Ontology based Chatbot (For E-commerce Website)

Anusha Vegesna
Information Technology
Dwarkadas J. Sanghvi College of
Engineering
Mumbai, India

Pranjal Jain
Information Technology
Dwarkadas J. Sanghvi College of
Engineering
Mumbai, India

Dhruv Porwal
Information Technology
Dwarkadas J. Sanghvi College of
Engineering
Mumbai, India

ABSTRACT
A working model of Ontology based chatbot is proposed that handles queries from users for an E-commerce website. It is mainly concerned with providing user the total control over the search result on the website. This chatbot helps the user by mapping relationships of the various entities required by the user, thus providing detailed and accurate information there by overcoming the drawbacks of traditional chatbots. The Ontology template is developed using Protégé which stores the knowledge acquired from the website APIs while the dialog manager is handled using Wit.ai. The integration of the dialog manager and the ontology template is managed through Python. The related response to the query will be formatted and returned to the user on the dialog manager.

General Terms
Artificial intelligence and Machine learning

Keywords
Ontology; chatbot; e-commerce; electronics ontology

1. INTRODUCTION
The rapid development of the E-commerce has resulted in the growth of excessive information which has become too vast and rich. This vast information poses a problem for users who wish to have a clear choice of their interest. Choosing from the vast ocean of information becomes the focus of the subsequent. Numerous websites with relevant results are existing but they don't provide the specific information needed directly. Various filtering techniques or options have to be applied to get the exact relevant information. Also, there are a variety of information extraction technologies, but it is difficult to find the "real" information that is required directly. We propose an Ontology based chatbot approach to get the desired appropriate answer to the query. This approach would help us retrieve the exact and direct information required without the effort of searching by the user. Chatbot is a computer program that interacts with users using natural languages [11]. Ontology gives the basic terms of the vocabulary and the relationships, to capture the relevant domain knowledge, and propose a common understanding of the field to identify the common vocabulary, and give a clear formal definition. Because ontology has a structured body, and e-commerce website also has the same format, we build e-commerce ontology first during the information extraction. Ontology-based information extraction is mainly used for the description of the data to achieve the extraction with less dependence on the page structure [12].

2. EXISTING SYSTEM

2.1 CiteSeerX: An E-commerce Website based chatbot [1]
Approach: This chatbot makes it easier for the user to find information on the E-commerce website. The user has an option to chat with the chatbot and ask it normal questions to get responses. The chatbot has pre-programmed responses.

Pitfalls: It has pre-programmed responses since it does not make use of Ontology.

2.2 Eliza [2]: AI Program that simulates the behavior of a therapist.
Approach: A program like Eliza requires knowledge of three domains:

1. Artificial Intelligence

2. Expert System

3. Natural Language Processing

Pitfalls and Comparison:

•The Bot does not learn anything on its own. It has a static database which it uses to respond to any user's query. In the proposed Ontology based Chatbot, it identifies the intent of the query and learns.

•It uses C language, whereas we use Java for creating an ontology template. Since Java is Platform independent it is more beneficial as compared to C.

•There is no NLP processing in this chatbot. For incorporating NLP, we are using a tool wit.ai which shall process the user query in any format asked.

3. PROPOSED SYSTEM
The proposed system is an Ontology based chat-bot which will be mainly based on the E-commerce domain. E-commerce website APIs (EBay website which is freely available) are used as the data source. Ontology template is built using PROTEGE platform that retrieves data from the data source (using Jape rules). Once the data is completely retrieved in the template then the user can ask questions to the bot and as per the questions the intent and other properties will be arranged using NLP in Wit.ai. Using the intent and other properties arranged, a function call will be made using Python language just like an API call to the Ontology template and the query will be answered by the chatbot from the retrieved data in the Ontology template.
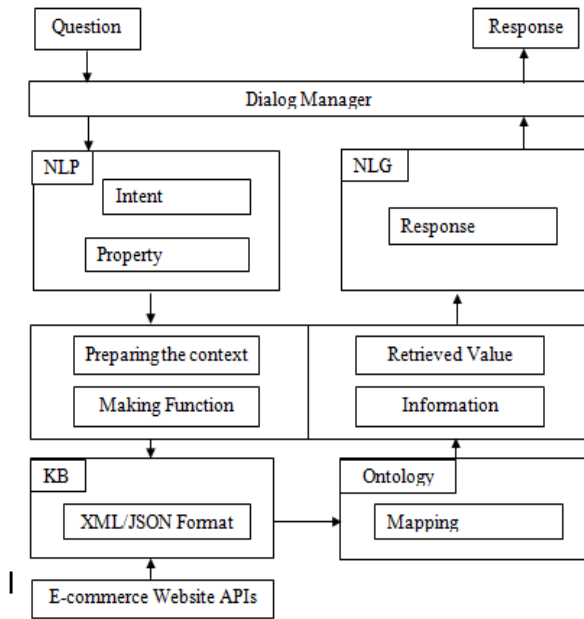
Fig 1: Architecture diagram

4. MODULAR DESCRIPTION

With the proposed architecture aforementioned, we have divided the whole implementation into three modules.

