

NATIONAL INSTITUTE OF TECHNOLOGY,ROURKELA



“Odisha Tourism Guide Chatbot”

Under The Supervision of
Prof. Pabitra Mohan Khilar
(Dept. of CSE)

Submitted By:

Raj Kapoor Gupta(115cs0016)

ODISHA TOURISM GUIDE CHATBOT

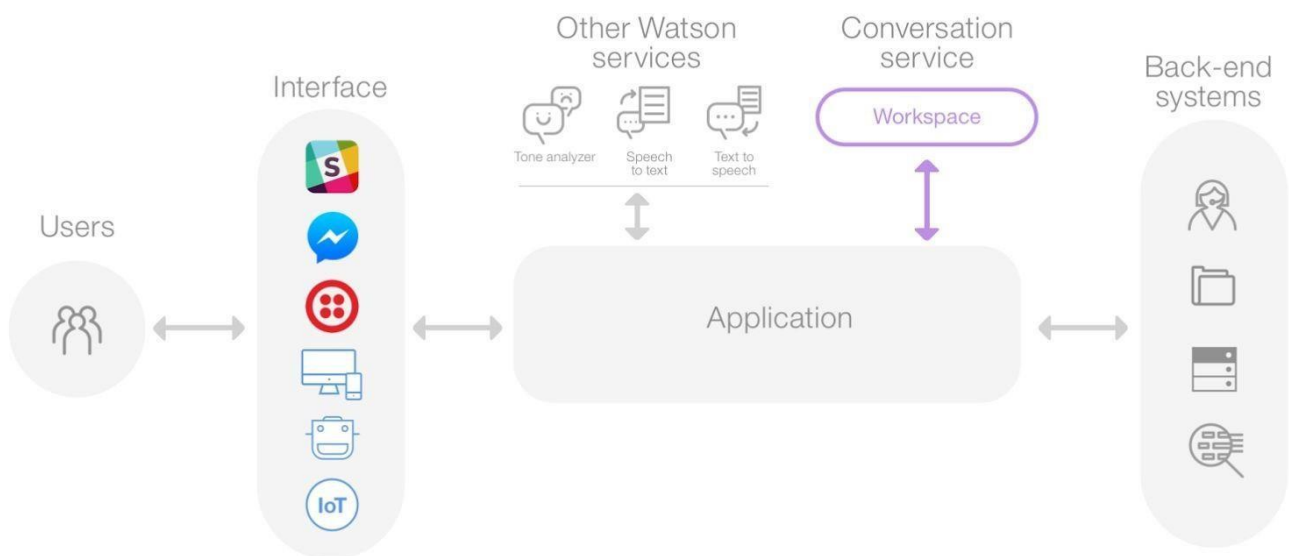
Abstract:

The dynamic topographical and vegetational strength of Odisha have combined to extract some of the most stunning destinations which not only offers a visual feast for the inquisitive visitors but also highly educative and culturally pleasing insight into one of the oldest civilizations of the world. However, it becomes very difficult to pick a destination for the purpose of travel owing to different requirements of the individual. We propose to create a travel app chatbot which will ask the individual's preferences, monetary constraints and number of days of visit and give a suitable suggestion based upon the training data we give it. We have collected the training data from various travel agencies nearby where we have gained a perspective of the type of questions one may ask and the ideal choices we should provide them with.

Key Words:

Android, Intent, Interface, Back-end system, Chatbot, Machine learning

Introduction:



Users interact with our application through the user **interface** that we implemented. For example, a simple chat window or a mobile app, or even a robot with a voice interfaces.

The **application** sends the user input to the Conversation service. The application connects to a *workspace*, which is a container for your dialog flow and training data. The service interprets the user input, directs the flow of the conversation and gathers information that it needs.

The application can interact with our **back-end systems** based on the user's intent and additional information. For example, answer question, open tickets, update account information, or place orders. There is no limit to what you can do.

Related Works:

There are different classes of Chatbots in the industry. The simplest kind are the purposeless mimicry agents which only provide the illusion of a conversation. These are based on deep learning sequence-to-sequence models. The second class comprises intention-based agents such as Apple's Siri. These agents have a simple understanding and can do real stuff. The Chatbot implemented in this project is an intention-based agent.

