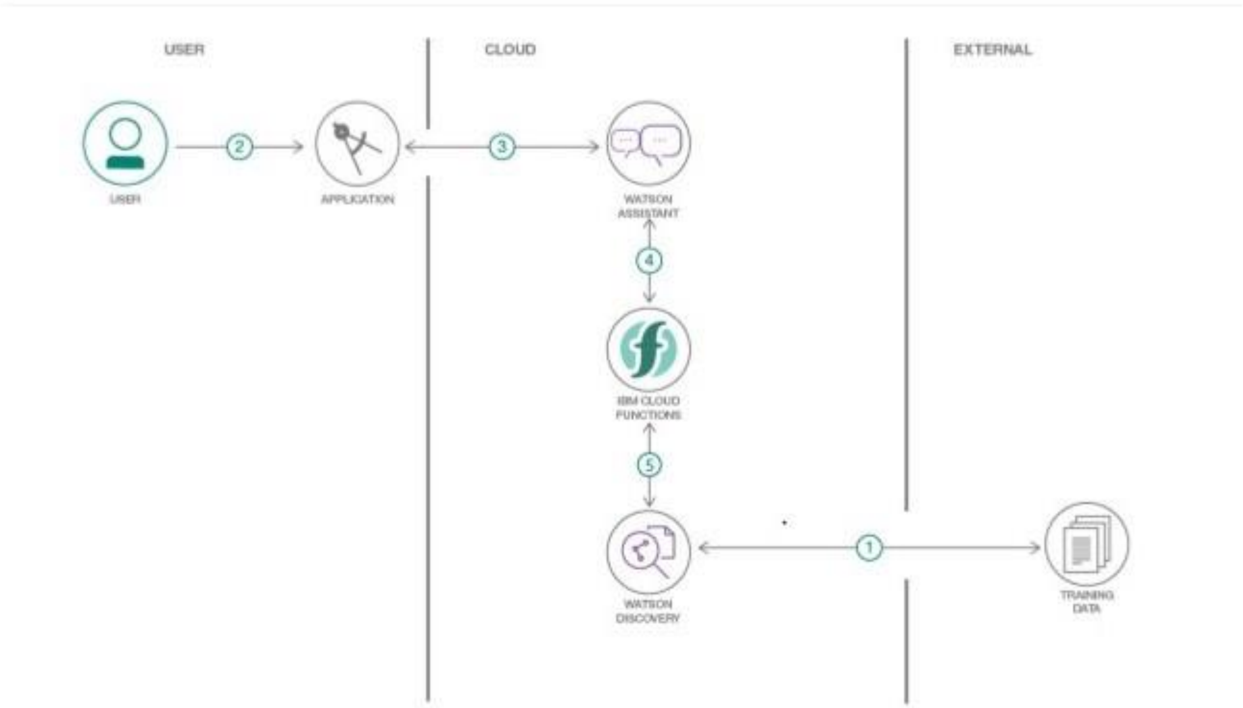


Fig1: Flowchart showing the steps for building the bot

2. LITERATURE SURVEY

2.1 Problem Encountered:

A typical customer care chat bot 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 response to the problem mentioned, we aim to develop a chat bot that uses smart document understanding to answer user’s questions. The chat bot shall be trained using IBM Watson Assistant. This chat bot shall also use the feature of IBM Watson Discovery to provide relevant sections of user’s manual as answer to user queries.

The following steps have been implemented in IBM Cloud to reach the desired solution:

1) Create IBM Cloud services

Create the following IBM Cloud Services:

- Watson Discovery
- Watson Assistant
- Node-RED

2) Configure Watson Discovery

a) Launch the Watson Discovery tool and create a new data collection by selecting the ‘Upload your own data’ option.

b) Give the data collection a unique name. When prompted, select and upload the user manual in pdf format from your device.

c) For adding the Smart Document Understanding feature to generate better query responses, click ‘Configure data’ option to start the SDU process. 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.

3) Create Action for IBM Cloud Functions

a) Start the IBM Cloud Functions service by selecting ‘Create Resource’ from IBM cloud. Create Resource from the IBM Cloud dashboard. Enter functions as the filter, then select the Functions card.

b) From the Functions main panel, click on the Actions tab. Then click Create. From the Create panel, select the Create Action option.

c) On the Create Action panel, provide a unique Action Name, keep the default package and select the Node.js 10 runtime.

d) Click the Create button to create the action.

e) Once the action is made, go to the Code Tab. In the editor window, cut and paste in the code from the disco-action.js file found in the action’s directory of your local repository. It simply connects to the Discovery service, makes a query against the collection, and then returns the response.

f) Select the Parameters tab. Add the following keys: url, environment_id, collection_id, iam_apikey

g) For values, please use the values associated with the Discovery service you created in the previous step. Now that the credentials are set, return to the Code panel and press the Invoke button.

h) Next, go to the Endpoints tab. Click the checkbox for Enable as Web Action. This will generate a public endpoint URL which would be used by the Watson Assistant.

4) Configure Watson Assistant

a) 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.

b) Create a new intent that can detect when the user is asking about operating the Product. From the Customer Care Sample Skill Panel, select Intents tab.

c) Click the Create intent button. Name the intent #Product_Information, and at a minimum, enter the possible user questions regarding the device.

d) Create new Dialog node named Product_Information to handle our intent.

e) Enable Webhooks for this Dialog. The dialog node should have a Return variable set automatically to $webhook_result_1. This is the variable name you can use to access the result from Discovery service query.

f) Go to Options tab, select Webhooks and enter the previously generated URL.

g) We can try running the chatbot using Try it option and can also check the context variables,

5) Create the UI using Node-RED

a) Install dashboard

b) We use the following nodes:

    i) Inject

    ii) Assistant

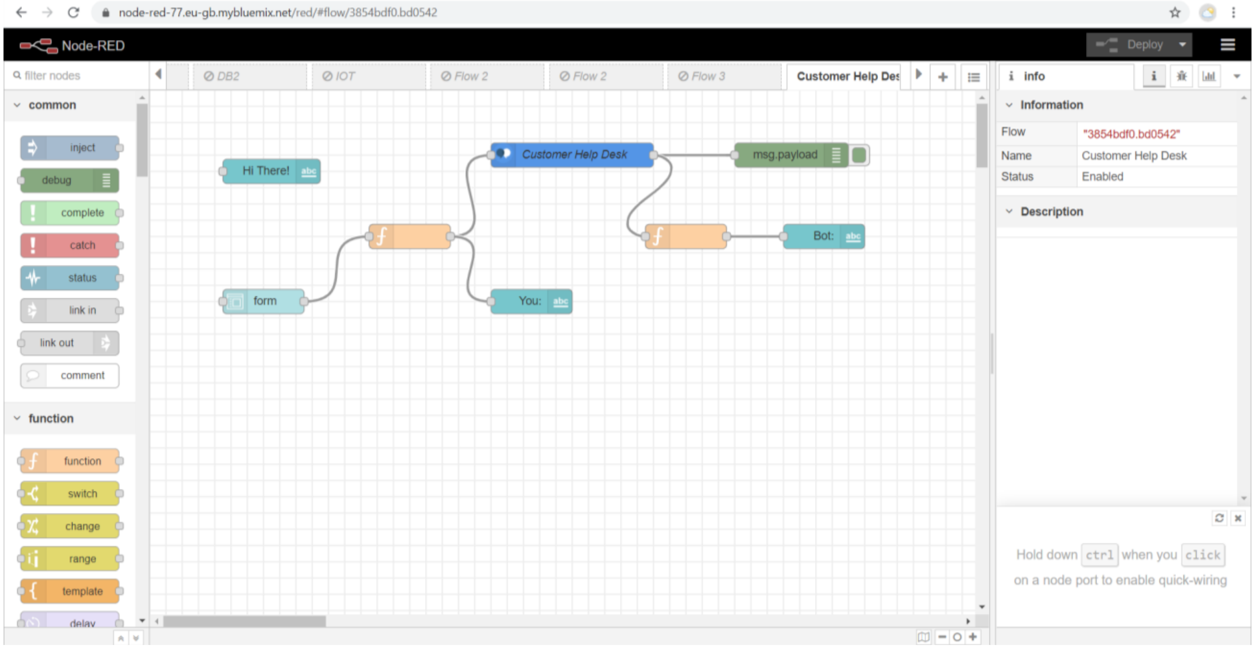
    iii) Debug

    iv) Function

    v) Ui_Form

    vi) Ui_Text

The flow is shown in the screenshot shown below:



