PROJECT REPORT

“Intelligent Customer Help Desk With Smart Document Understanding”

JAIMIN .D

Email ID – jaimindharani1213@gmail.com

SBID : SB20200006386

Project ID - SPS_PRO_99

CONTENTS

1 INTRODUCTION

1.1 Overview - The project "Intelligent Customer Help Desk With Smart Document Understanding " will be a customer care chatbot is which can answer simple questions, such as store locations and hours, directions, and maybe even making appointments. 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. 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 Requirements

The chatbot should answer all the simple questions and questions about the operation of a device efficiently.

Functional Requirements

1) The chatbot should respond to any input it receives. 2) The chatbot should be able to query to Watson Discovery Service. 3) It should be able to use the Smart Document Understanding feature of Watson Discovery.

Software Requirements

Github, Slack, IBM Cloud Platform, IBM Watson Services.

Project Deliverables

Efficient and exact answers to queries posted by the users will be the project deliverables.

Project Team

I will be developing it individually.

Project Schedule

The project is scheduled to be completed by 5th June 2020.

1.2 **Purpose -** The chatbot should answer all the simple questions and questions about the operation of a device efficiently.

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

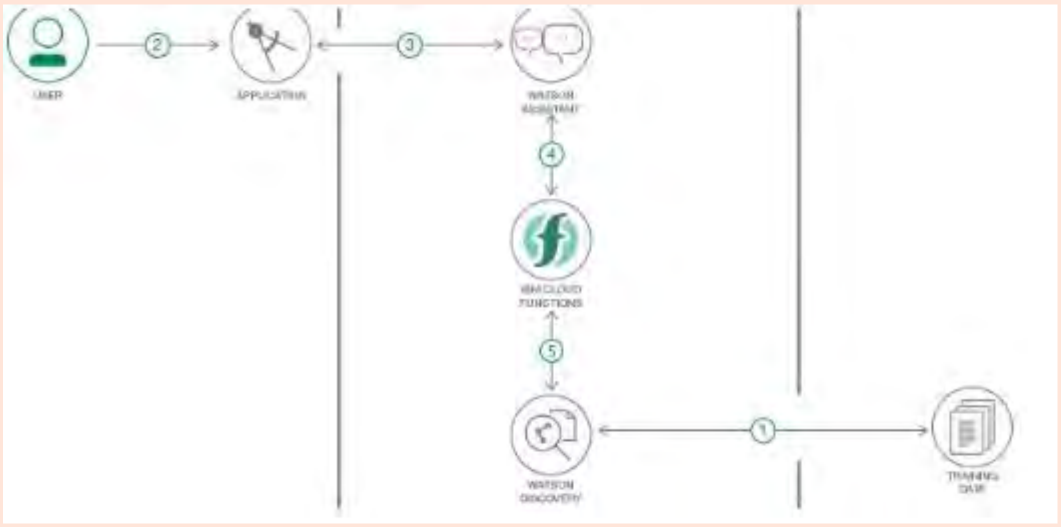
2 LITERATURE SURVEY

2.1 **Existing problem** – The chatbot isn't able to answer when the question falls out of scope. That is if it is beyond a particular set of common queries and if it is of a specific topic then the chatbot isn't able to answer effectively.

2.2 **Proposed solution** – The Watson assistant is connected to Watson discovery through webhooks and cloud functions which helps in answering specific questions proposed by the users.

3 THEORITICAL ANALYSIS

3.1 Block diagram / 3.2 Software designing



1. The document is annotated using Watson Discovery SDU.
2. The user interacts with the backend server via the app UI. The frontend app UI is a chatbot that engages the user in a conversation.
3. Dialog between the user and backend server is coordinated using a Watson Assistant dialog skill.
4. If the user asks the product operation question, a search query is passed to a predefined IBM Cloud Function action.
5. Cloud function action will query the Watson Discovery service and return the results.

4 EXPERIMENTAL INVESTIGATIONS:

1) Create IBM Cloud Services

- **a.** Watson Discovery
- **b.** Watson Assistant
- **c.** Node Red

2) Configure Watson Discovery

After creating and launching the discovery from the Catalog, Import the document on which on which we need to train the discovery service. We have selected the ecobee3 user guide located in the data directory of our local repository.

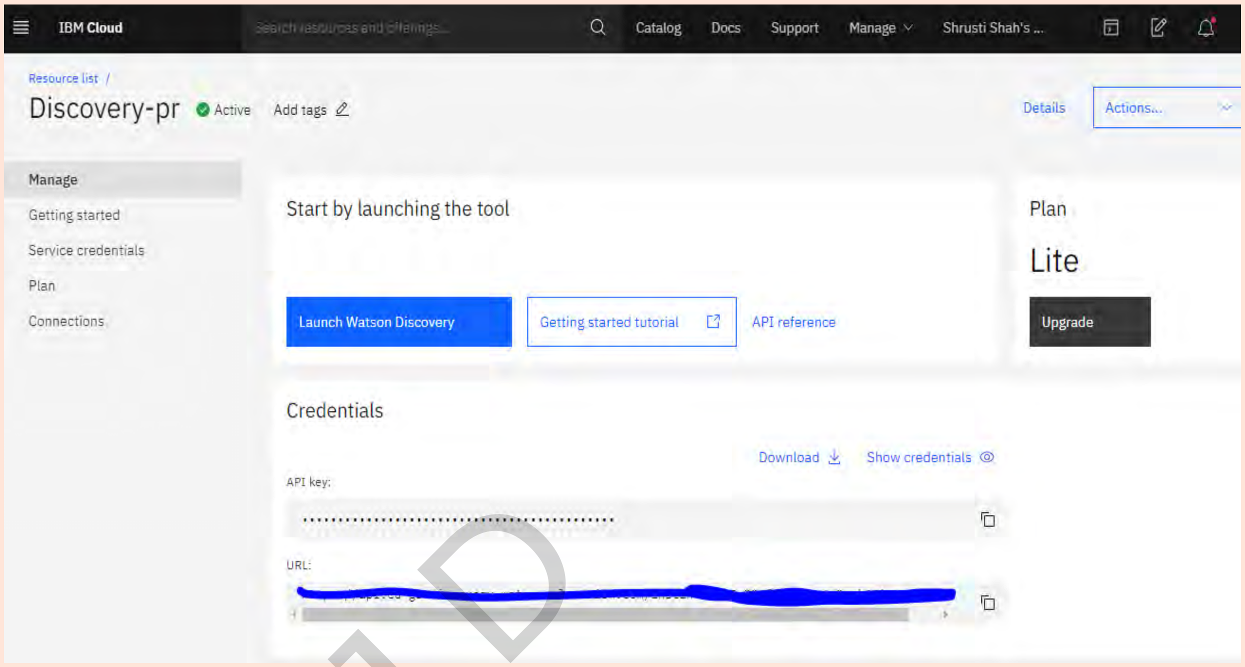
The Ecobee3 is a popular residential thermostat that has a WIFI interface and multiple configuration options. The result of the queries performed without configuring the data present in the document won't be that accurate. But the results improve significantly after applying SDU (Smart Document Understanding).

This can be done easily by clicking on the configure setting and then labelling each word or element present in the document as their respective label such as title, subtitle, text, image and Footer. Some of the labels are not present in the lite plan.

In the lite plan we are provided with limited content of IBM Watson, the labels help us in segmentation of the document which helps the discovery to understand the document better and provide better results. The results provided by the discovery can be improved, all the results are shown in assistant in which the discovery finds the sentiment to be positive i.e. matching between the question or query entered by the user and the data of the document. Better the sentiment analysis accurate the results are.

Follow the below mentioned steps:

- After creating the discovery from the catalog, we will be redirected to this base page of discovery where the name of the discovery along with its API Key and URL are mentioned. These credentials will be used in further steps.



IBM Cloud | Search resources and offerings... | Catalog | Docs | Support | Manage | Shrusti Shah's ...

Resource list /

Discovery-pr Active Add tags

Details | Actions...