Cognitive Search Vs. Traditional Search :

Traditional search uses Boolean queries, text relevance and other methods to get hits from search queries whereas in cognitive search we use machine learning techniques to do the same techniques. In cognitive search we train the engine on a few example queries and the engine learns from those queries. The engine gets better with every search as it learns from every query and gives better results. Cognitive search uses Natural Language Processing and machine learning to understand and organize data, predict the intent of the search query, improve relevance of data which eliminates the hassles of keyword based traditional search.

Motivation And Objectives:

The project focuses on the development of the mobile travel guide application by applying software engineering principles and software development methods based on an existing one. Mobile tourism is an emerging trend in the field of tourism involves the use of mobile devices as electronic guides. Users are able to get enriched travel information on the Internet; however, there is a limitation to get information timely when they are on move. In the process of development, a use case diagram and structure diagrams were described. In this research, the existing condition, design idea and implementation methods of application which were built on Android environment were described. Furthermore, the analysis and design of the application were based on the actual using situation.

The main objective of this research is to develop a mobile travel guide application with added functions to an existing application. We decided to design this application because a lot of people think there are similar products already exist on the market. However, after we conducted the market research, there was only one single function on most of travel guide applications, and full-featured products were not listed. Thus, the purpose of designing this product is to make a travel guide application which contains possible integration of a number of features. Therefore, users may use a more convenient application.

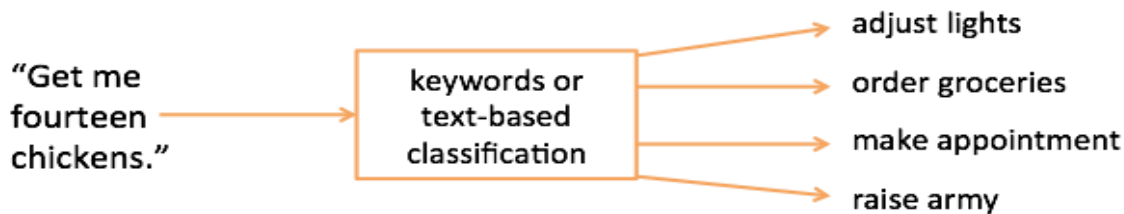
Mechanism:

Intention-based agents understand language as commands and they use that understanding to perform tasks in the world. Understanding what we say as command language requires solving two problems:

1. Identifying what the user wants the machine to do (the “intent”).
2. Figuring out the details of the intent so the machine can take action.

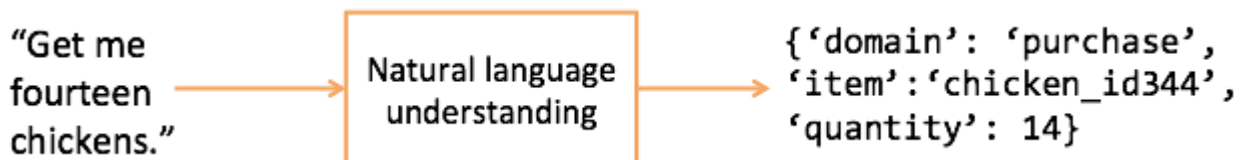
In our project we used sample “intents” - what the user is asking for – and used sample “dialogues” - the response that the bot should give – and trained the Watson Conversation api on these sample responses. Once the model is trained it can handle a variety of user inputs and not just the ones that are used in the sample intents.

We can explain intent determination using the following example.



When the user asks about chickens, the machine has to figure out which of its four capabilities matches the intent of the person. The assistant can determine the intent using either keywords or text-based classification. To use keywords, you simply associate words and phrases with intents. To do text-based classification, you label a bunch of statements with the correct intents and then train a classifier over them. The model is trained using convolutional neural network.

Once the agent has determined the intent, in our example it is to order groceries, it needs to convert the statement details into a machine-readable form, such as a Python dictionary.



Having the command in this kind of dictionary form is called a frame and slot semantics, where the frame is the topic and the slots are the individual features and values. Once we have the machine readable form, the computer can do with it what it wants, just like any other instruction. This is natural language understanding.