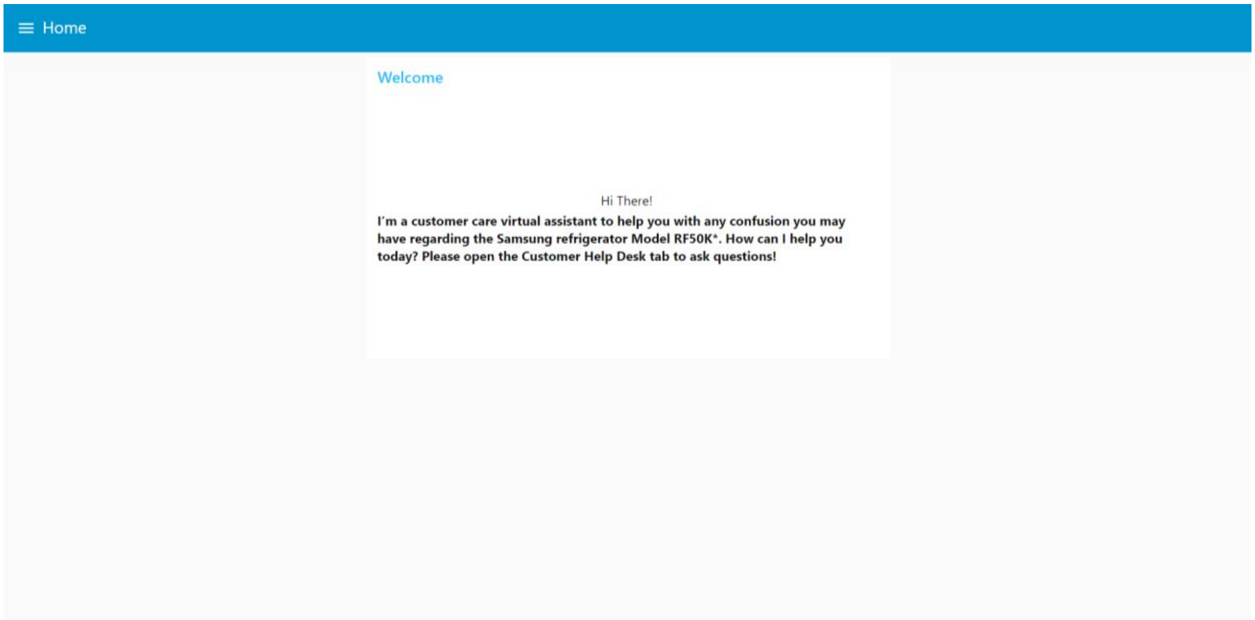
Flow	"3854bdf0.bd0542"
Name	Customer Help Desk
Status	Enabled

Hold down ctrl when you click on a node port to enable quick-wiring

Fig2: Node-RED flow for the chat bot

6) Deploy and run Node-RED Application

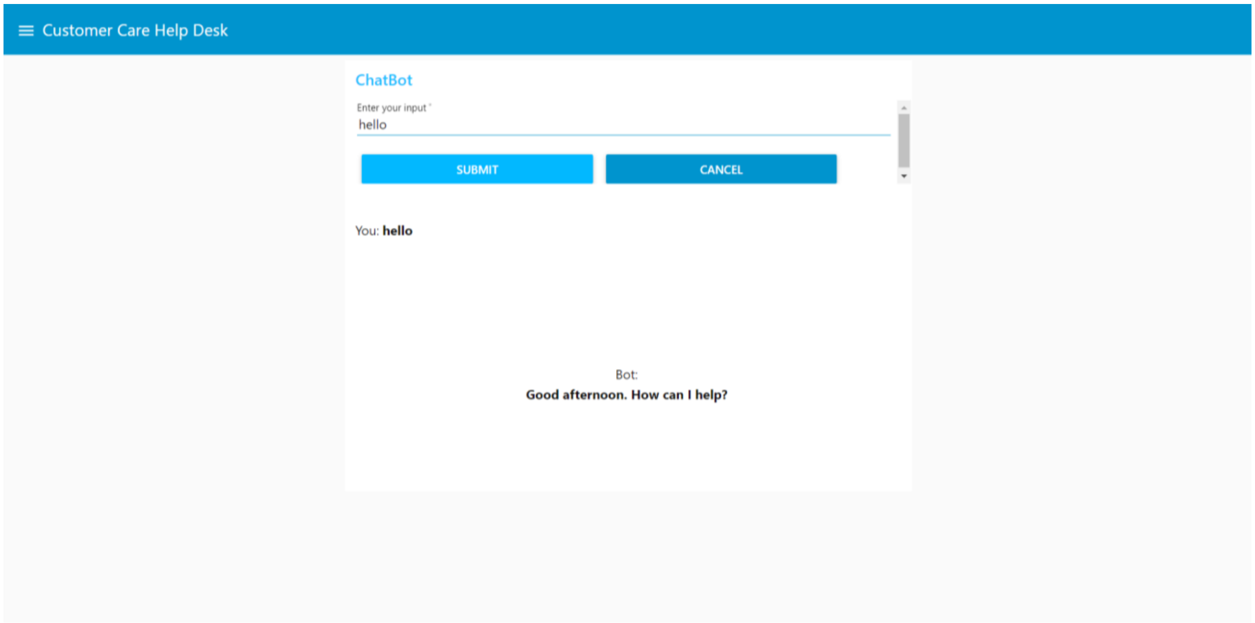
a) Deploy the Node-RED flow and open the user interface.



Hi There!

I'm a customer care virtual assistant to help you with any confusion you may have regarding the Samsung refrigerator Model RF50K*. How can I help you today? Please open the Customer Help Desk tab to ask questions!

Fig3(a): User Interface for chat bot



hello

You: **hello**

Bot:
Good afternoon. How can I help?

Fig3(b): User Interface for chat bot

3. THEORETICAL ANALYSIS

3.1 Requirement Analysis:

Project Requirements: Python, IBM Cloud, IBM Watson

Functional Requirements: IBM cloud

Technical Requirements: AI, ML, WATSON AI, PYTHON

Software Requirements: Watson assistant, Watson discovery.