Manage
Getting started
Service credentials
Plan
Connections

Start by launching the tool

Launch Watson Discovery | Getting started tutorial | API reference

Plan

Lite

Upgrade

Credentials

Download | Show credentials

API key:

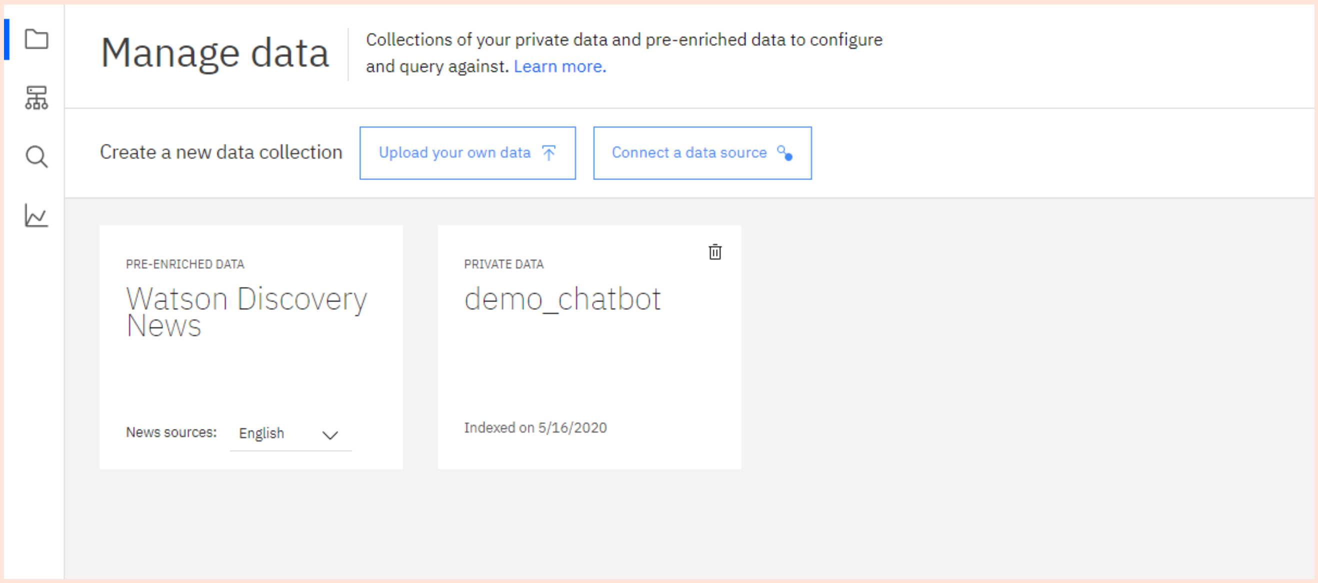
••

URL:

[redacted]

Click on the Launch Watson Discovery [1] to launch the discovery.

- Now in the next step we have to upload the data by clicking, upload your data. Here we have already uploaded the data as manual.



Manage data

Collections of your private data and pre-enriched data to configure and query against. Learn more.

Create a new data collection | Upload your own data | Connect a data source

PRE-ENRICHED DATA

Watson Discovery News

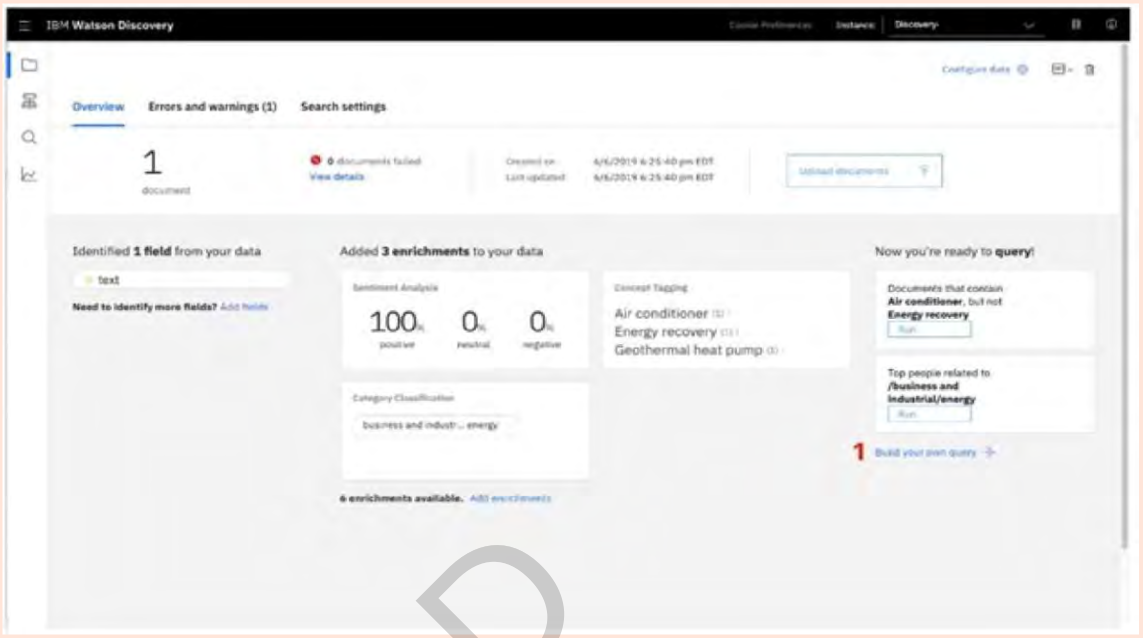
News sources: English

PRIVATE DATA

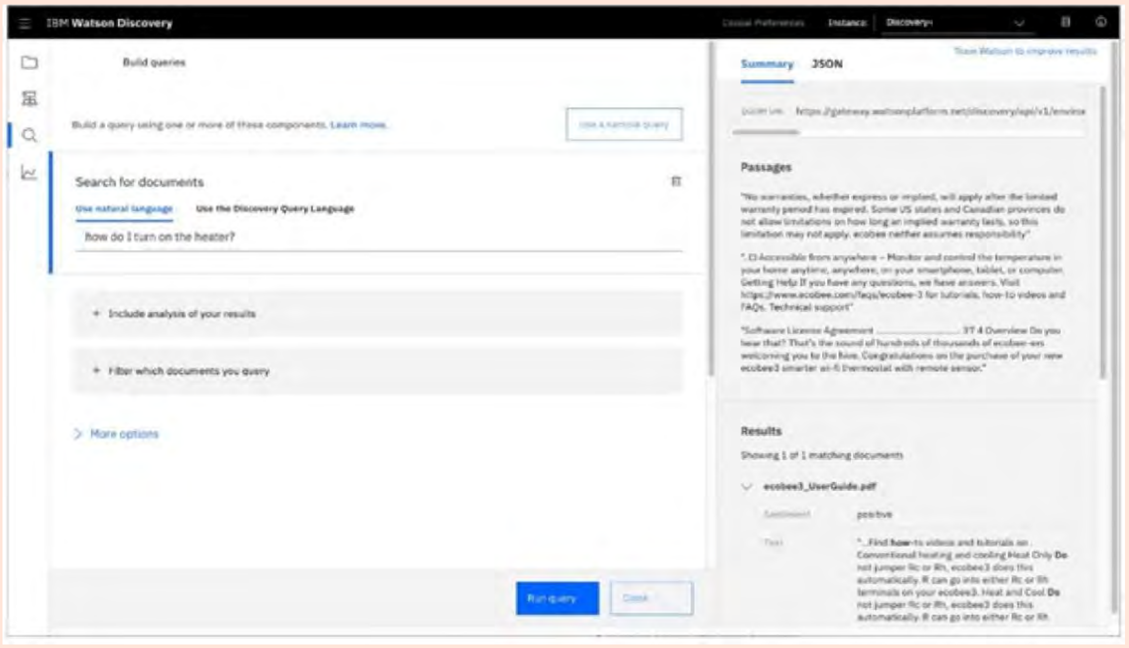
demo_chatbot

Indexed on 5/16/2020

- After uploading the document we will see the page like this, we can ignore the warnings section as it is due to the normalization process and is of no worry. Now click on the build your own query [1] , so that we can have an idea how the results have significantly improved after configuring the data.