Natural Language Understanding is done using context-free grammars (compositional semantics). Context-free grammars consist of production rules that have a single nonterminal on the left and a string on the right side that can consist of terminals and nonterminals. To each of these production rules, we add a semantic attachment. An example set of rules is shown below.

```

$Order → $Purchase $ItemAmount, dunion(s[0],s[1])
$Purchase → is_purchase(tokens), {'domain': 'purchase'}
$ItemAmount → $Amount $Item, dunion(s[0],s[1])
$Amount → is_number(tokens), {'amount': get_number(tokens)}
$Item → is_item(tokens), {'item': get_item(tokens)}
  
```

Nonterminals begin with „\$“. The first rule says that an order consists of an intent to purchase and an item and amount of that item. The semantics of that rule are combined via a dictionary union of the semantics of \$Purchase (s[0]) and the semantics of \$ItemAmount (s[1]). The next rule says that an intent to purchase is determined by the function is_purchase, as shown below. (This, of course,

is a compact way to describe a set of rules. I.e., one with “get me” on the right-hand side, and another with “buy”, and another with “grab me some.”)

The next rule says that \$ItemAmount consists of an \$Amount and an \$Item, which are defined in the following two rules. Each of those rules has an associated semantics at the end. We use parsing to apply those production rules to text to get semantics. Parsing is done using bottom-up dynamic programming. Dynamic programming consists of solving sub problems and then reusing those solutions over and over again. The parsing process builds up the parse by looking at pairs of words, and for each pair, it loops over all n possible words in between. Because the parsing algorithm deals with pairs of sub-parses, the production rules above each have either a single terminal or two non-terminals on the right-hand side.

LIBRARIES AND API'S USED:

- *WATSON CONVERSATION API'S:* With the IBM Watson™ Conversation service, you can build a solution that understands natural-language input and uses machine learning to respond to customers in a way that simulates a conversation between humans.
- *ANDROID STUDIO AND SDK TOOLS:* Android Studio provides the fastest tools for building apps on every type of Android device.

DATA SOURCE USED:

The data source used for displaying the description, mode of transport and hotels to a place have been scrapped from wikitravel.org using JSoup, i.e Java Based Web Scraping Tool. This ensures that as people visit the website and update their reviews of the place on the web page the data will be automatically updated in our application. Thus in this manner , the dynamic nature of the dataset used is beneficial to the working of the application.



Result And Discussion:

The interfacing has been done in the form of an app using java in android studio.

Front End

For the UI/UX design of the app, three xml files have been added:

activity_main: Contains the List View to display all the chat bubbles and the enter text window at the bottom of the screen in which the user enters his/her query. item_mine_message: Contains the chat bubble of the user's input. The himanshusoni package of chat bubbles have been used for this alongside with a message icon as an image view.

item_other_message: Contains the chat bubble of the bot output. The himanshusoni package of chat bubbles have been used for this alongside with a bot icon drawn in vector graphics as an image view.

Back End

The back end has three files namely:

ChatMessage.java: This contains the getter and setter method of the chat bubbles.

ChatMessageAdapter.java: This is an adapter which links the chat bubbles to the List View of the UI design. The class checks whether the message is an user input or a bot generated output and depending on the result gives the proper of orientation of display.

MainActivity.java: This is the class where the main activity starts. A conversation Service has been created using the user id and password in the bluemix account. This conversation service is then given an input by the user in the form of a message request. The response to this request is then displayed using the OnResponse() method. The context of the response is stored in a list and this context is then fed into further requests so that Watson does not have to start the conversation from start. The conversation is initialized with a blank input and a null context.

The build grade of the app has used the com.ibm.watson.developer_cloud:java-sdk:3.7.2 package for the integration.

The apk file can be found here:

<https://drive.google.com/open?id=18UbAkSSyAJYMcPGYxAqVY8oOYHs8sM8>

WORKING MODEL:

Some screenshots of the app are shared below for reference: The wikitravel.org page is being scraped.

Bhitarkanika is a national park in [India's Odisha](#) state (formerly Orissa).