4. PRACTICAL APPLICATION

4.1 Watson Discovery:

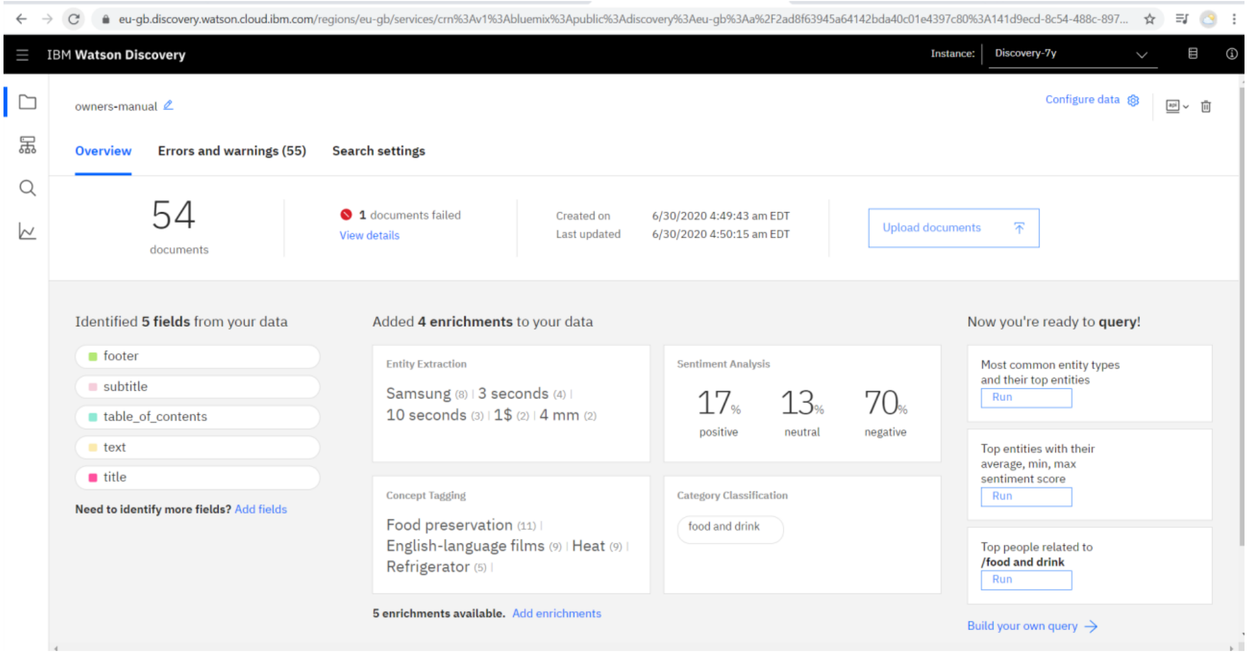

eu-gb.discovery.watson.cloud.ibm.com/regions/eu-gb/services/crn%3Av1%3Abluemix%3Apublic%3Adiscovery%3Aeu-gb%3Aa%2F2ad8f63945a64142bda40c01e4397c80%3A141d9ecd-8c54-488c-897...
IBM Watson Discovery
Instance: Discovery-7y
owners-manual
Configure data
Overview
Errors and warnings (55)
Search settings
54
documents
1 documents failed
View details
Created on 6/30/2020 4:49:43 am EDT
Last updated 6/30/2020 4:50:15 am EDT
Upload documents
Identified 5 fields from your data
footer
subtitle
table_of_contents
text
title
Need to identify more fields? Add fields
Added 4 enrichments to your data
Entity Extraction
Samsung (8) | 3 seconds (4) | 10 seconds (3) | 1$ (2) | 4 mm (2)
Sentiment Analysis
17% positive
13% neutral
70% negative
Concept Tagging
Food preservation (11) | English-language films (9) | Heat (9) | Refrigerator (5) |
Category Classification
food and drink
5 enrichments available. Add enrichments
Now you're ready to query!
Most common entity types and their top entities
Run
Top entities with their average, min, max sentiment score
Run
Top people related to /food and drink
Run
Build your own query →

Fig4(a): Watson Discovery window, after uploading and configuring the document

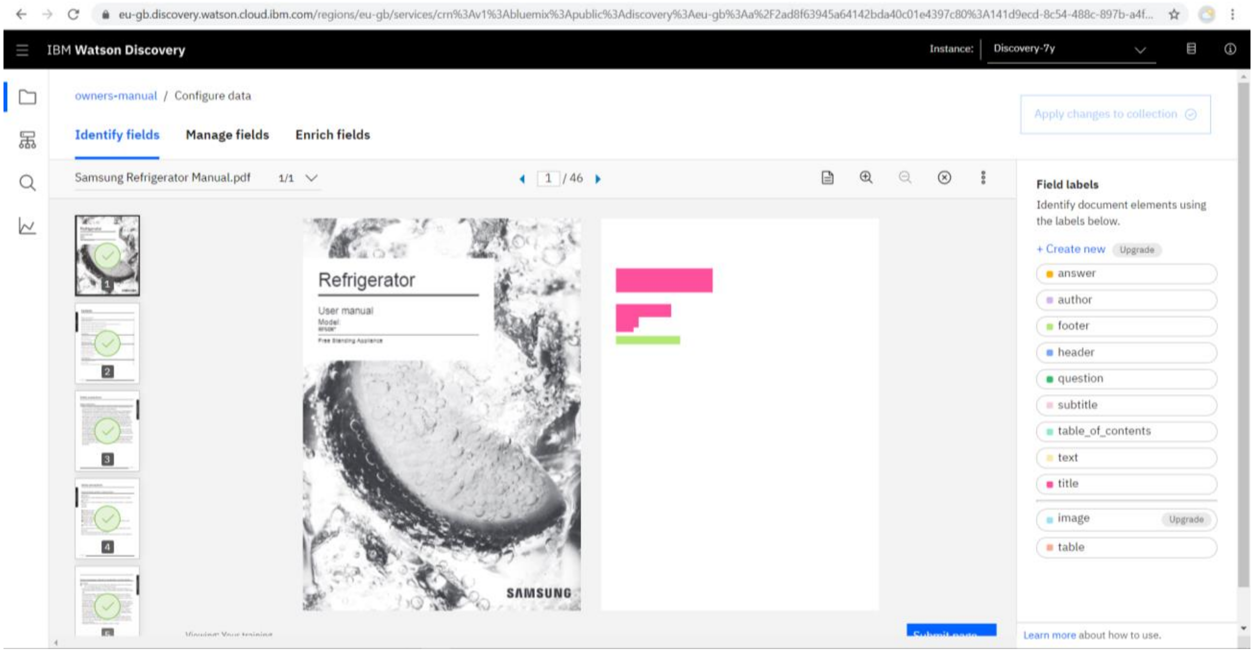
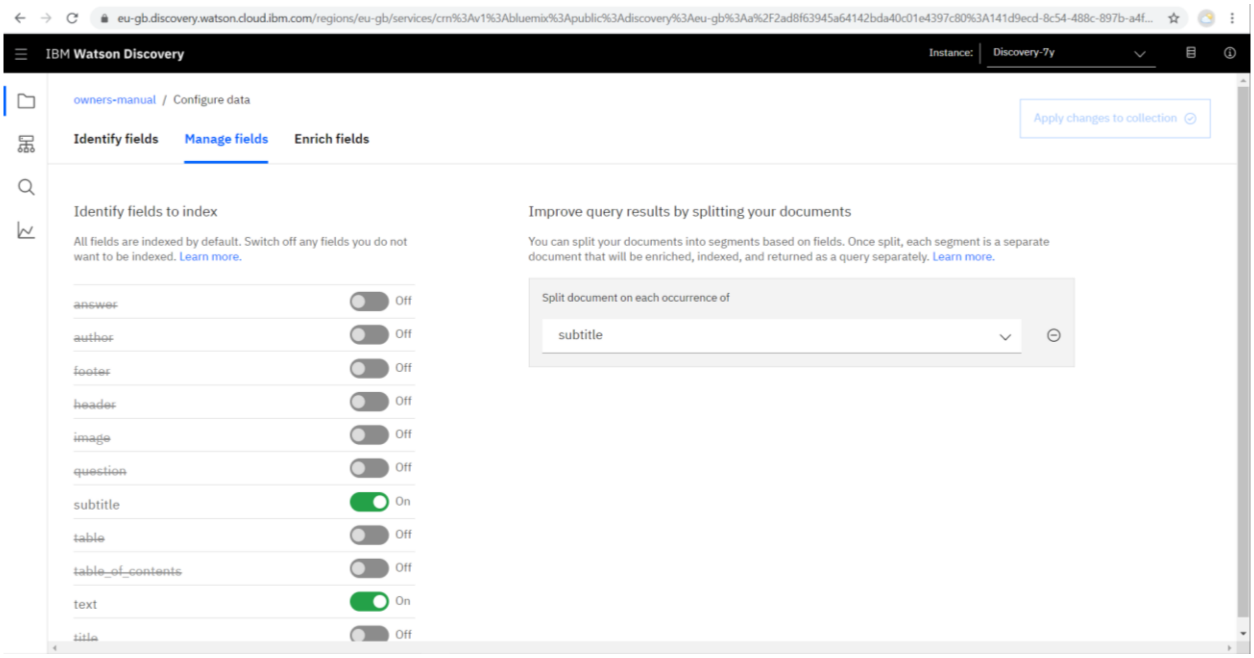

eu-gb.discovery.watson.cloud.ibm.com/regions/eu-gb/services/crn%3Av1%3Abluemix%3Apublic%3Adiscovery%3Aeu-gb%3Aa%2F2ad8f63945a64142bda40c01e4397c80%3A141d9ecd-8c54-488c-897b-a4f...
IBM Watson Discovery
Instance: Discovery-7y
owners-manual / Configure data
Apply changes to collection
Identify fields
Manage fields
Enrich fields
Samsung Refrigerator Manual.pdf
1/1
1 / 46
Refrigerator
User manual
SAMSUNG
Field labels
Identify document elements using the labels below.
+ Create new
Upgrade
answer
author
footer
header
question
subtitle
table_of_contents
text
title
image
Upgrade
table
Submit page
Learn more about how to use.

Fig4(b): Watson Discovery window showing the step of annotation of the document



IBM Watson Discovery

owners-manual / Configure data

Identify fields | Manage fields | Enrich fields

Identify fields to index

All fields are indexed by default. Switch off any fields you do not want to be indexed. Learn more.