4.1 Knowledge base (KB)

The approach relies on the use of exiting tools, resources and information (e.g. FAQ, API, system logs) in order to create a KB[6]. For this very reason, we use APIs of ecommerce domains that are freely available.

- API are annotation for application programming interface which are a set of functions and procedures that allow the creation of applications which access the features or data of an operating system, application, or other service.
- API calls are used retrieve data either in XML, JSON or Excel format. This format forms the basic blocks to create the ontology template.
- API [3] calls are made using a platform called 'Datacol'. It takes in the links of the data selected and creates a spreadsheet of all the parameters selected.
- Excel sheet forms the basis for creating the ontology template which forms the second module.

Fig 2: Extracted data in KB

4.2 Ontology Template

Ontology gives the basic terms of the vocabulary and the relationships, to capture the relevant domain knowledge, and

propose a common understanding of the field to identify the common vocabulary, and give a clear formal definition.[7]

- Ontology follows java object oriented approach, such as inheritance to avoid redundancy that prevails in the existing systems.
- It uses Data from API calls to create classes, properties and named individual.
- Protégé [4], platform to create ontology is used in the project to portray different classes and properties.
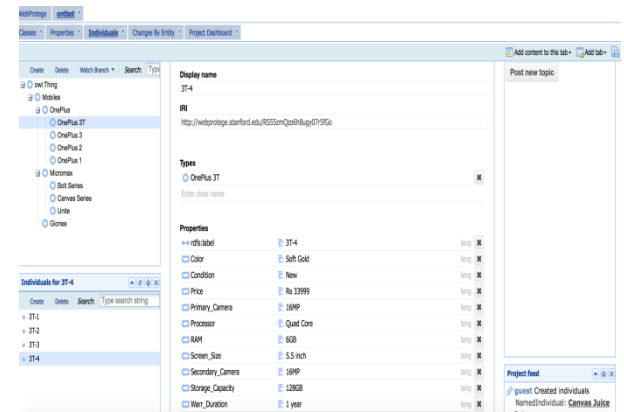

Fig 3: Mapping of dependencies on the ontology template

4.3 Dialog Manager

For Api.ai [5] provides functionalities to enable a conversation with the user.

- The dialog manger provided by api.ai will help us obtain the intents of the query posted by the user using NLP.
- Entities are created mentioning various products, synonyms for the product, and properties of the various products.
- With the intents and entities recognized by the platform in user's question, it generates a JSON code.
- This JSON code is used to obtain data from the protégé template and is then forwarded to messaging bots which forms the interface for the project.
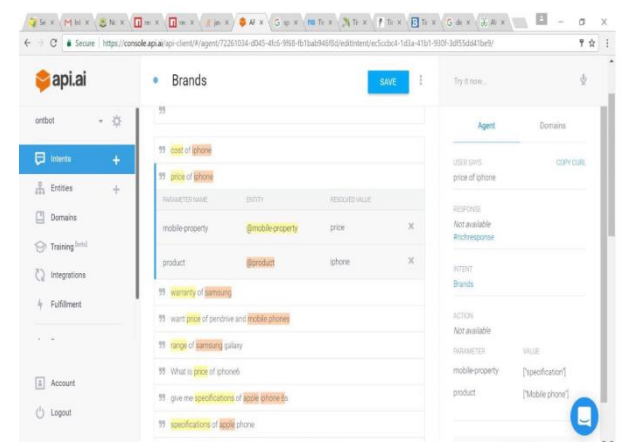

Fig 4: Creation of intents and entities

5. IMPLEMENTATION

5.1 Setting up the intents and entities in API.AI

API.AI is a platform that understands natural language which is easy for developers and non-developers to design and integrate intelligent and sophisticated conversational user

interfaces into mobile apps, web applications, devices, and bots. The goal of this approach is to make the process of creating and integrating sophisticated conversational user interfaces as simple as possible. Entities represent concepts and serve as a powerful tool for extracting parameter values from natural language inputs. The entities that are used in a particular agent will depend on the parameter values that are expected to be returned as a result of agent functioning. In other words, a developer need not create entities for every concept mentioned in the agent – only for those required for actionable data. Intent represents a mapping between what a user says and what action should be taken by software. Intent interfaces have the following sections:

