



NATURAL LANGUAGE PROCESSING BASED CHATBOT FOR IOT DASHBOARDS

Kedar S Chikorde¹, Asst. Prof. Pankaja B Patil²

^{1,2}Department of Master of Computer Applications, Gogte Institute of Technology
Visvesvaraya Technological University, Belagavi, Karnatak

ABSTRACT

Natural Language Processing based chatbot is a programmed module that interacts with human being. These chatbots can be integrated with various IoT dashboards developed as web applications or mobile applications. Chatbot can act as replacement for humans and make conversation with application user, take some decisions in critical situations.

Paper takes an example of Weather Forecasting application which monitors real time IoT data continuously & then sends it to the dashboard application.

Index Terms: Artificial Intelligence, Chatbot, Machine Learning, Big Data

I. INTRODUCTION

The chatbot is an emerging technology for customer service support and dialogue system. There are two types of the chatbots. First, the most common one which uses the predefined set of rules. Second, more sophisticated and advanced version uses *Artificial Intelligence*.

Typical embedded IoT devices or sensors which are placed remotely will continuously send the data over the Internet to the application. The application can display the information sent to it on dashboard. At the same time this data can be analysed by the AI module using machine learning.

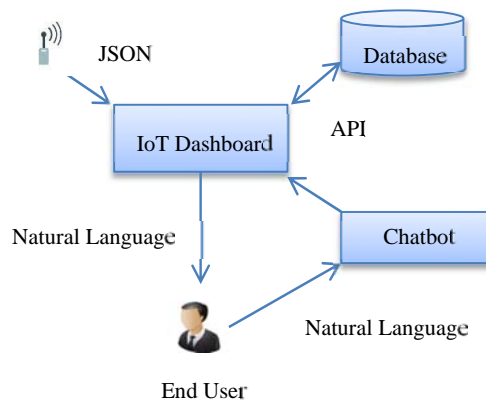


Fig. 1 Proposed model of IoT Dashboard with Chatbot

The users of the IoT dashboard can ask the questions in natural language. Chatbot will answer queries based on analysed data. This gives the user a feel of talking to an expert on the other side.

Fig. 1 gives the concept of how chatbot can be helpful when integrated with an IoT dashboard. The end user communicates with the chatbot and asks for the information or data of interest. The chatbot processes the input statements and responds to the user. This response can be a conversational response or a queried data from the database.

II. NATURAL LANGUAGE PROCESSING

Chatbots are used to do the communication between the computer and human being. There are various ways to build a chatbot. Artificial Intelligent chatbots are straight forward to develop using the AIML package provided by Python. AIML stands for Artificial Intelligent Markup Language. AIML is the standard XML file with file extension aiml. Each AIML file

contains category element in which pattern and template are sub-elements. Pattern refers to user inputs and template is the response to what user may expect. These files are processed by package aiml. We create standard start up file which acts as an entry point to all AIML files of chatbot.

An example of standard start up file that loads all other aiml files for chatbot named as startup.xml.

```
<aiml version="1.0.1" encoding="UTF-8">
  <category>
    <pattern>load aiml b</pattern>
  <template>
    <learn>chat.aiml</learn>
  </template>
</category>
</aiml>
```

An example of aiml file that defines the conversation named as chat.aiml.

```
<aiml version="1.0.1" encoding="UTF-8">
  <category>
    <pattern>HI</pattern>
    <template>Hello, I am bot</template>
  </category>
</aiml>
```

Python script that uses the aiml package and processes all aiml for chatbot named as myChatbot.py.

```
import aiml kernel =
aiml.kernel()
kernel.learn("startup.xml")
kernel.respond("load aiml b")
while
True:
    Print kernel.respond(raw_input(">> "))
```



```
C:\WINDOWS\system32\cmd.exe
W:\AI Python>python myChatbot.py
Loading startup.xml... done (0.02 seconds)
Loading chat.aiml... done (0.00 seconds)
Loading solar_current_chat.aiml... done (0.00 seconds)
>> hi
Hi, I am Smart Assistance!
>> what is your task?
I am bot. I can talk with you.
>> how do you operate?
I have Artificial Intelligence
>> save
Saving brain to bot_brain.brn... done (0.00 seconds)
>> quit
W:\AI Python>
```

Fig. 2 Console output of the chatbot

There are other open source packages available to build chatbot, like Apache's OpenNLP, which is open source library provided by an Apache to build interactive chatbots using Java. There are readymade frameworks like Amazon's Alexa, Google's API.AI and many more on

Internet which can be used directly into the application.

Capturing the entities within the natural language is another challenge. The entities can be like date-time, date of month, day of week, month of year or current running year. Also, custom entities must be supported. The challenge is when same word is having different meaning. For example, current can be treated as current in amperes in IoT dashboard or it can be current running data. Making understanding of user context is very difficult in natural language processing.

III. MACHINE LEARNING

Machine learning is still developing area in the Computer Science industry. It is not yet successful completely. Machine Learning algorithms are developed based on the data it receives. Still it is generating a lot of revenue as it processes tons of data from big data analytics.

In Natural Language Processing, machine learning has its own adverse effect. It self learns some of the natural language processing and context for inputs that user makes.

Algorithms are developed to take each input from the user and log it for machine learning.

Also, on IoT dashboards all parameter values have to be in some range. For example, weather forecasting dashboards have critical parameters that should not exceed some threshold. In such cases, application should automatically alert for out of range values.