~~answer~~ Off
~~author~~ Off
~~footer~~ Off
~~header~~ Off
~~image~~ Off
~~question~~ Off
subtitle On
~~table~~ Off
~~table_of_contents~~ Off
text On
~~title~~ Off

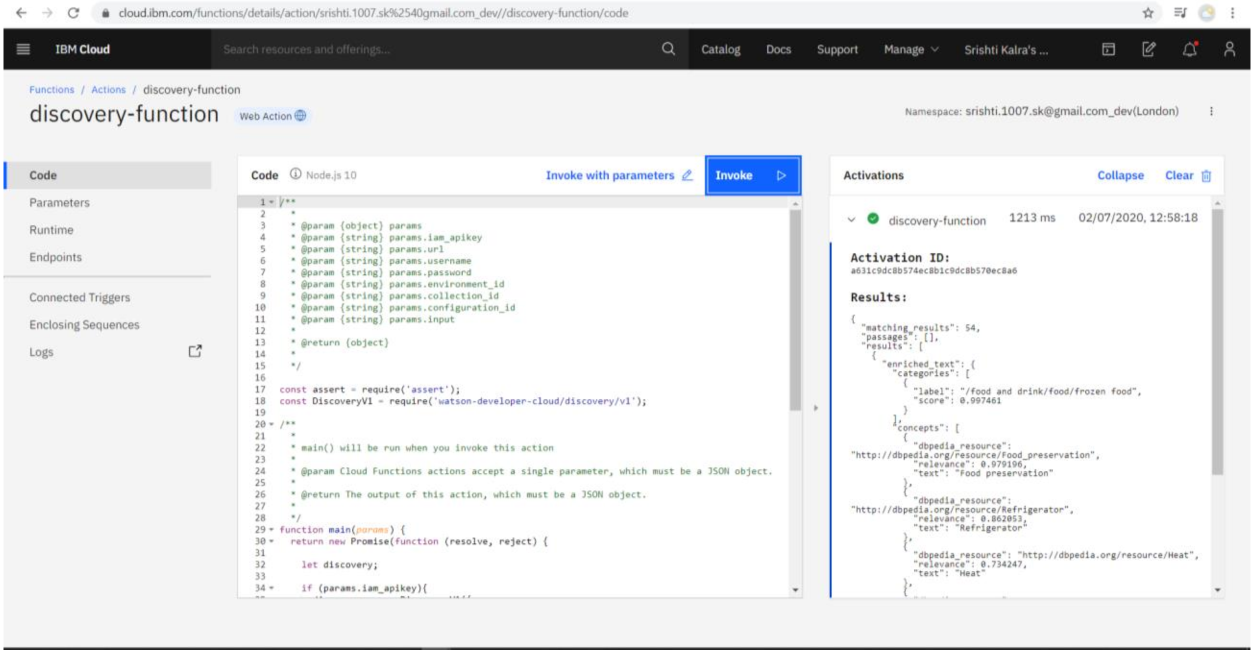
Improve query results by splitting your documents

You can split your documents into segments based on fields. Once split, each segment is a separate document that will be enriched, indexed, and returned as a query separately. Learn more.

Split document on each occurrence of: subtitle

Fig4(c): Managing the fields so as to keep only the relevant fields selected

4.2 Cloud Function:



Functions / Actions / discovery-function

discovery-function (Web Action)

Namespace: srishti.1007.sk@gmail.com_dev(London)

Code — Node.js 10

```
/**
  *
  * @param {object} params
  * @param {string} params.iam_apikey
  * @param {string} params.url
  * @param {string} params.username
  * @param {string} params.password
  * @param {string} params.environment_id
  * @param {string} params.collection_id
  * @param {string} params.configuration_id
  * @param {string} params.input
  *
  * @return {object}
  *
  */

const assert = require('assert');
const DiscoveryV1 = require('watson-developer-cloud/discovery/v1');

/**
  *
  * main() will be run when you invoke this action
  *
  * @param Cloud Functions actions accept a single parameter, which must be a JSON object.
  *
  * @return The output of this action, which must be a JSON object.
  *
  */
function main(params) {
  return new Promise(function (resolve, reject) {

    let discovery;

    if (params.iam_apikey){
```

Activations

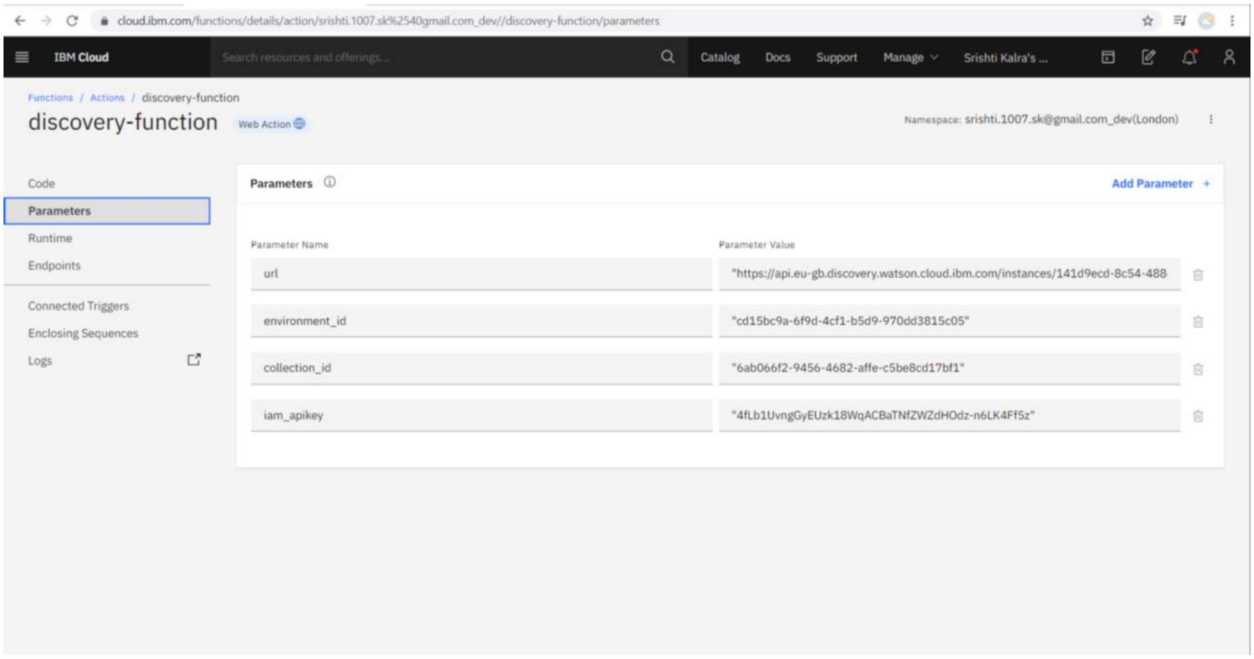
discovery-function 1213 ms 02/07/2020, 12:58:18

Activation ID:
a631c9dc8b574ec8b1c9dc8b570ec8a6

Results:

```
{
  "matching_results": 54,
  "passages": [],
  "results": [
    {
      "enriched_text": {
        "categories": [
          {
            "label": "/food and drink/food/frozen food",
            "score": 0.997461
          }
        ],
        "concepts": [
          {
            "dbpedia_resource":
"http://dbpedia.org/resource/Food_preservation",
            "relevance": 0.979196,
            "text": "Food preservation"
          },
          {
            "dbpedia_resource":
"http://dbpedia.org/resource/Refrigerator",
            "relevance": 0.862053,
            "text": "Refrigerator"
          },
          {
            "dbpedia_resource": "http://dbpedia.org/resource/Heat",
            "relevance": 0.734247,
            "text": "Heat"
          },
          {
```

Fig5(a): IBM Cloud Function window with the code for the web action


cloud.ibm.com/functions/details/action/srishti.1007.sk%2540gmail.com_dev//discovery-function/parameters
IBM Cloud
Search resources and offerings...
Catalog
Docs
Support
Manage
Srishti Kalra's ...
Functions / Actions / discovery-function
Web Action
Code
Parameters
Runtime
Endpoints
Connected Triggers
Enclosing Sequences
Logs
Add Parameter +