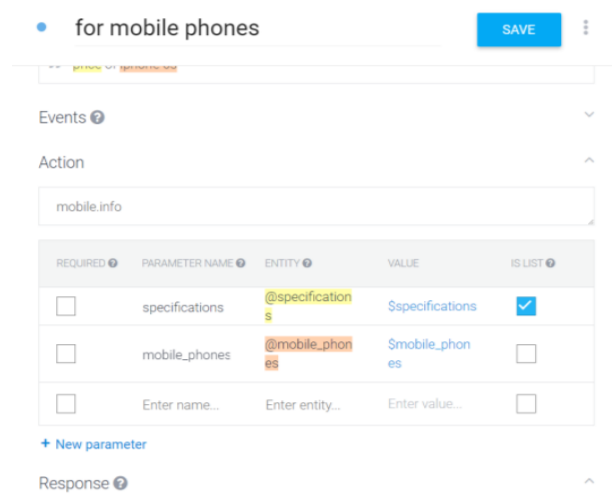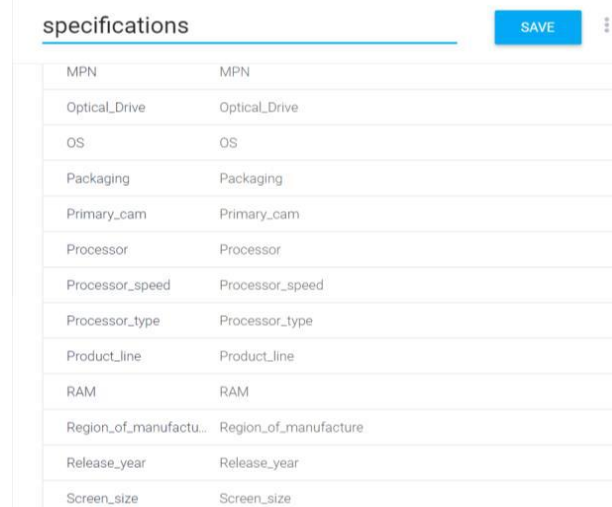• User says

• Action

• Response

• Contexts [8]

Fig 5: Setting intents and entities

5.2 Extracting data from EBay website

The data that is required as an input to the API.AI is extracted using a platform called 'Octoparse'. This Software is a cloud-based web crawler that helps you easily extract any web data without coding in real time. It selects data to be extracted using X-paths and collects data with accuracy. The software crawls from all sorts of data loading techniques. There is a

fully-fledged built-in browser to load all the data from different sources. It will then convert the data into structured tables. The file created can be exported in either Excel sheet or maybe a .csv file and all the scrapped data is stored in the file for further use.
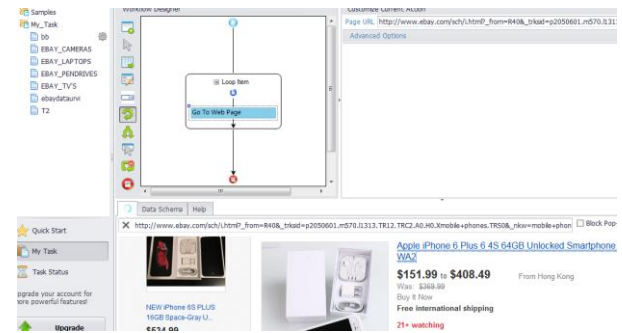

Fig 6: Extracting information from the website

Fig 7: Extracted data in excel sheet

5.3 Converting the data to JSON format:

After extracting the data from website in CSV format, we need to convert them into JSON format. API.AI accepts inputs in JSON format and also gives outputs in JSON format. Hence, data in JSON format is the best way to retrieve results based on user query.

Fig 8: Code for the conversion

5.4 Upload the data on an online database:

The data in JSON Format has to be uploaded in an online database system, which can be accessed to retrieve results based on parameter and intents. We use MongoDB in this case. MongoDB [9] is a free and open-source cross-platform document-oriented database program. It is classified as a NoSQL database program, MongoDB uses JSON-like documents with schemas. mLab's MongoDB hosting platform is the fastest growing cloud Database-as-a-Service in the world. Get started with a free database and expert support.