If you are looking for a place off the beaten track for your next holiday, Bhitarkanika could well be your destination. For sea, sand and adventure. Yes, adventure. Because the Bhitarkanika group of islands in northeastern Odisha offers not just great beaches but exciting trekking trails through forests teeming with wildlife and some thrilling boat rides.

And the islands being in Odisha, you could package your holiday with visits to the state's famous ancient temples at the great heritage centres of Puri, Bhubaneswar and Konarak. (They are always worth a second visit if you have done them before.)

The Bhitarkanika islands can be reached from Bhadrak on the Calcutta-Madras railway line that also passes through Bhubaneswar and Cuttack. A night train from Calcutta's Howrah station will reach Bhadrak (297 km) at dawn. (Alternatively, you could take a train in the opposite direction from Bhubaneswar).

✓ Understand



Odisha Tourism Chatbot

Hi

Hello. What is your name?

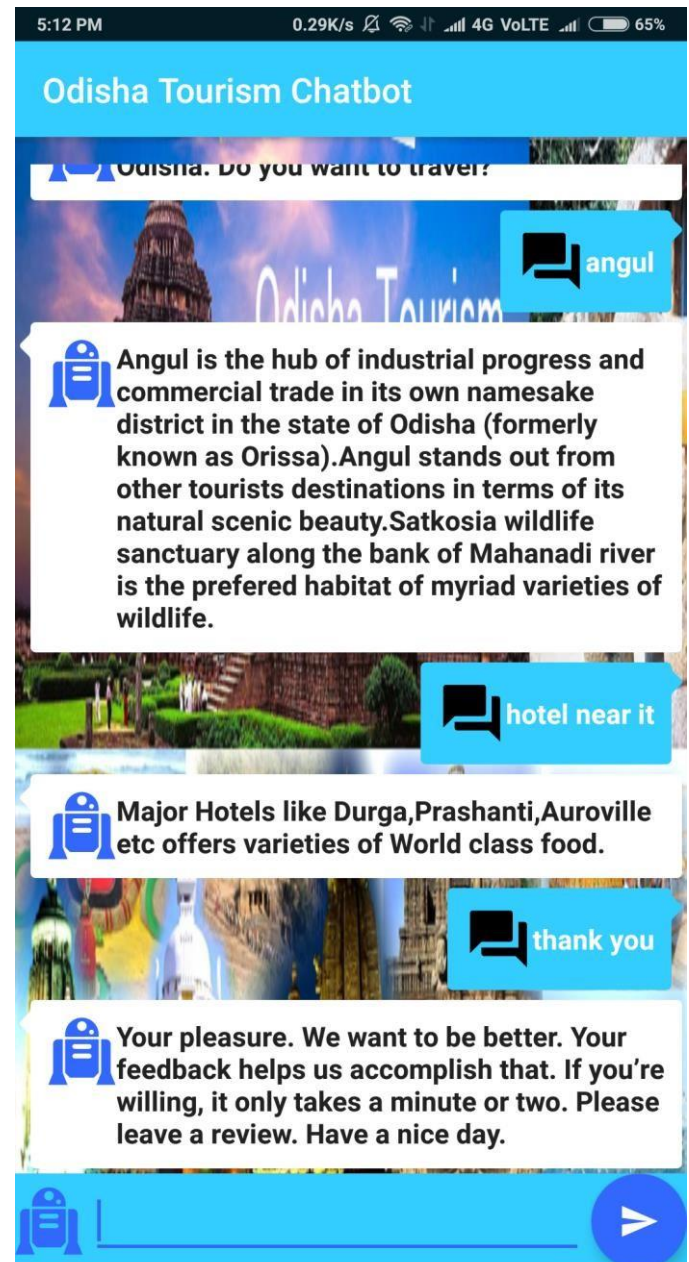
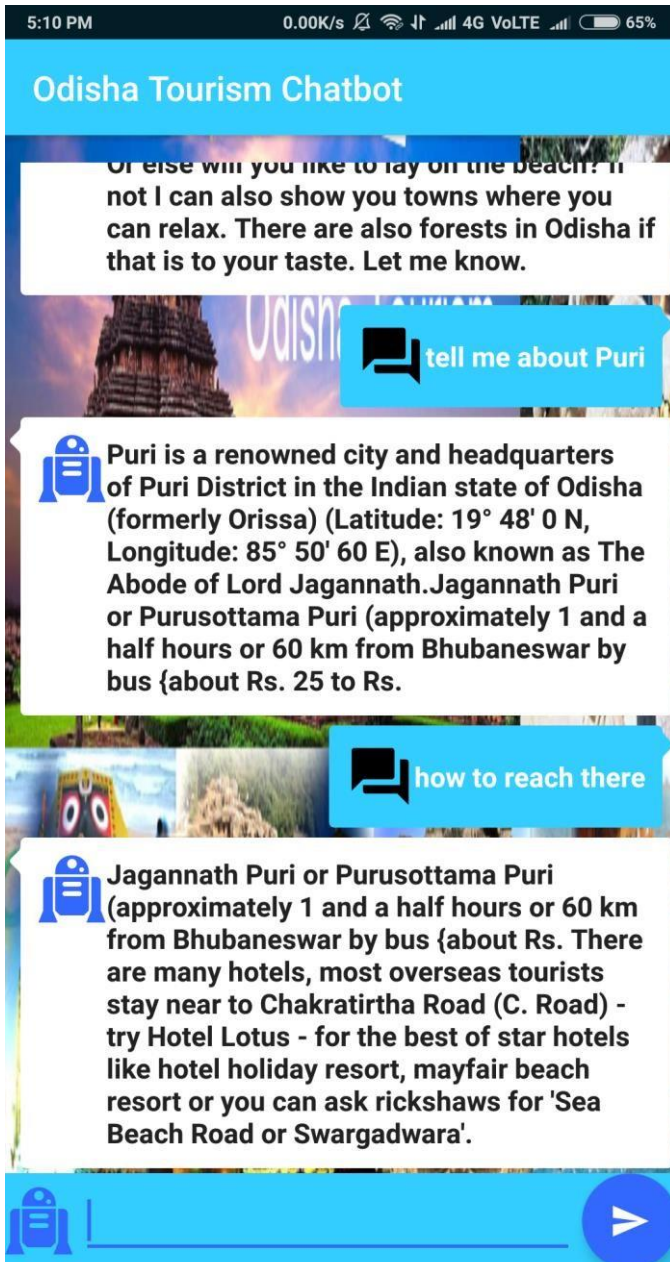
I am hrushi