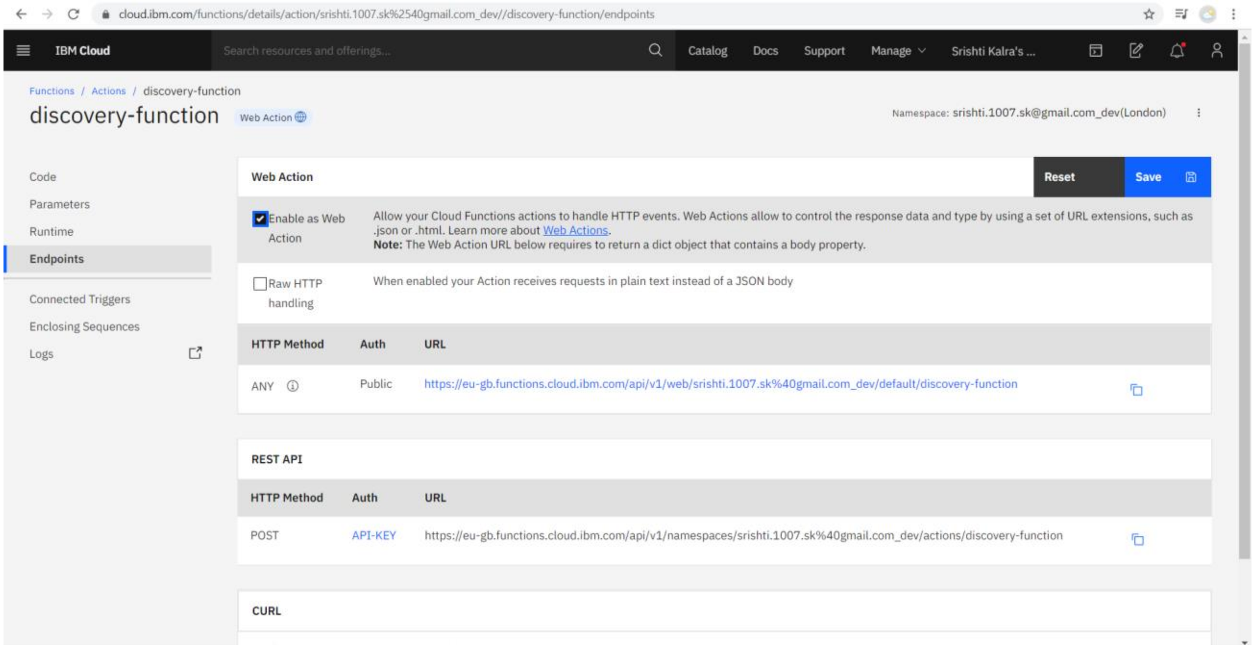
discovery-function

Namespace: srishti.1007.sk@gmail.com_dev(London)

Parameters

Parameter Name	Parameter Value
url	"https://api.eu-gb.discovery.watson.cloud.ibm.com/instances/141d9ecd-8c54-488
environment_id	"cd15bc9a-6f9d-4cf1-b5d9-970dd3815c05"
collection_id	"6ab066f2-9456-4682-affe-c5be8cd17bf1"
iam_apikey	"4fLb1UvngGyEUzk18WqACBaTNfZWZdHOdz-n6LK4Ff5z"

Fig5(b): IBM Cloud Function window showing the parameters to be set


cloud.ibm.com/functions/details/action/srishti.1007.sk%2540gmail.com_dev//discovery-function/endpoints
IBM Cloud
Search resources and offerings...
Catalog
Docs
Support
Manage
Srishti Kalra's ...
Functions / Actions / discovery-function
Web Action
Code
Parameters
Runtime
Endpoints
Connected Triggers
Enclosing Sequences
Logs
Reset
Save

discovery-function

Namespace: srishti.1007.sk@gmail.com_dev(London)

Web Action

☑ Enable as Web Action — Allow your Cloud Functions actions to handle HTTP events. Web Actions allow to control the response data and type by using a set of URL extensions, such as .json or .html. Learn more about Web Actions.
Note: The Web Action URL below requires to return a dict object that contains a body property.

☐ Raw HTTP handling — When enabled your Action receives requests in plain text instead of a JSON body

HTTP Method	Auth	URL
ANY	Public	https://eu-gb.functions.cloud.ibm.com/api/v1/web/srishti.1007.sk%40gmail.com_dev/default/discovery-function

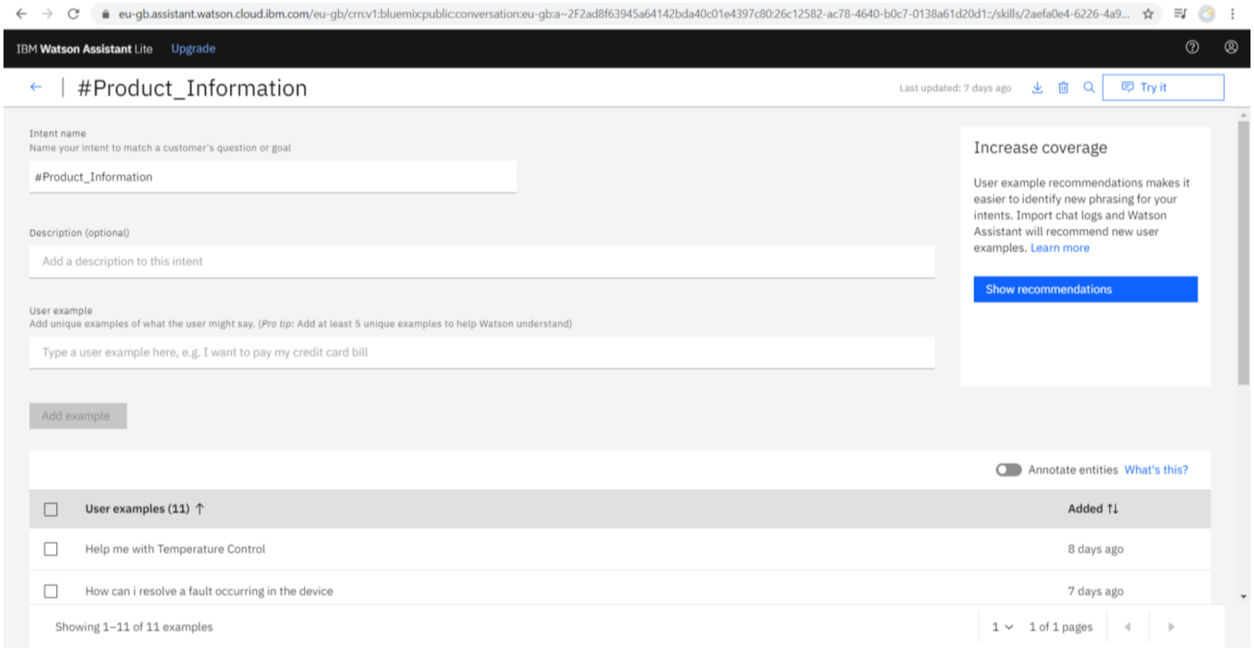
REST API

HTTP Method	Auth	URL
POST	API-KEY	https://eu-gb.functions.cloud.ibm.com/api/v1/namespaces/srishti.1007.sk%40gmail.com_dev/actions/discovery-function

CURL

Fig5(c): IBM Cloud Function window with the Web action link generated

4.3 Watson Assistant:



eu-gb.assistant.watson.cloud.ibm.com/eu-gb/crn:v1:bluemix:public:conversation:eu-gb:a~2F2ad8f63945a64142bda40c01e4397c80:26c12582-ac78-4640-b0c7-0138a61d20d1::/skills/2aefa0e4-6226-4a9...

#Product_Information

Intent name
Name your intent to match a customer's question or goal

#Product_Information

Description (optional)

Add a description to this intent

User example
Add unique examples of what the user might say. (*Pro tip*: Add at least 5 unique examples to help Watson understand)

Type a user example here, e.g. I want to pay my credit card bill

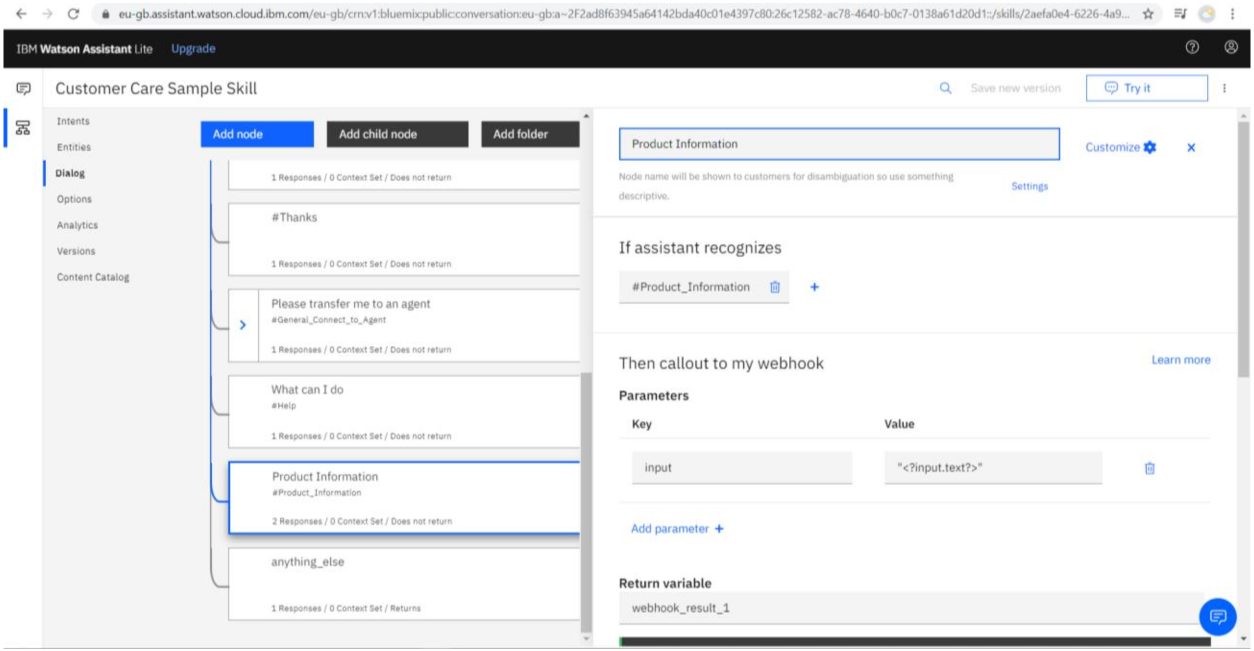
Increase coverage

User example recommendations makes it easier to identify new phrasing for your intents. Import chat logs and Watson Assistant will recommend new user examples. Learn more