IV. EXAMPLE DASHBOARD

The chatbot can be integrated with various IoT dashboard applications which is having the real time data to display on it. For convenience let us say we have created the weather forecasting application that continuously listen for the data that is been sent using various sensors. Fig. 1 illustrates how sensors send the data to dashboard application and how chatbot will help in analysing it and use.

The following Fig. 3 is an example of dashboard integrated with the chatbot. The user of the application can ask questions in the natural language. Like a weather forecasting application may have noise details, wind details, temperature details. The user can ask questions like “Give me noise details for next week” or “Noise temperature details of the Belgavi for next two days”.

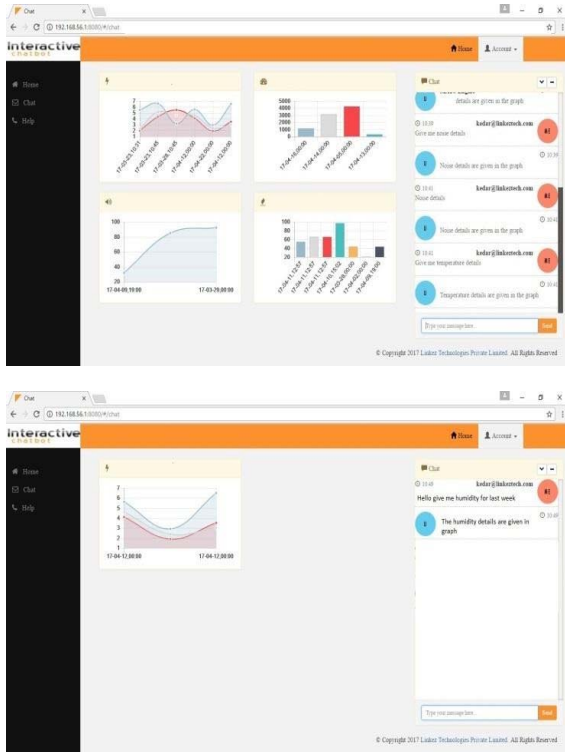


Fig. 3 Chat Example

As shown in the figures, the user is querying for the various parameters using the chatbot. Like this the chatbot can be integrated with various real time IoT dashboards.

The analysis part can be done by big data and machine learning using the Map Reduce and Hadoop. The data which is sent by these sensors can be stored on any SQL or NoSQL database and user can make REST calls to get the information they are interested in. Useability of the application will increase and user will get the feel of talking to human on the other side. Fig. 4 shows the analysis on the data.

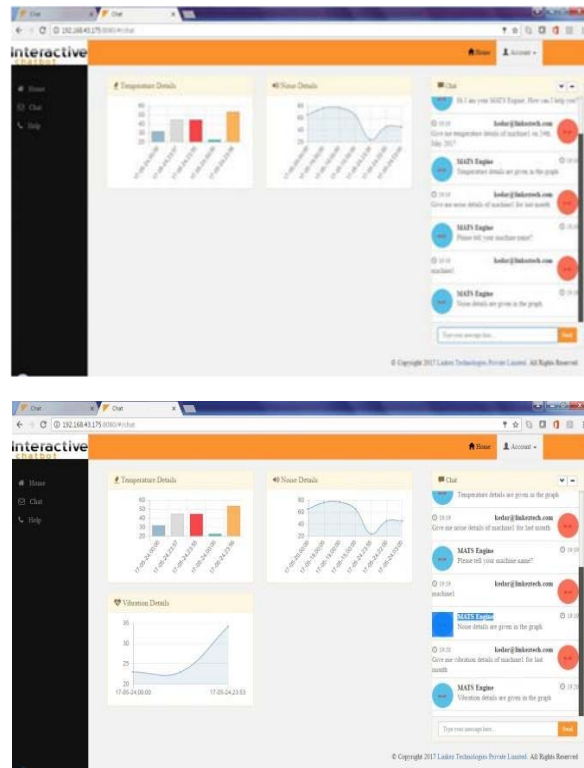


Fig. 4 Analysis on data

IV. APPLICATIONS

The cost involved in customer service support will be reduced by chatbot integration with the application. The analysis of data made by the sensors will help in checking the quality of user experience and making system intelligent to take appropriate decision. Also, giving the customer service support 24 X 7 is possible with chatbot. The problems which are occurring very frequently can be solved by training the application to look for similar problems and solve itself. Periodical updates through chatbot are possible.

V. CONCLUSION

Chatbot will aid in rich user experience of getting real time assistance from the machine. It will reduce human efforts in customer service. Scheduled alerts will help in critical scenarios. Response time for the queries is minor and accurate.

ACKNOWLEDGMENT

I acknowledge and thank Mr. Divyeshkumar Shah, Founder & CEO, Linkez Technologies Private Limited to give me the platform to explore this technology. I also thank

Ms. Pankaja Patil, Asst. Prof. Gogte Institute of Technology for guiding and encouraging me.

REFERENCES

- [1] Zed Shaw, “Learn Python the Hard Way”, third edition, A Very Simple Introduction to the Terrifyingly Beautiful World of Computers and Code.
- [2] Hans Fangohr, Introduction to Python for Computational Science and Engineering (*A beginner’s Guide*).
- [3] <https://developers.google.com/edu/python/>
- [4] <http://machinelearningmastery.com/how-to-implement-a-machinelearning-algorithm/>
- [5] <http://www.devdungeon.com/content/ai-chat-bot-python-aiml>
- [6] <https://en.wikipedia.org/wiki/Chatbot>