mLab is a fully managed cloud database service that hosts MongoDB databases. mLab runs on cloud providers Amazon, Google, and Microsoft Azure, and has partnered with platform-as-a- service providers. We upload the JSON files as collections under a mlab database
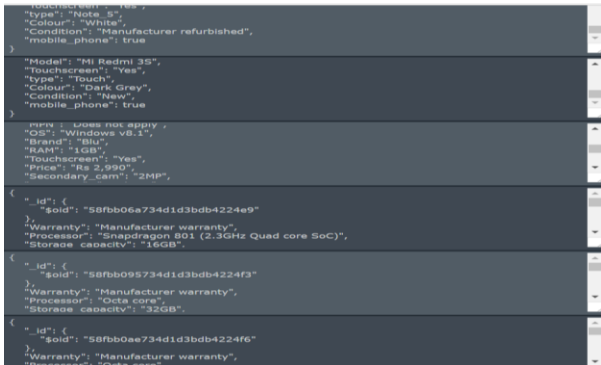

Fig 9: Online database

5.5 Setting up a Facebook Messenger App

A Facebook Chat Messenger application has to be developed as the user interface or the chat module where user can ask their queries. This can be done on developers.facebook.com, which the Facebook developer platform to build and deploy apps on Facebook. After developing the app, it has to be sent to Facebook review authorities which test the app thereafter giving permissions to deploy the app for Facebook users.
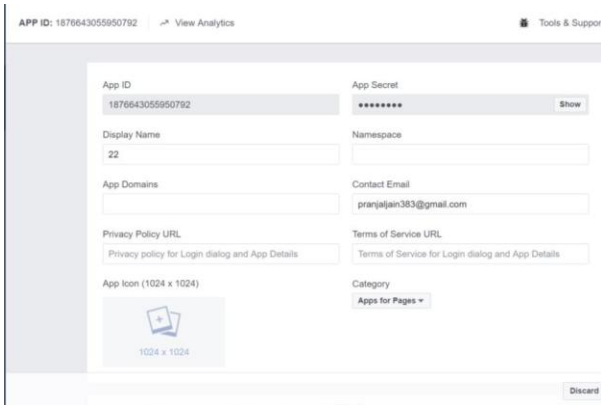

Fig 10: FB messenger App

5.6 Setting up Webhook for API.AI and Facebook Messenger

Webhook integration allows you to pass information from a matched intent into a web service and get a result from it. A WebHook is an HTTP callback: an HTTP POST that occurs when something happens; a simple event-notification via HTTP POST [10]. A web application implementing WebHooks will POST a message to a URL when certain things happen. IBM Bluemix is a platform-as-a-service environment developed by IBM. It supports several programming languages and services as well as the DevOps development methodology in an integrated way to create, execute, deploy and manage applications in the cloud. We use IBM Bluemix's Node-red services to develop set up a webhook which uses Facebook's access token to connect to api.ai's agent. It accepts queries from Facebook users. It converts these queries into messages which it sends to api.ai agent. The agent parses the queries and picks up the intents from the query. It does this using In-built Natural Language

processing algorithms. The agent then responds with parameters based on user intents and entities which are set.
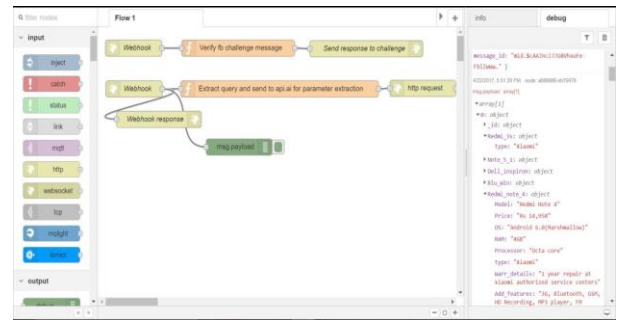
Fig 11: Web hook setup

5.7 Using parameters acquired from API.AI to retrieve answers from online database

IBM Bluemix is a platform-as-a-service environment developed by IBM. It supports several programming languages and services 1 as well as the DevOps development methodology in an integrated way to create, execute, deploy and manage applications in the cloud. We use IBM Bluemix's Node-red services to develop set up a webhook which sends the parameters obtained from api.ai's agent. This parameters can be used to select the appropriate collection from online MongoDB database. The required answers are retrieved using the parameters and sent to API.AI.

Fig 12: Code for the data flow

5.8 Sending the answer back to the user

Natural Language Generation is applied on the answers retrieved from the online databases. The answer generated is then sent back to the user using the same webhook between api.ai and face book.