□	User examples (11) ↑	Added ⇅
□	Help me with Temperature Control	8 days ago
□	How can i resolve a fault occurring in the device	7 days ago

Showing 1–11 of 11 examples

Fig6(a): IBM Watson Assistant window showing the created intent



eu-gb.assistant.watson.cloud.ibm.com/eu-gb/crn:v1:bluemix:public:conversation:eu-gb:a~2F2ad8f63945a64142bda40c01e4397c80:26c12582-ac78-4640-b0c7-0138a61d20d1::/skills/2aefa0e4-6226-4a9...

Customer Care Sample Skill

1 Responses / 0 Context Set / Does not return

#Thanks
1 Responses / 0 Context Set / Does not return

Please transfer me to an agent
#General_Connect_to_Agent
1 Responses / 0 Context Set / Does not return

What can I do
#Help
1 Responses / 0 Context Set / Does not return

Product Information
#Product_Information
2 Responses / 0 Context Set / Does not return

anything_else
1 Responses / 0 Context Set / Returns

Product Information

Node name will be shown to customers for disambiguation so use something descriptive.

If assistant recognizes

#Product_Information +

Then callout to my webhook

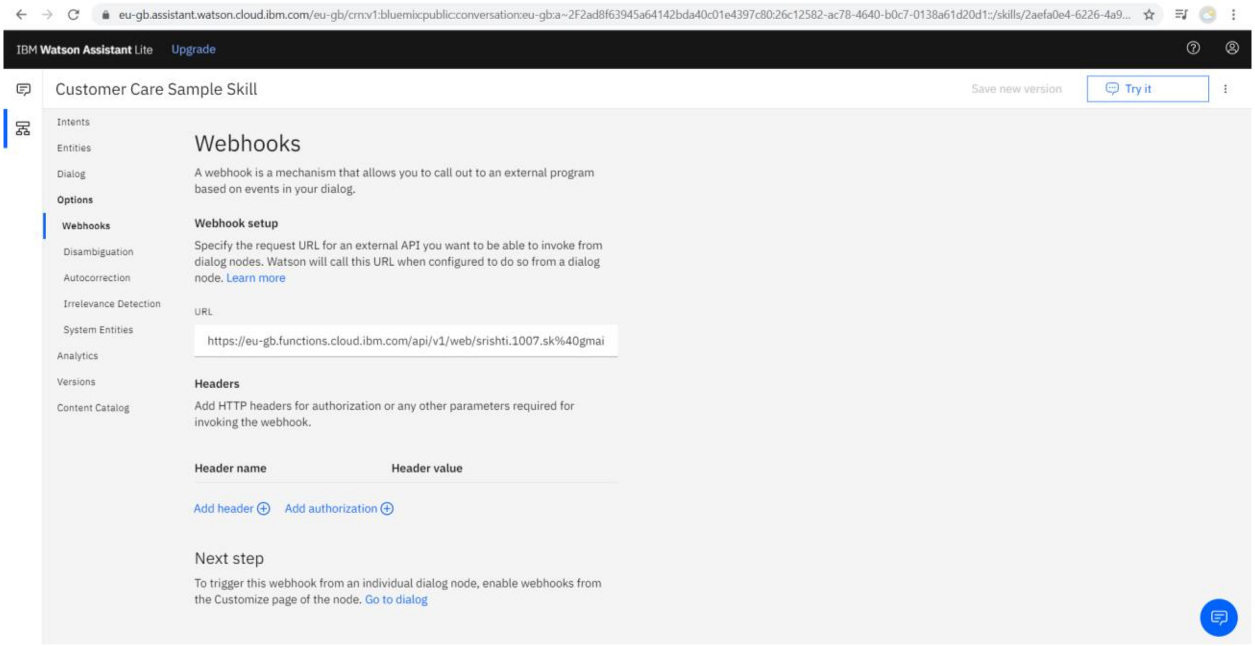
Parameters

Key	Value
input	"<?input.text?>"

Return variable

webhook_result_1

Fig6(b): IBM Watson Assistant window showing the created dialog


eu-gb.assistant.watson.cloud.ibm.com/eu-gb/crn:v1:bluemix:public:conversation:eu-gb:a~2F2ad8f63945a64142bda40c01e4397c80:26c12582-ac78-4640-b0c7-0138a61d20d1::/skills/2aefa0e4-6226-4a9...
IBM Watson Assistant Lite
Upgrade
Customer Care Sample Skill
Save new version
Try it
Intents
Entities
Dialog
Options
Webhooks
Disambiguation
Autocorrection
Irrelevance Detection
System Entities
Analytics
Versions
Content Catalog

Webhooks

A webhook is a mechanism that allows you to call out to an external program based on events in your dialog.

Webhook setup

Specify the request URL for an external API you want to be able to invoke from dialog nodes. Watson will call this URL when configured to do so from a dialog node. Learn more

URL

https://eu-gb.functions.cloud.ibm.com/api/v1/web/srishti.1007.sk%40gmai

Headers

Add HTTP headers for authorization or any other parameters required for invoking the webhook.

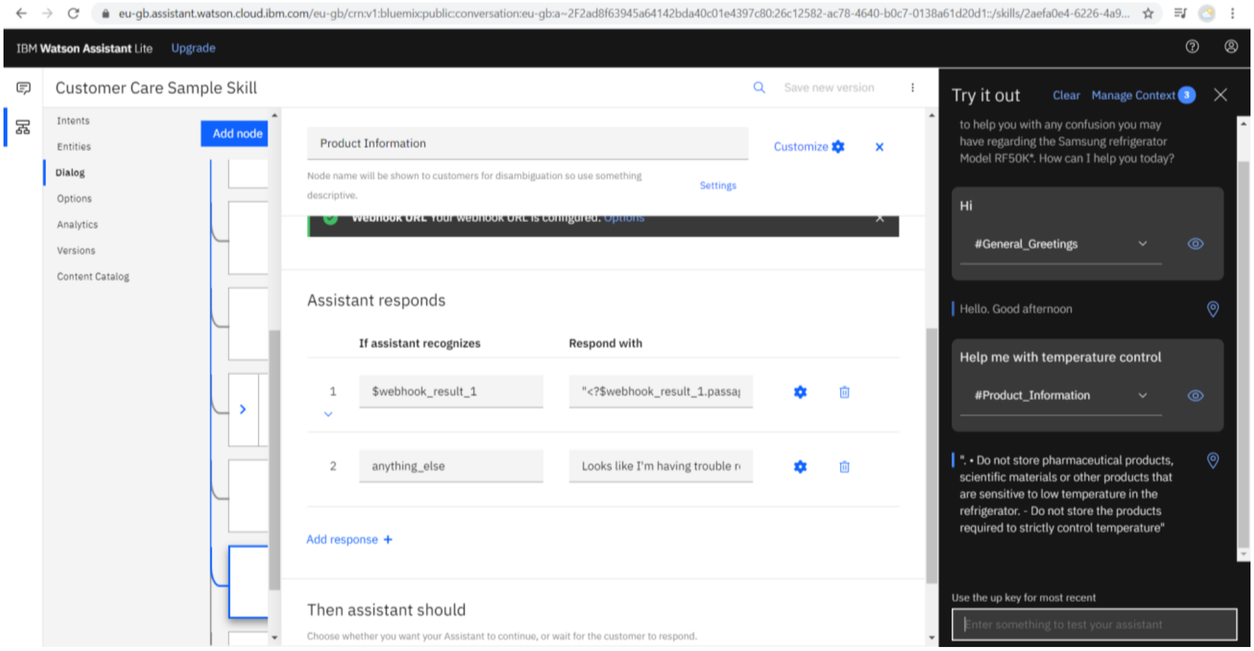
Header name	Header value

Add header ⊕ Add authorization ⊕

Next step

To trigger this webhook from an individual dialog node, enable webhooks from the Customize page of the node. Go to dialog

Fig6(c): IBM Watson Assistant window showing the URL entered for Webhooks


eu-gb.assistant.watson.cloud.ibm.com/eu-gb/crn:v1:bluemix:public:conversation:eu-gb:a~2F2ad8f63945a64142bda40c01e4397c80:26c12582-ac78-4640-b0c7-0138a61d20d1::/skills/2aefa0e4-6226-4a9...
IBM Watson Assistant Lite
Upgrade
Customer Care Sample Skill
Save new version
Intents
Entities
Dialog
Options
Analytics
Versions
Content Catalog
Add node
Customize
Settings
Try it out
Clear
Manage Context
3

Product Information

Node name will be shown to customers for disambiguation so use something descriptive.