IBM Watson Discovery
Instance: Discovery
Configure data
Overview
Errors and warnings (1)
Search settings
1
document
0 documents failed
View details
Created on 6/6/2019 6:25:40 pm EDT
Last updated 6/6/2019 6:25:40 pm EDT
Upload documents
Identified 1 field from your data
text
Need to identify more fields? Add fields
Added 3 enrichments to your data
Sentiment Analysis
100% positive
0% neutral
0% negative
Concept Tagging
Air conditioner
Energy recovery
Geothermal heat pump
Category Classification
business and industr... energy
6 enrichments available. Add enrichments
Now you're ready to query!
Documents that contain Air conditioner, but not Energy recovery
Run
Top people related to /business and industrial/energy
Run
1
Build your own query

- Enter the queries related to thermostat and view the results. As you will see, the results are not very useful and not even that relevant. The next step is to annotate the document with SDU.


IBM Watson Discovery
Instance: Discovery
Build queries
Build a query using one or more of these components. Learn more.
Use a sample query
Search for documents
Use natural language
Use the Discovery Query Language
How do I turn on the heater?
Include analysis of your results
Filter which documents you query
More options
Run query
Clear
Summary
JSON
Train Watson to improve results
Query URL
Passages
Results
Showing 1 of 1 matching documents
ecobee3_UserGuide.pdf
Sentiment
positive
Text

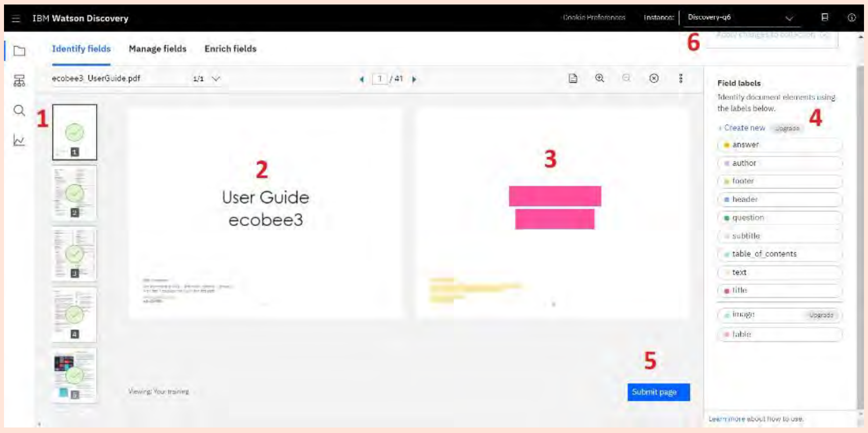
"No warranties, whether express or implied, will apply after the limited warranty period has expired. Some US states and Canadian provinces do not allow limitations on how long an implied warranty lasts, so this limitation may not apply. ecobee neither assumes responsibility"

"...Accessible from anywhere – Monitor and control the temperature in your home anytime, anywhere, on your smartphone, tablet, or computer. Getting Help If you have any questions, we have answers. Visit https://www.ecobee.com/faqs/ecobee-3 for tutorials, how-to videos and FAQs. Technical support"

"Software License Agreement .... 37 4 Overview Do you hear that? That's the sound of hundreds of thousands of ecobee-ers welcoming you to the hive. Congratulations on the purchase of your new ecobee3 smarter wi-fi thermostat with remote sensor."

"...Find **how**-to videos and tutorials on . Conventional heating and cooling Heat Only **Do** not jumper Rc or Rh, ecobee3 does this automatically. R can go into either Rc or Rh terminals on your ecobee3. Heat and Cool **Do** not jumper Rc or Rh, ecobee3 does this automatically. R can go into either Rc or Rh

- Below is the layout of Identify fields tab of the SDU annotation panel:



The aim is to annotate all the pages of the document, so that discovery can learn what text is important and what text can be ignored.

[1] is the list of pages we have in the document. In the lite plan we are unable to upload a document more than 50,000 words and any document with more than that will be trimmed to this range.

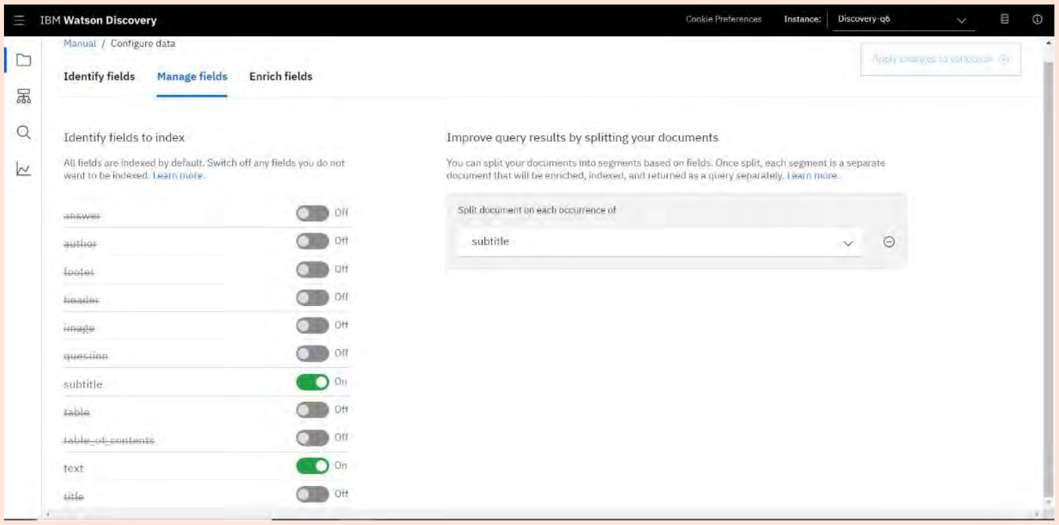
[2] is the current page being annotated.

[3] is the page in which we have to provide a label to each text, for example as shown above we have the Title and text, at the bottom we have the footer.

[4] is the list of labels we can provide in which the image label is not available in the lite plan.

We have to click the submit page button [5] after annotating each page individually, after few pages the discovery will automatically give the label to the remaining pages. Click [6] after completing the annotation of the document.

- For further segmentation and making the sub documents, we have to manage the fields. Here we are provided with the option of identifying field to index i.e. what all texts are important for us, as we can see in the below snip, we have turned on only subtitle and text because they are the only 2 labels in which we are interested. On the right side we have the option splitting the document as per choice of our label. We have selected subtitle here. This can vary as per different needs of user.