Fig 13: Screenshot from the FB messenger
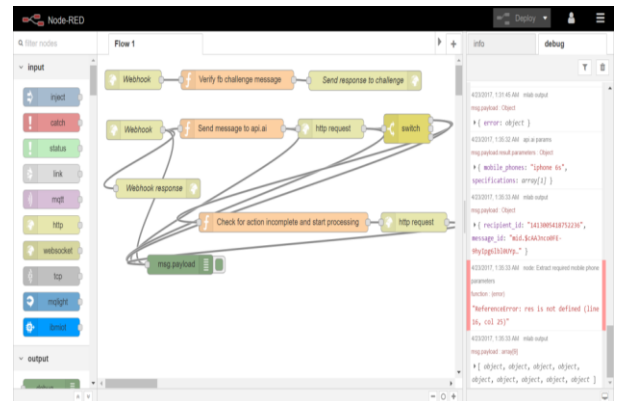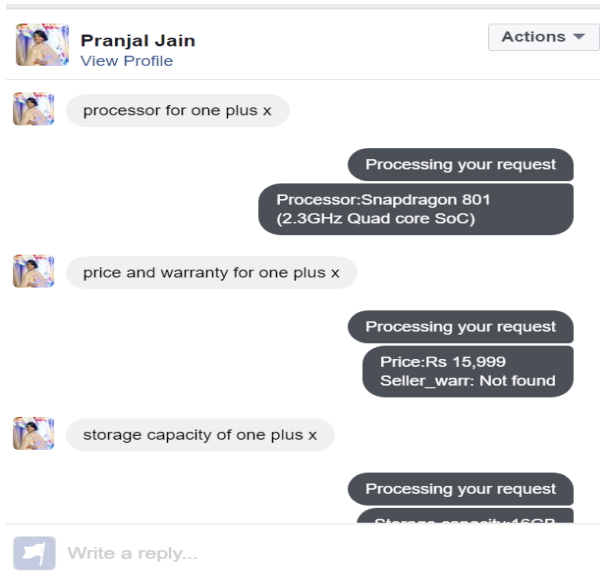
6. CONCLUSION

Through the proposed project and research, the conclusion drawn is that the new proposed system will help the users obtain information by overcoming major flaws in the existing chat bots which could not understand the relationship between entities and properties. The proposed system successfully maps relationships and retrieves data. Using this study we can build the chat bot further to help the users to understand the details of products without logging in on the website, in this case the users who have queries and need information directly to be acquired successfully. This project will alleviate any pitfalls the user comes across during intensive search on the website and ameliorates their efficiency. The user after obtaining information, can then log in on the website to purchase any product. Since this project is successfully deployed on Facebook we have firsthand experience on how accurately the bot responds with the data that is parsed through Protege and saved in the mongo DB database. With sufficient training of the ontology bot we can now successfully interact with ontology bot thus calling it the Ontbot. As a future work, the proposed system can be enhanced by improving the entity recognition with the use of efficient algorithms (like KNN, Similarity) in order to diversify the given answers.

7. REFERENCES

[1] An E-Commerce Website based Chatbot - Siddharth Gupta, Deep Borkar, Chevelyn De Mello, Saurabh Patil International Journal of Computer Science and Information Technologies.

[2] "ELIZA - Wikipedia, the free encyclopedia.," [Online]. Available: http://en.wikipedia.org/wiki/ELIZA.

[3] eBay Developers program, Available: http://developer.ebay.com/Devzone/shopping/docs/CallRef/FindProducts.html#Samples.

[4] Ontology Development 101: A Guide to Creating Your First Ontology- Natalya F. Noy and Deborah L. McGuinness, Stanford University.

[5] http://www.girliemac.com/blog/2017/01/06/facebook-apiai-bot-nodejs/

[6] Agnese Augello, Giovanni Pilato, Alberto Machi, Alberto Machi, An Approach to Enhance Chatbot Semantic Power and Maintainability: Experiences within the FRASI Project, Semantic Computing (ICSC), 2012 IEEE Sixth International Conference.

[7] Yutao Ren, Lihong Jiang, Fenglin Bu, Hongming Cai, Rules and Implementation for Generating Ontology from Relational Database,2012 Second International Conference on Cloud and Green Computing.

[8] Dialogflow - Wikipedia, the free encyclopedia."[Online]. Available: https://en.wikipedia.org/wiki/Dialogflow.

[9] MongoDB - Wikipedia, the free encyclopedia."[Online]. Available: https://en.wikipedia.org/wiki/MongoDB.

[10] T. Davis, "What is a WebHook?," Web Hooks. [Online]. Available:https://webhooks.pbworks.com/w/page/13385124/FrontPage.

[11] Abu Shawar, B. and Atwell, E., (2007). Chatbots: Are They Really Useful? in Proceedings of LDV-Forum 2007 Band 22 (1) pp.31-50.

[12] OntBot: Ontology based chatbot Hadeel Al-Zubaide; Ayman A. Issa International Symposium on Innovations in Information and Communications Technology.