Webhook URL Your webhook URL is configured. Options

Assistant responds

	If assistant recognizes	Respond with
1	$webhook_result_1	"<?$webhook_result_1.passa
2	anything_else	Looks like I'm having trouble r

Add response +

Then assistant should

Choose whether you want your Assistant to continue, or wait for the customer to respond.

to help you with any confusion you may have regarding the Samsung refrigerator Model RF50K*. How can I help you today?

Hi

#General_Greetings

Hello. Good afternoon

Help me with temperature control

#Product_Information

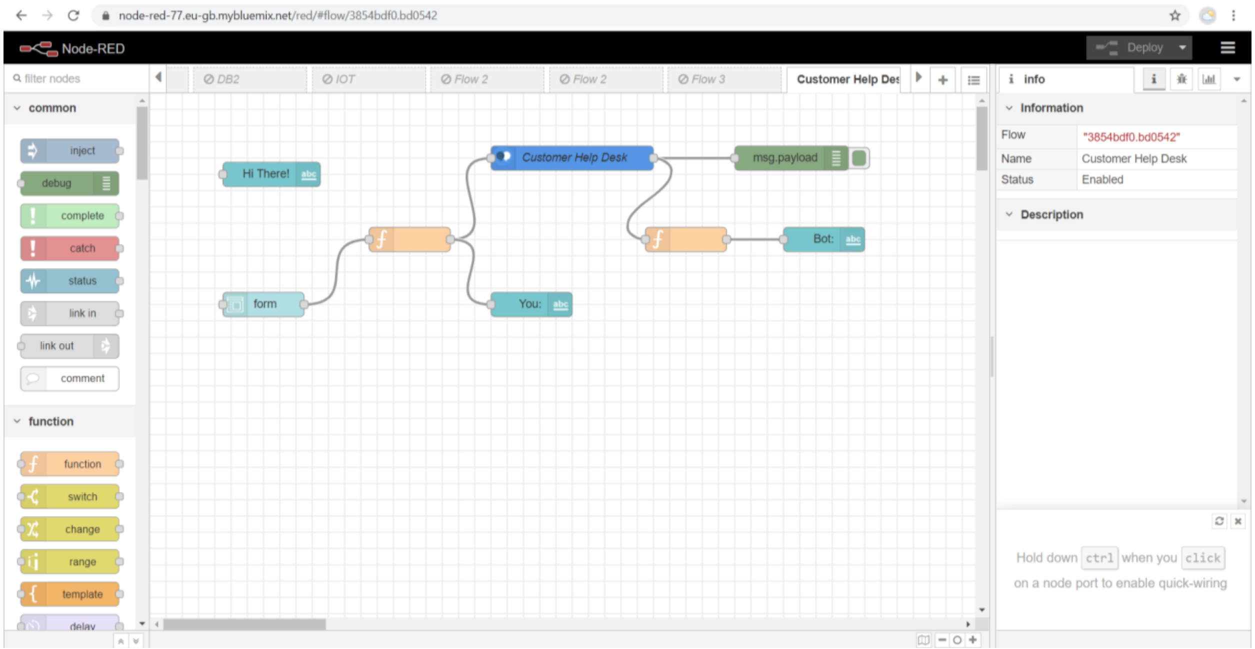
". • Do not store pharmaceutical products, scientific materials or other products that are sensitive to low temperature in the refrigerator. - Do not store the products required to strictly control temperature"

Use the up key for most recent

Enter something to test your assistant

Fig6(d): IBM Watson Assistant window showing the trial of Bot

4.4 Node-RED:



node-red-77.eu-gb.mybluemix.net/red/#flow/3854bdf0.bd0542

Flow "3854bdf0.bd0542"

Name Customer Help Desk

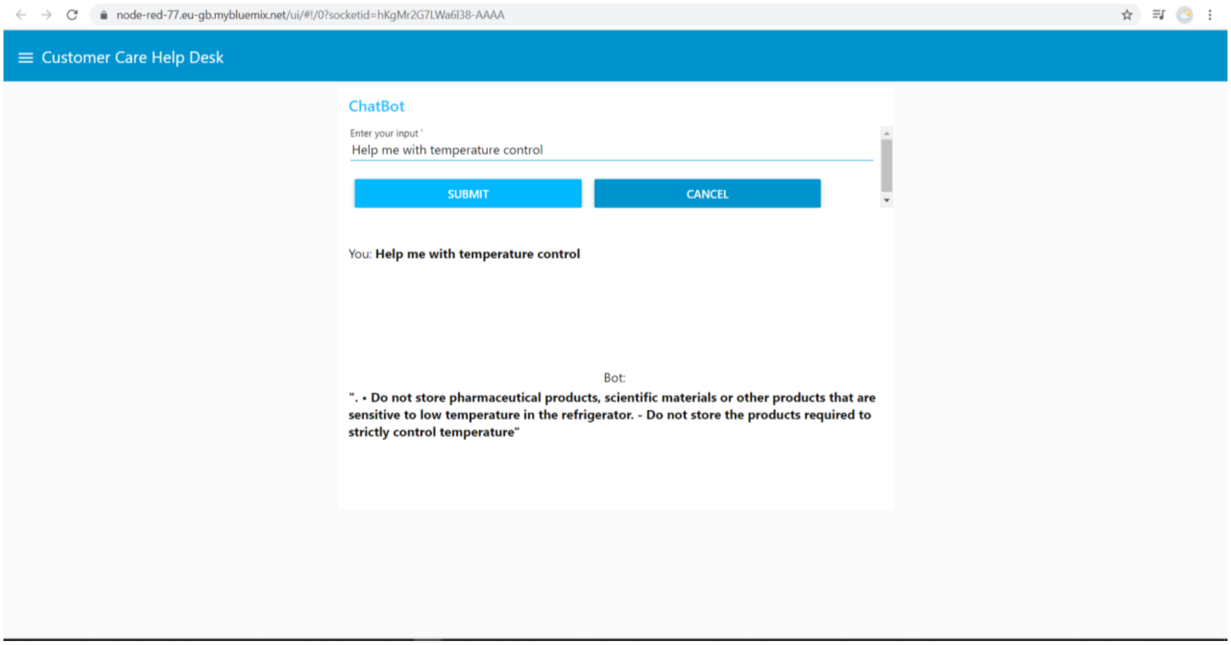
Status Enabled

Hold down ctrl when you click on a node port to enable quick-wiring

Fig7: Node-RED flow for Chat Bot

5. RESULTS

The chat bot was successfully made using the mentioned IBM Cloud Services. Below are a few screen shots of the working chat bot.


node-red-77.eu-gb.mybluemix.net/ui/#!/0?socketid=hKgMr2G7LWa6l38-AAAA
Customer Care Help Desk
ChatBot
Enter your input *
SUBMIT
CANCEL