IBM **Watson Discovery** Cookie Preferences **Instance:** Discovery-q6

Manual / Configure data

Identify fields **Manage fields** **Enrich fields**

Apply changes to collection

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

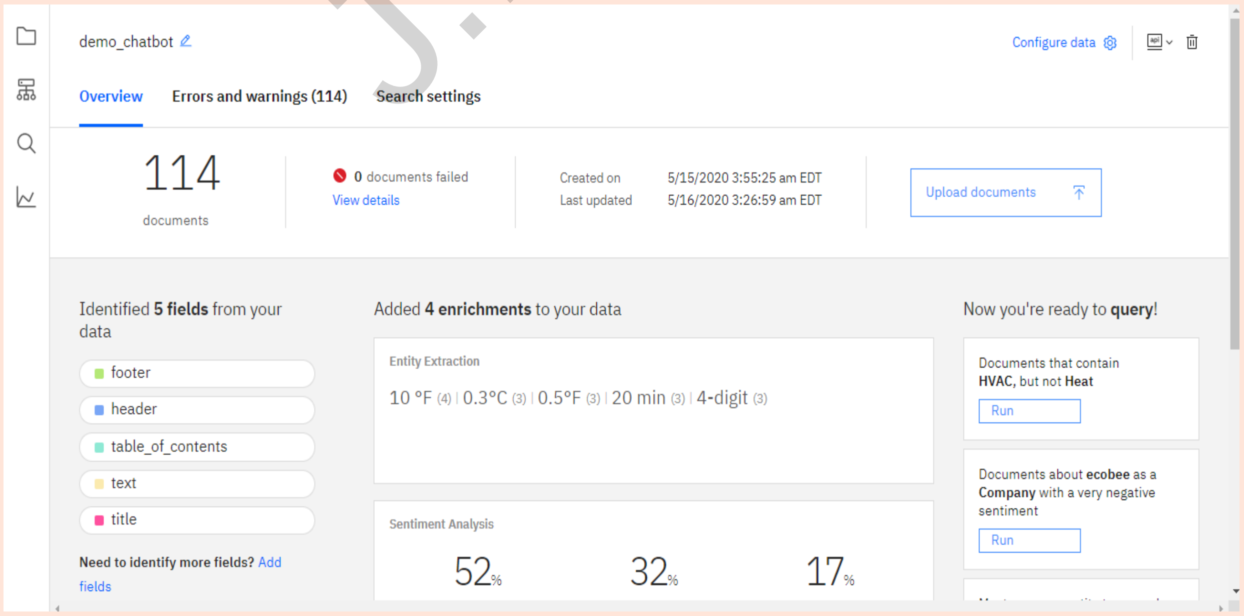
You can split your documents into segments based on fields. Once split, each segment is a separate document that will be enriched, indexed, and returned as a query separately. Learn more.

Split document on each occurrence of

subtitle

After doing everything as mentioned above, we have to click on Apply changes to Collection.

- It will take some time to process and after that we will have multiple documents as shown below, the document we uploaded earlier is segmented in 104 documents as shown below.


demo_chatbot

Configure data

Overview Errors and warnings (114) Search settings

114 documents

0 documents failed
View details

Created on 5/15/2020 3:55:25 am EDT
Last updated 5/16/2020 3:26:59 am EDT

Upload documents

Identified **5 fields** from your data

- footer
- header
- table_of_contents
- text
- title

Need to identify more fields? Add fields

Added **4 enrichments** to your data

Entity Extraction

10 °F (4) | 0.3°C (3) | 0.5°F (3) | 20 min (3) | 4-digit (3)

Sentiment Analysis

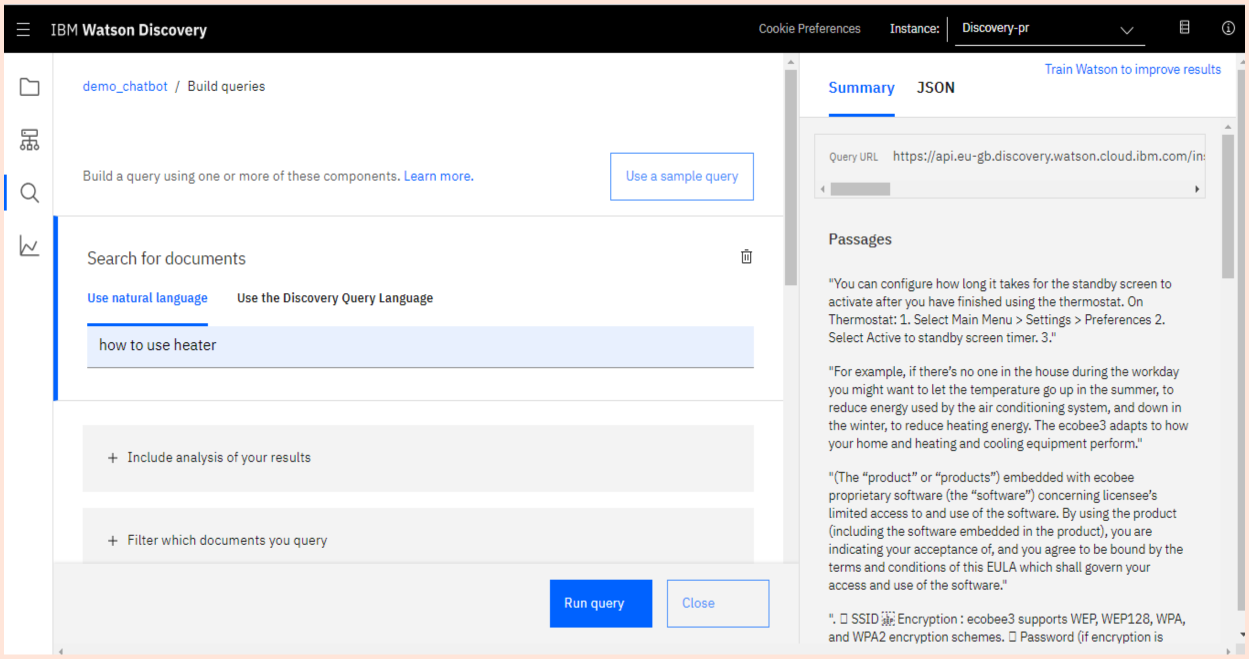
52% 32% 17%

Now you're ready to **query!**

Documents that contain **HVAC,** but not **Heat**
Run

Documents about **ecobee** as a **Company** with a very negative sentiment
Run

- Return to the query panel and try to build the same query as earlier and check the result. We will observe an improved result.


IBM Watson Discovery

Cookie Preferences Instance: Discovery-pr

demo_chatbot / Build queries

Build a query using one or more of these components. Learn more.

Use a sample query

Search for documents

Use natural language | Use the Discovery Query Language

how to use heater

+ Include analysis of your results

+ Filter which documents you query

Run query | Close

Train Watson to improve results

Summary | JSON

Query URL https://api.eu-gb.discovery.watson.cloud.ibm.com/in

Passages

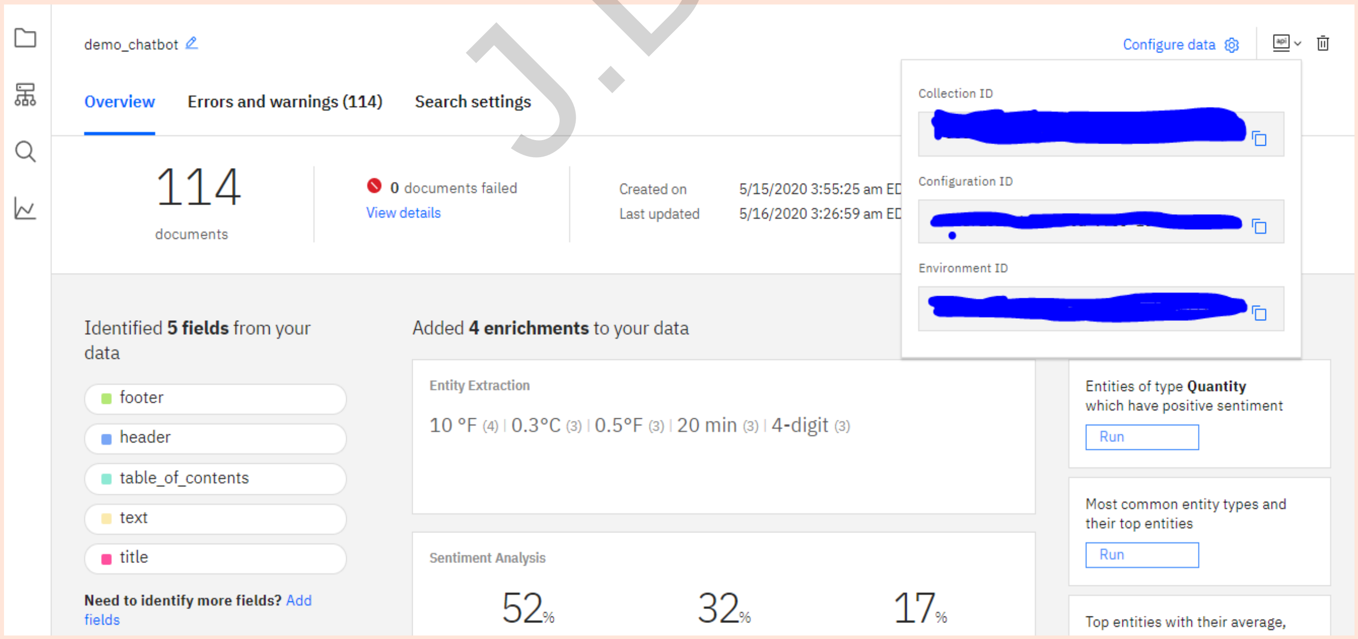
"You can configure how long it takes for the standby screen to activate after you have finished using the thermostat. On Thermostat: 1. Select Main Menu > Settings > Preferences 2. Select Active to standby screen timer. 3."