Where do you want to go? I can show you hill-stations if your in the mood of trekking. Or else will you like to lay on the beach? If not I can also show you towns where you can relax. There are also forests in Odisha if that is to your taste. Let me know.

show me a hill station

You can visit the tea gardens of Daringbadi. You can climb the hills of Mahendragiri. You can see the grandeur of Taptapani. Let me know which place do you want to visit.





Conclusion And Futurework:

As the industry grows, more competitors will enter the market and growth will be exponential. Chatbots will be almost indistinguishable from their human counterparts, if anything, even better. Chatbots will power 85% of all customer service interactions by the year 2020. In fact, the average person will have more conversations with bots than with their spouse. we want things to work better and faster than they did in the past. That is just the way humans are. We will try to make things more efficient. Speech recognition has improved tremendously in the last couple of years, but it's nowhere it should be in our app.

Sources And References:

- [1] *S. Bird, E. Klein and E. Loper* , In “NATURAL LANGUAGE PROCESSING WITH PYTHON” page 140-142
- [2] *D. Jurafsky and J. Martin* , In “SPEECH AND LANGUAGE PROCESSING” page 217-219
- [3] *Andrew Ng* In “ DEEP LEARNING” page 147-160
- [4] *Huang, Jizhou, Ming Zhou, and Dan Yang*. "Extracting Chatbot Knowledge from Online Discussion Forums." *IJCAI*. Vol. 7. 2007.
- [5] The Watson Conversation API and its documentation
- [6] IBM Bluemix cloud platform and its documentation
- [7] Stackoverflow.com
- [8] Youtube.com
- [9] Tutorialspoint.com