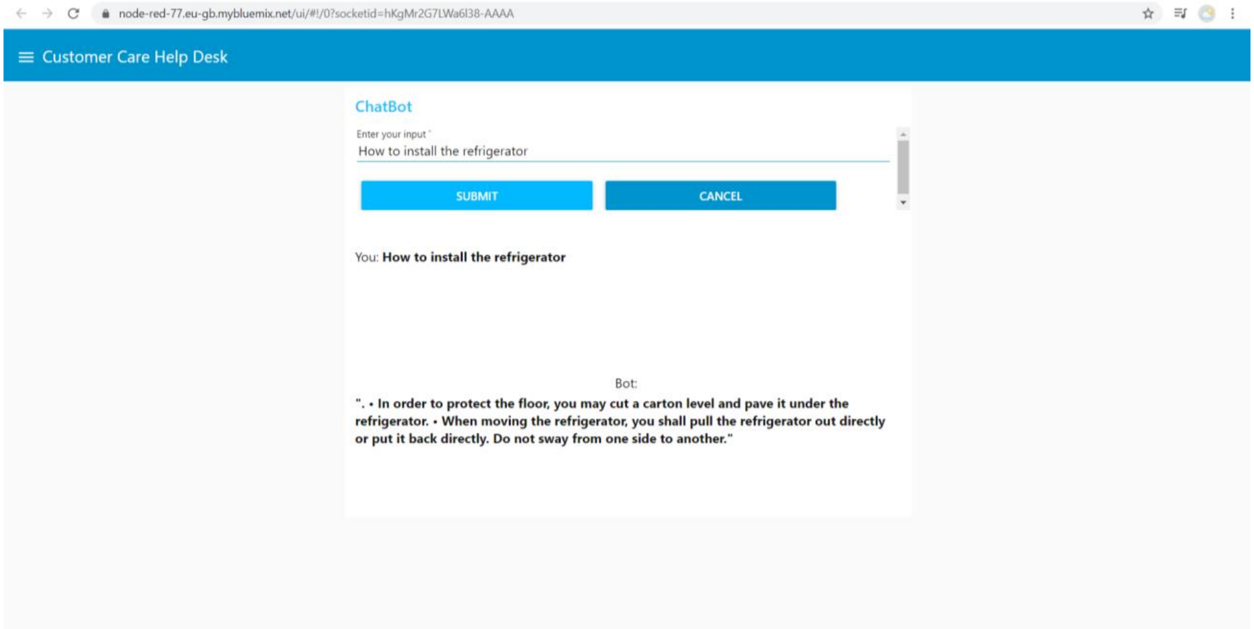
Help me with temperature control

You: **Help me with temperature control**

Bot:

". • Do not store pharmaceutical products, scientific materials or other products that are sensitive to low temperature in the refrigerator. - Do not store the products required to strictly control temperature"

Fig8(a)


node-red-77.eu-gb.mybluemix.net/ui/#!/0?socketid=hKgMr2G7LWa6l38-AAAA
Customer Care Help Desk
ChatBot
Enter your input *
SUBMIT
CANCEL

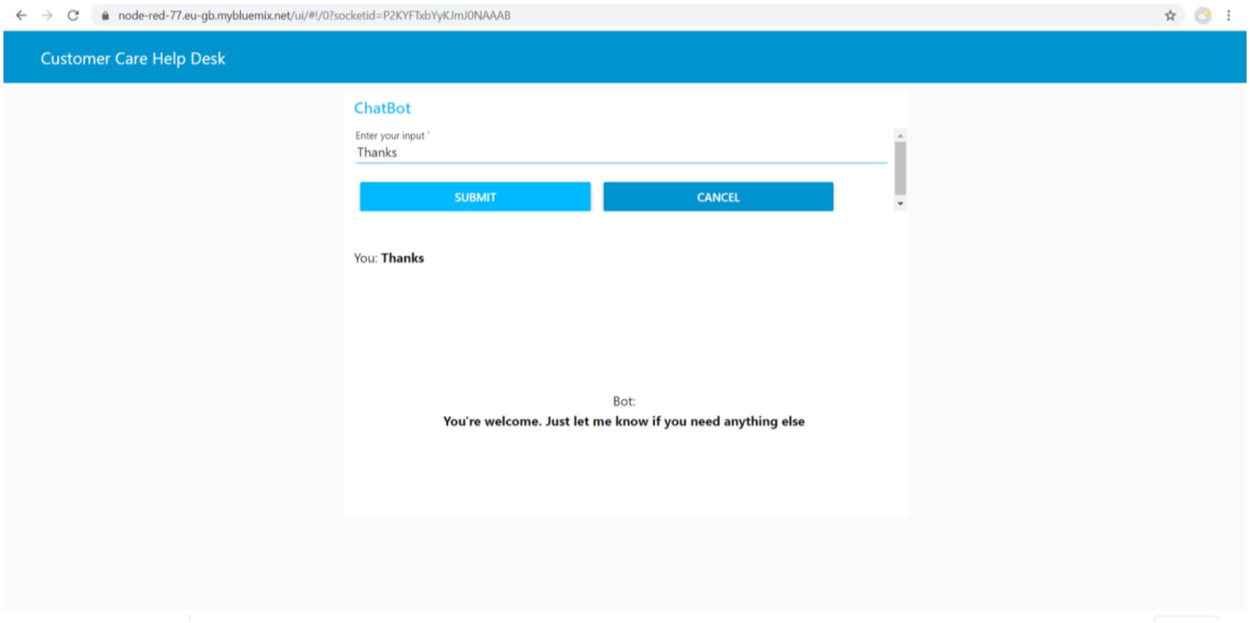
How to install the refrigerator

You: **How to install the refrigerator**

Bot:

". • In order to protect the floor, you may cut a carton level and pave it under the refrigerator. • When moving the refrigerator, you shall pull the refrigerator out directly or put it back directly. Do not sway from one side to another."

Fig8(b)



node-red-77.eu-gb.mybluemix.net/ui/#!/0?socketid=P2KYFTxbYyKJmJ0NAAAB

Customer Care Help Desk

ChatBot

Enter your input *
Thanks

You: **Thanks**

Bot:
You're welcome. Just let me know if you need anything else

Fig8(c)

Fig8(a), 8(b) and 8(c) display the working User Interface of the chat bot

6. APPLICATIONS

Some applications can be: -

1) A chatbot trained with Smart Document Understanding can be implemented for use in any customer care platform, making the job of customer care agents much easier.

2) Help User: This chatbot will be useful for the user to ask the assistant the queries related to the Product and will give them clear guidance about the Product. If the Assistant doesn't know about a certain query, it will redirect to the correct person for it.

3) Companionship: The primary function of the chatbot is to be a virtual companion – To speak with senior people on general topics like the weather, nature, hobbies, movies, music, news, etc.

The chatbot asks questions, reacts to the answers, is able to speak on various topics, and share interesting news and facts from Google.

4) Content delivery: Media Publishers have realized that chatbots are a powerful way to engage with their audiences and monitor engagement to gain valuable insights on reader interests. Chat with the CNN and Wall Street Journal Chatbots on Facebook Messenger and receive the latest news directly in Messenger, without having to visit their websites.

7. CONCLUSION

This chatbot will be useful for the user to ask the assistant the queries related to the Product and will give them clear guidance about the product. If the Assistant doesn’t know about a certain query, it will redirect to the correct person for it. Chatbots are quickly making transformational changes and allowing businesses to thrive through customer interactions. The feedback and survey through chatbots strengthen the position of businesses as they analyze the reason behind different levels of customer approval. Use of conversational AI chatbots only means better engagement and relentless need for customer satisfaction in the near future.

8. FUTURE SCOPE

Future Scope of this chatbot can be by adding the following to make it more advance: -

1) Integration with IoT Devices: Car speakers, smart home devices, and wearables are just a few examples where the virtual assistant is departing from its original hardware and making its way to in-context devices. These integrations ensure that virtual assistants can always be near their human and ready to support any need.

2) Voice-control: Voice recognition can be added with the virtual assistant. Then the customer can control application by using his voice.

3) Smarter Virtual Assistants: Much of what virtual assistants do now are basic skills, such as

retrieving data and basic computation. As natural language processing (NLP) continues to mature, virtual assistants will improve their comprehension and response capabilities, allowing for their use to become more widespread and complex. Also, as machine learning progresses, we may see virtual assistants become smarter and begin to learn and predict customer needs.

9. REFERENCES

1. https://www.youtube.com/embed/LOCkV-mENq8
2. https://www.ibm.com/cloud/get-started
3. https://developer.ibm.com/tutorials/how-to-create-a-node-red-starter-application/
4. https://nodered.org/
5. https://github.com/watson-developer-cloud/node-red-labs
6. https://www.youtube.com/embed/s7wmiS2mSXY
7. https://www.ibm.com/watson/products-services
8. https://www.youtube.com/embed/5z3i5IsBVnk
9. https://developer.ibm.com/components/watson-assistant/series/learning-path-watson-assistant
10. https://medium.com/@rimaibrahim/node-red-watson-discovery-chatbot-telegram-ce616ddcd0d9
11. https://www.youtube.com/embed/Jpr3wVH3FVA