"For example, if there's no one in the house during the workday you might want to let the temperature go up in the summer, to reduce energy used by the air conditioning system, and down in the winter, to reduce heating energy. The ecobee3 adapts to how your home and heating and cooling equipment perform."

"(The "product" or "products") embedded with ecobee proprietary software (the "software") concerning licensee's limited access to and use of the software. By using the product (including the software embedded in the product), you are indicating your acceptance of, and you agree to be bound by the terms and conditions of this EULA which shall govern your access and use of the software."

". SSID Encryption : ecobee3 supports WEP, WEP128, WPA, and WPA2 encryption schemes. Password (if encryption is

- Next, we have to store the credentials of Discovery which can be viewed as shown below:


demo_chatbot

Configure data

Overview | Errors and warnings (114) | Search settings

114 documents

0 documents failed
View details

Created on 5/15/2020 3:55:25 am ED
Last updated 5/16/2020 3:26:59 am ED

Collection ID [redacted]

Configuration ID [redacted]

Environment ID [redacted]

Identified **5 fields** from your data

- footer
- header
- table_of_contents
- text
- title

Need to identify more fields? Add fields

Added **4 enrichments** to your data

Entity Extraction

10 °F (4) | 0.3°C (3) | 0.5°F (3) | 20 min (3) | 4-digit (3)

Sentiment Analysis

52% 32% 17%

Entities of type **Quantity** which have positive sentiment
Run

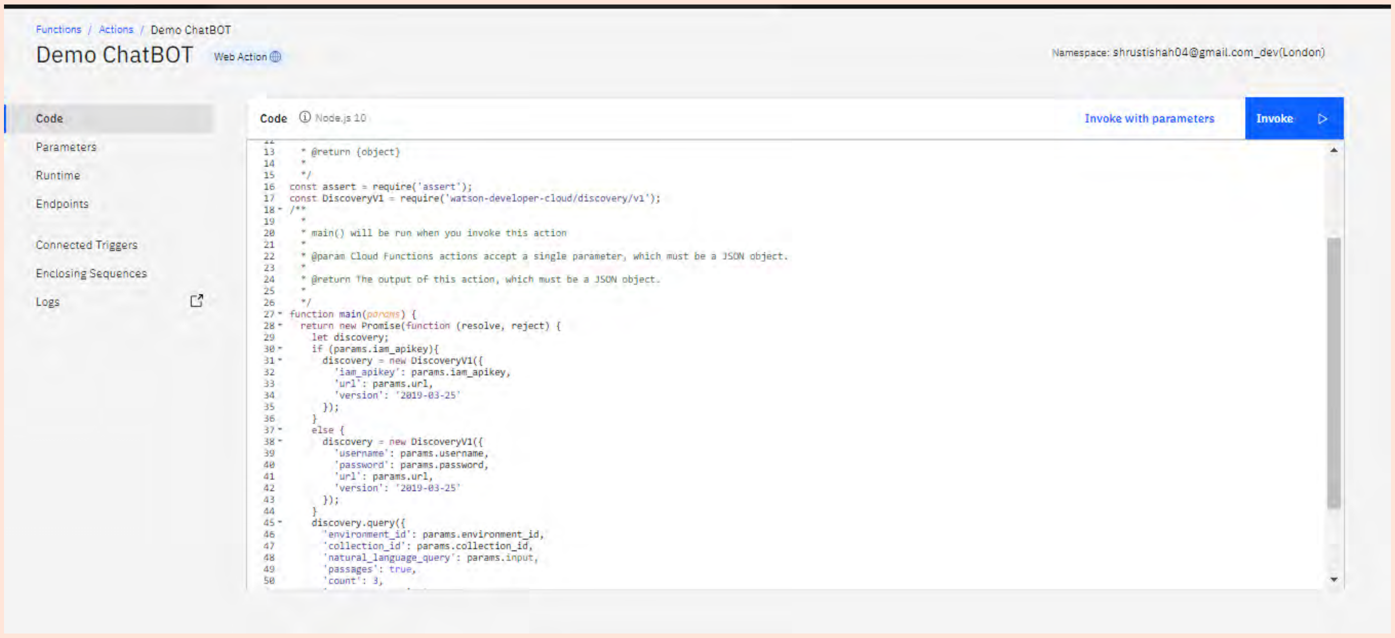
Most common entity types and their top entities
Run

Top entities with their average,

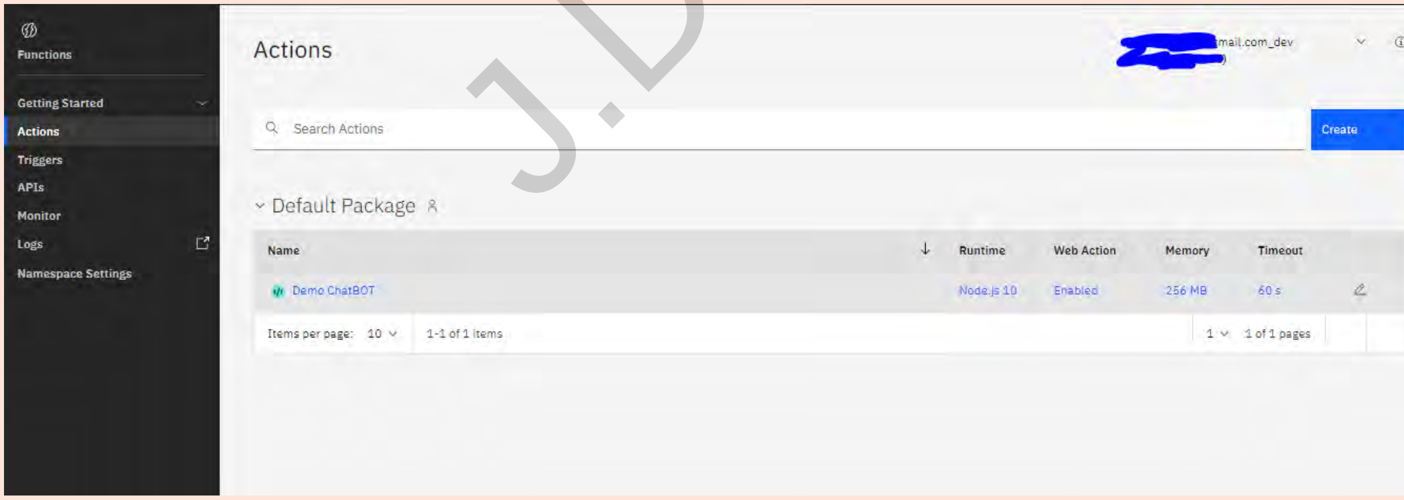
- We already have the API key and the URL.
- Next, we have to create the IBM Cloud Function Action:

  It is used to link the discovery with assistant, so that our queries can be answered by the discovery. After selecting the action from the IBM catalog, we have to click on the action tab as shown on the left menu. Here we made the Information function.

