Automated Customer Support System Based on Email Analytics

Providing customer support in a timely manner is very important to enhance customer experience. Organizations receive communication, from their customers, through various channels like emails, phone calls, applications etc. Organizations need to understand the intent and content of each of the communication and ask customers for any additional information required to fulfill their requests. Manually processing each email request requires a lot of time to go through the emails, understand them and respond to them in a quick and appropriate manner. Additionally, manual processing is error prone. Delay in acting on customer requests could affect customer satisfaction levels. This code pattern addresses these issues. It will demonstrate Watson's ability to automate email responses that are related to business processes. Customers will benefit from more timely responses and clear instructions for what information is needed in order to fulfill their requests.

In this code pattern we take use cases of a telecom domain customer support who need to action upon email requests they receive from customer. We will consider request scenarios for **enabling a service**, **disabling a service**, **changing plan** and **Adding family member to plan**.

As a customer support, one should

- Know intent of email.

- Know information available in these emails.

- Identify information that is missing.

- Auto-composes responses and sends emails.

Steps to be performed:

1. Deploy custom model, built for telecom use cases using Watson Knowledge Studio, to Watson Natural Language Understanding.

2. Node-RED flow gets (polls) for customer emails and retrieves them.

3. Customers are validated by matching customer email id and customer data available in Cloudant database for customer records.

4. Information (entities like name, phone number) in emails are identified using Watson NLU.

5. Intent (or service request type) of email is identified using Watson NLC.

6. Email content, entities, intent are saved in Cloudant email database for each request.

7. Customer support user accesses nodejs application deployed on IBM cloud.

8. Application fetches emails, entities and intents from Cloudant email database.

9. Responses to email are auto populated (auto composed) and customer support user can just click the send button to send response to customer.