Then we can post the code which will help us to link the discovery.



```
  * @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){
      discovery = new DiscoveryV1({
        'iam_apikey': params.iam_apikey,
        'url': params.url,
        'version': '2019-03-25'
      });
    }
    else {
      discovery = new DiscoveryV1({
        'username': params.username,
        'password': params.password,
        'url': params.url,
        'version': '2019-03-25'
      });
    }
    discovery.query({
      'environment_id': params.environment_id,
      'collection_id': params.collection_id,
      'natural_language_query': params.input,
      'passages': true,
      'count': 3,
```



Name	Runtime	Web Action	Memory	Timeout
Demo ChatBOT	Node.js 10	Enabled	256 MB	60 s

Items per page: 10 | 1-1 of 1 items | 1 | 1 of 1 pages

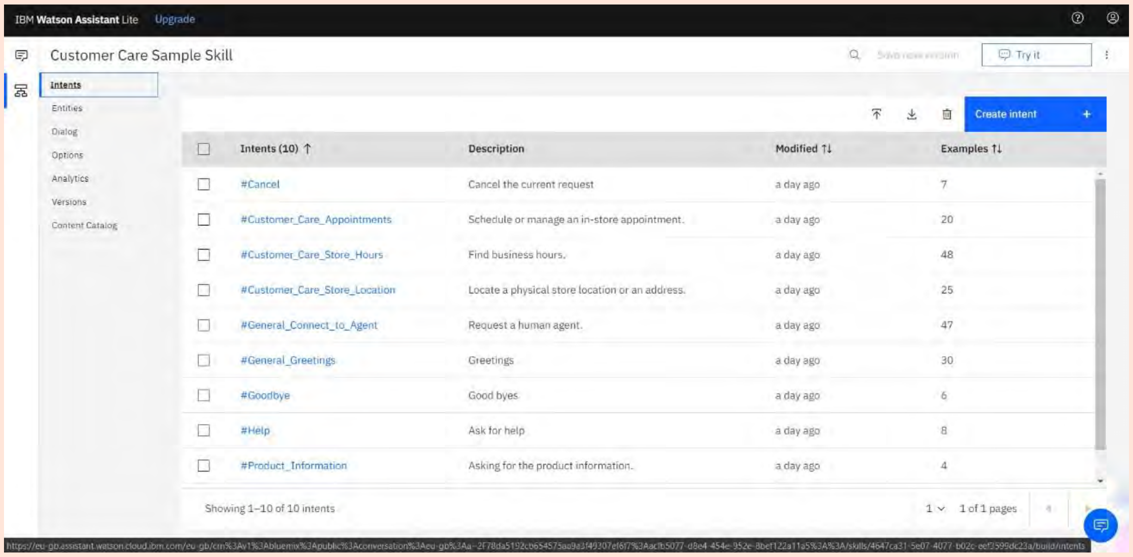
We can make the parameters as per the code and paste the parameter value from the discovery credentials. After that we have to click on the endpoint and enable the web action which will generate a public URL and it will be further used.

- Next, we have to make the Watson assistant and use the sample customer care skill for convenience. We can add intent related to product information and the related entities and dialog flow.

  Intents- These are the categories which we mention or we expect the user input to be, for example: Greetings can be an intent and in it we can have examples as Good Morning, Good Evening and all.

Entities- These are used to mention the usual typos of the user and the synonyms like some people write the good morning as gm, good morning, gud morning, so we can cover all these also instead of returning a message to rephrase.

Dialog- Here we mention the outputs to be given, these can be static as well as dynamic.



Customer Care Sample Skill

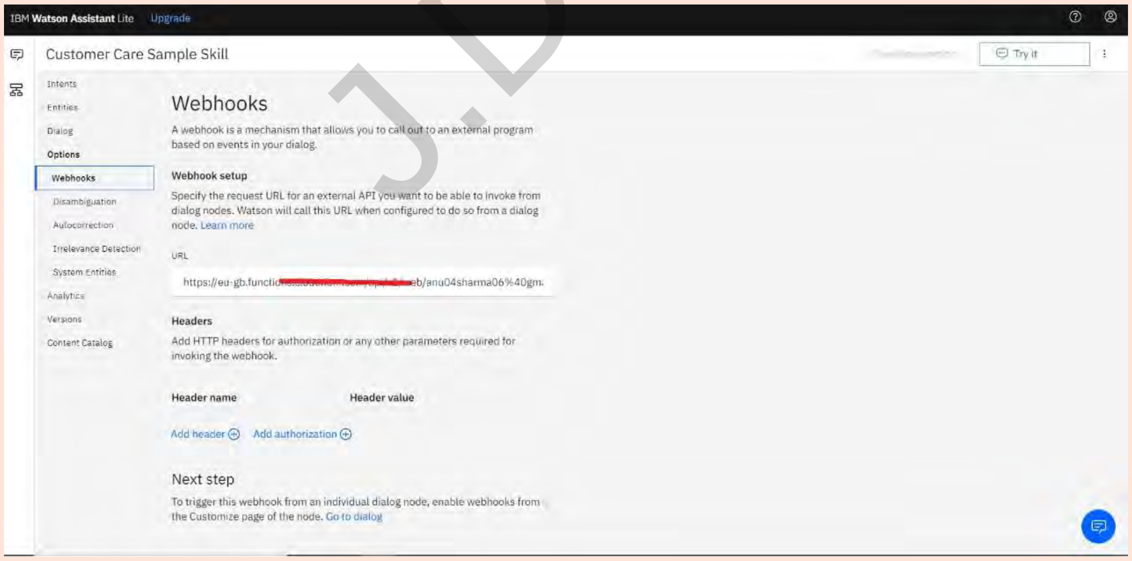
Intents
Entities
Dialog
Options
Analytics
Versions
Content Catalog

	Intents (10)	Description	Modified	Examples
☐	#Cancel	Cancel the current request	a day ago	7
☐	#Customer_Care_Appointments	Schedule or manage an in-store appointment.	a day ago	20
☐	#Customer_Care_Store_Hours	Find business hours.	a day ago	48
☐	#Customer_Care_Store_Location	Locate a physical store location or an address.	a day ago	25
☐	#General_Connect_to_Agent	Request a human agent.	a day ago	47
☐	#General_Greetings	Greetings	a day ago	30
☐	#Goodbye	Good byes	a day ago	6
☐	#Help	Ask for help	a day ago	8
☐	#Product_Information	Asking for the product information.	a day ago	4

Showing 1–10 of 10 intents   1   1 of 1 pages

https://eu-gb.assistant.watson.cloud.ibm.com/eu-gb/crn%3Av1%3Abluemix%3Apublic%3Aconversation%3Aeu-gb%3Aa~2F78da5192cb654575aa9a3f49307ef6f7%3Aacfb5077-d8e4-454e-952e-8bef122a11a5%3A%3A/skills/4647ca31-5e07-4077-b02c-eef3599dc23a/build/intents

- Next, we have to mention the URL that we got earlier.



Customer Care Sample Skill

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.functi[redacted]eb/anu04sharma06%40gm

Headers

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

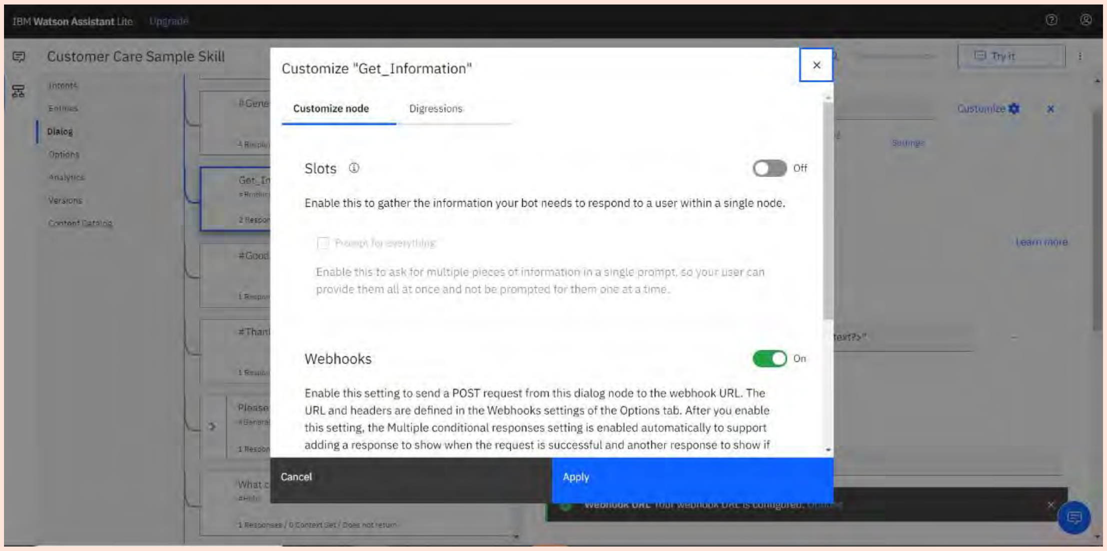
Header name   Header value

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

We have to enable the webhooks which enables our dialog to send a POST request to the webhook URL.



Customize "Get_Information"

Customize node | Digressions

Slots — Off

Enable this to gather the information your bot needs to respond to a user within a single node.

Prompt for everything

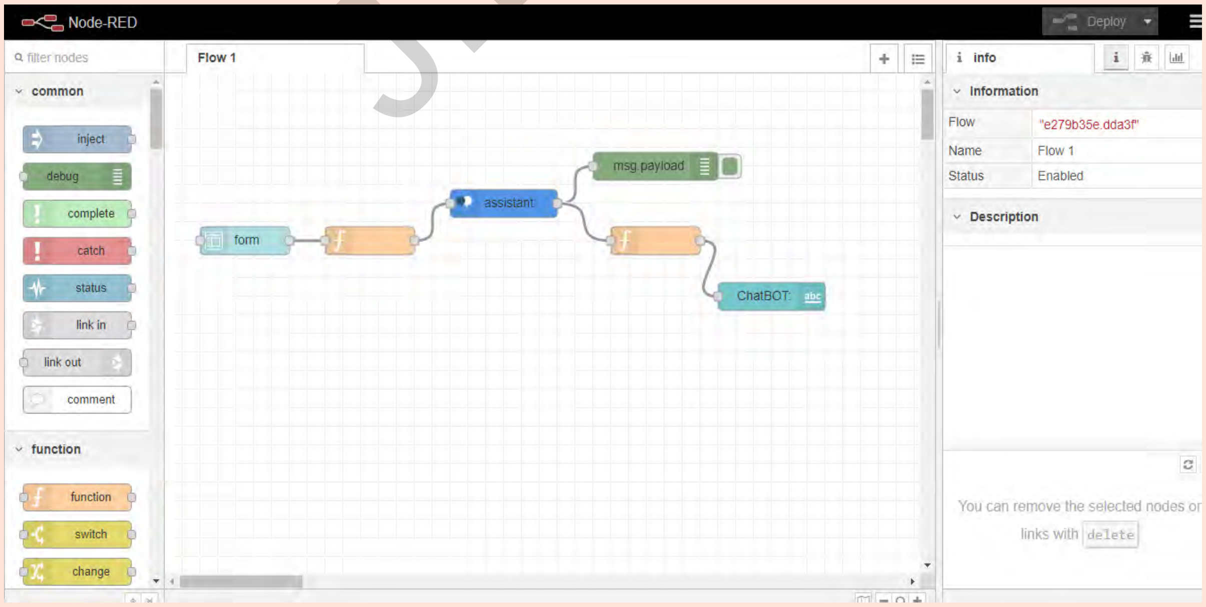
Enable this to ask for multiple pieces of information in a single prompt, so your user can provide them all at once and not be prompted for them one at a time.

Webhooks — On

Enable this setting to send a POST request from this dialog node to the webhook URL. The URL and headers are defined in the Webhooks settings of the Options tab. After you enable this setting, the Multiple conditional responses setting is enabled automatically to support adding a response to show when the request is successful and another response to show if

Cancel | Apply

- After this we have to make the node-red flow, and link everything. We will get a UI from the node. The roles of different nodes can be understood by the references mentioned in in the end. The final flow will look like as shown below.

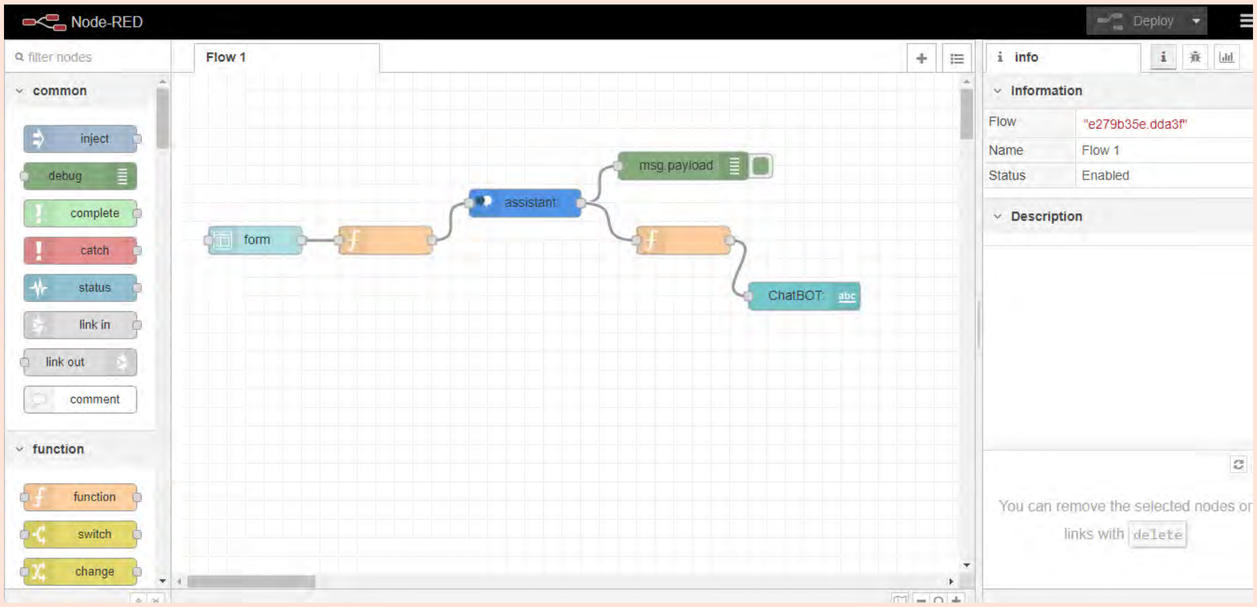


Flow	"e279b35e.dda3f"
Name	Flow 1
Status	Enabled

You can remove the selected nodes or links with `delete`

The UI we have is the basic one but can be improved by writing the HTML code in the template node. We can vary the background colour also from the node-red.
This is how the initial flow looked like but we imported another flow just to improve the UI
This basic flow can be extended to what we have used next, we have imported the flow to make thigs easy and to have an interesting UI.

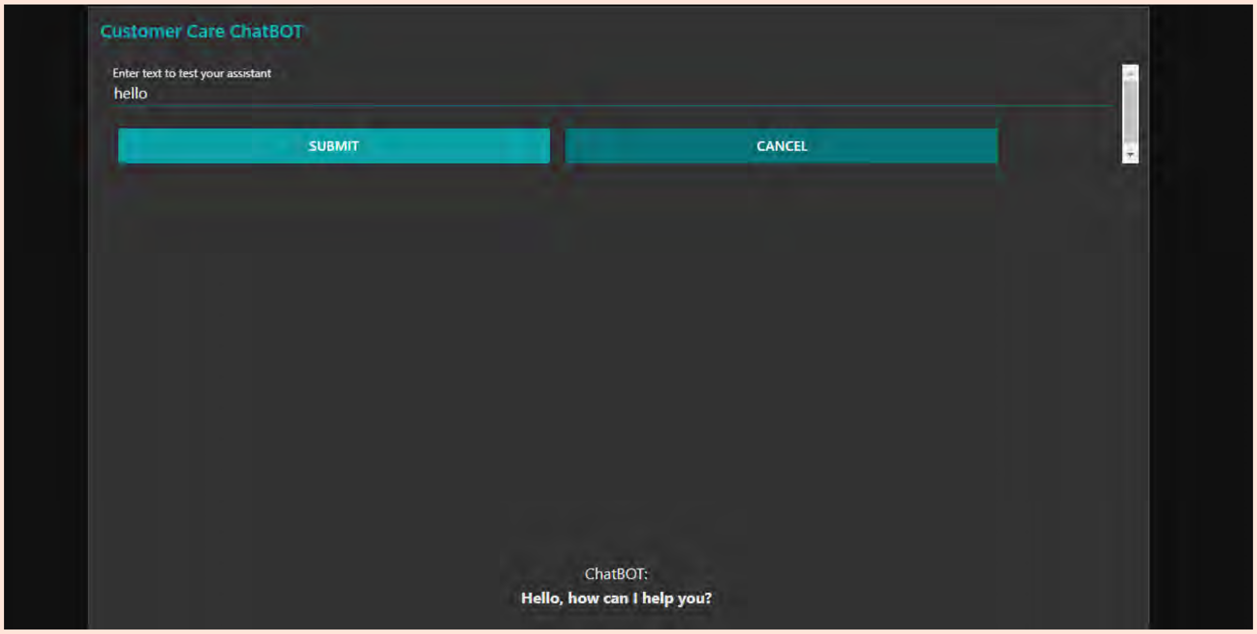
5 Flowchart



Flow	"e279b35e.dda3f"
Name	Flow 1
Status	Enabled

You can remove the selected nodes or links with delete

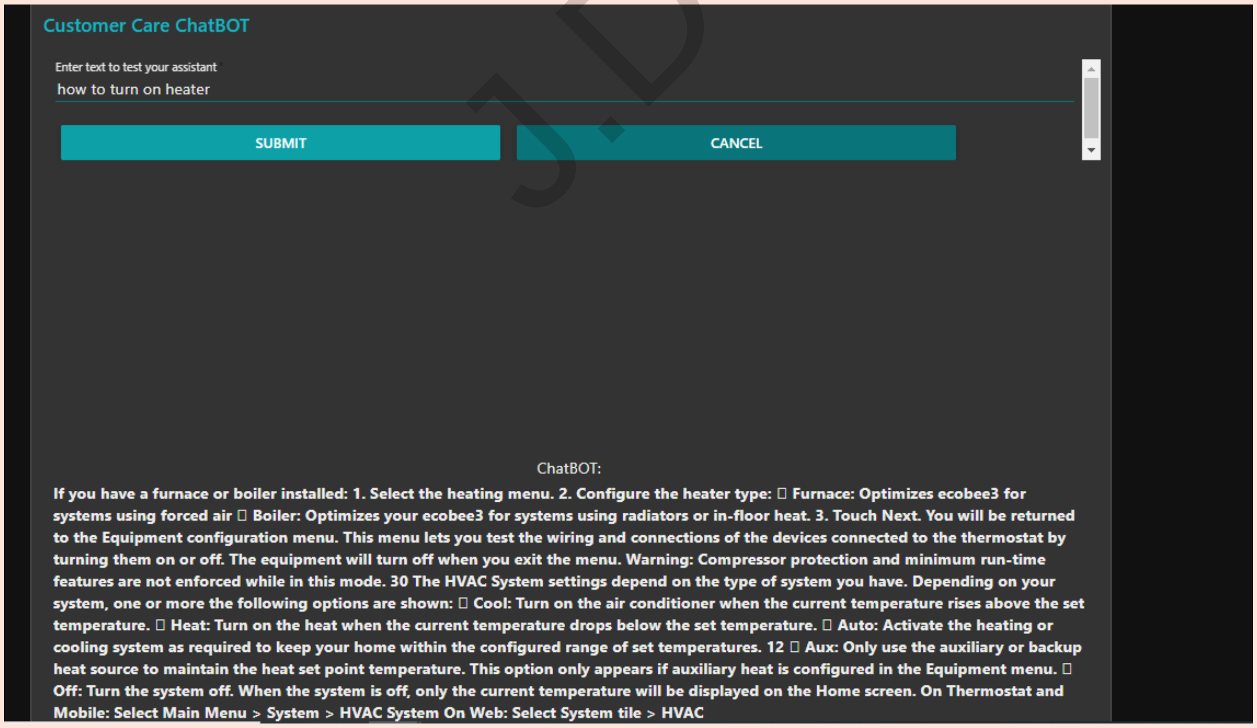
6 Result



hello

ChatBOT:

Hello, how can I help you?



how to turn on heater

ChatBOT:

If you have a furnace or boiler installed: 1. Select the heating menu. 2. Configure the heater type: Furnace: Optimizes ecobee3 for systems using forced air Boiler: Optimizes your ecobee3 for systems using radiators or in-floor heat. 3. Touch Next. You will be returned to the Equipment configuration menu. This menu lets you test the wiring and connections of the devices connected to the thermostat by turning them on or off. The equipment will turn off when you exit the menu. Warning: Compressor protection and minimum run-time features are not enforced while in this mode. 30 The HVAC System settings depend on the type of system you have. Depending on your system, one or more the following options are shown: Cool: Turn on the air conditioner when the current temperature rises above the set temperature. Heat: Turn on the heat when the current temperature drops below the set temperature. Auto: Activate the heating or cooling system as required to keep your home within the configured range of set temperatures. 12 Aux: Only use the auxiliary or backup heat source to maintain the heat set point temperature. This option only appears if auxiliary heat is configured in the Equipment menu. Off: Turn the system off. When the system is off, only the current temperature will be displayed on the Home screen. On Thermostat and Mobile: Select Main Menu > System > HVAC System On Web: Select System tile > HVAC

7 Advantages and Disadvantages

Advantages:

- Companies can use these to decrease the work flow to the representatives.
- Reduce the number of reps.
- Cost Efficient.
- Decrease in the number of calls diverted to representatives.
- Less work load on employees.

Disadvantages:

- Sometimes the chatbot misleads the customers.
- The discovery returns wrong results when not properly configured.
- Giving same answer for different sentiments.
- Sometimes is unable to connect the customer sentiments and intents.

8 Application

- It can be deployed in popular social media applications like Facebook, Slack and Telegram.
- Chatbot can be deployed at any website to clear the basic doubts of the customer.

9 Conclusion

By following the above-mentioned steps, we can create a basic chatbot which can help us to answer the basic questions of the customer or user related to location of the office, working hours and the information about the product. We successfully create the intelligent helpdesk smart chatbot using Watson Assistant, Watson Cloud Function, Watson Discovery and Node-Red.

10 Future Scope

We can import the pre-built node-red flow and can improve our UI, moreover we can make a data base and use it to show the recent chats to the customer. We can also improve the results of discovery by enriching it with more fields and doing the Smart Data Annotation more accurately. We can get the premium version to increase the scope of our chatbot in terms of the calla and requests.

We can also include Watson text to audio and Speech to text services to access the chatbot handsfree. These are few of the future scopes which are possible.

11 Appendix

11.1 Code:

Cloud Function:

```
 /**
  *
  * @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){
    discovery = new DiscoveryV1({
      'iam_apikey': params.iam_apikey,
      'url': params.url,
      'version': '2019-03-25'
    });
  }
  else {
    discovery = new DiscoveryV1({
      'username': params.username,
      'password': params.password,
      'url': params.url,
      'version': '2019-03-25'
    });
  }
  discovery.query({
    'environment_id': params.environment_id,
    'collection_id': params.collection_id,
    'natural_language_query': params.input,
    'passages': true,
    'count': 3,
    'passages_count': 3
  }, function(err, data) {
    if (err) {
      return reject(err);
    }
    return resolve(data);
  });
});
```

11.2 References

- https://www.ibm.com/cloud/architecture/tutorials/cognitive_discovery
- http://www.iotgyan.com/learning-resource/integration-of-watson-assistant-to-node-red
- https://www.ibm.com/cloud/get-started
- https://developer.ibm.com/tutorials/how-to-create-a-node-red-starter-application/
-