Data Warehousing and Analytics for ChatBot

by
NISHI MEHTA
201811047

A Project Report Submitted in Partial Fulfilment of the Requirements for the Degree of

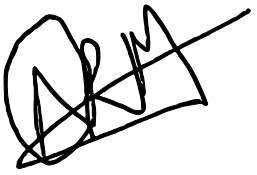
MASTER OF TECHNOLOGY
in
INFORMATION AND COMMUNICATION TECHNOLOGY
to

DHIRUBHAI AMBANI INSTITUTE OF INFORMATION AND COMMUNICATION TECHNOLOGY

05, 2020

Declaration

I hereby declare that
(i) the project comprises of my original work towards the degree of Master of Technology in Information and Communication Technology at DA-IICT and has not been submitted elsewhere for a degree,
(ii) due acknowledgement has been made in the text to all the reference material used.



Nishi Mehta

Certificate

This is to certify that the project work entitled has been carried out by Nishi Mehta (201811047) for the degree of Master of Technology in Information and Communication Technology at Dhirubhai Ambani Institute of Information and Communication Technology under my/our supervision.

Prof. Suman K. Mitra
Project Supervisor

Acknowledgements

I would like to extend my heartfelt gratitude to the Department of Information and Communication Technology, DAIICT to have provided me with an opportunity to work practically in the industry for the Major Project.

I would also like to extend the vote of thanks to both my Guide and Mentors of both the Institute and the Organization where I worked during the period. Both Prof. Suman Mitra and Mr. Anwesh Roy has been very keen on providing the required guidance during the course of the project and have played a key role in its timely completion.

I would also like to thank Mr. Vaibhav Nag and other teammates and mentors namely Mr Jay Mehta, Mr. Shripati Bhat, Mr. Imtiaz Bellary and Mr. BG Srinivas for their continuous guidance and insights which helped me in providing the right direction to my thoughts and attaining the desired objective.

Special Mentions for Mr. Phani Patchava and Mr. Shripati Bhat for providing us with the training during the initial onboarding period. Their efforts did make us feel very comfortable and confident once we actually started working for our teams.

Lastly, I would like to thank my parents and colleagues for their support through the duration of the project.

Contents

Declaration **1**

Certificate **1**

Acknowledgements **2**

Contents **3**

Abstract **4**

List of Figures **5**

List of Tables **6**

Data Warehousing and Analytics for ChatBot **7**

1.1 Introduction 7

1.2 Organization of the Project 8

1.3 Project Definition 9

1.4 Scope 9

Proposed Data Warehouse Approaches and Flows **10**

2.1 AWS based Data Warehouse 10

2.2 Elasticsearch as Data Warehouse 11

2.3 Finalized Data Warehouse 12

Data Analytics **14**

3.1 Introduction 14

3.2 Dimension for Analysis 14

In-Depth Analysis **16**

4.1 Measures of Analysis 16

4.1.1 Session Duration and Frequency of Interaction 16

4.1.2 Same Queries, Similar Queries, Queries with similar response 17

4.1.3 Default Message Analysis 18

Fight Against Covid Bot **19**

5.1 Covid Bot 19

Summary and Conclusion **21**

6.1 Summary and Conclusion 21

6.2 Pending Tasks 22

References **23**

Abstract

For businesses it has become mandatory to solve the issues and problems that the customers are facing to ensure the customer loyalty. Man has always looked to take help of machines to remove human intervention and once again machines succeeded in doing that. Isn't that strange we are using machines to provide services to humans? Well, that's how technology has grown exponentially! This time it is the customer service industry which we are talking about and the invention responsible for this is ChatBot.

The main aim of this project is to set up a data warehouse for a chatbot platform, set up ETL jobs so as to be able to push the existing as well as the incremental data to the warehouse which becomes the source for all our OLAP tasks.

The data warehouse is then further used for creating dashboards using efficient BI tools for the business team to be able to roll out their weekly reports with ease and be able to take the necessary measures.

One of the main intentions behind data analysis is to improve the quality of the bot that can further improve overall performance of the bot. For this purpose we have used interactions data in order to analyze usability, engagement and efficiency of the bot.

The overall intention is to come up with a quality index for the bot based on some of the parameters which can further help to know the quality of the bot as well as suggest some changes to owners of the bot and also analyze that the changes that are made reflects in improving the bot without actually affecting the quality/engagement of the bot.

In today's world when there are a plethora of different data warehouses as well as visualization tools are available, to choose one is a tedious task. One of the main aim of this project is to find an efficient data warehouse and BI tool that will be used for analyzing the data.

Keywords: Data Warehousing, Data Analysis, BI tools, NLP, ML Algorithms

List of Figures

		Page No.
Figure 1	Organization of the Project	8
Figure 2	Session Duration For a Particular Bot	16
Figure 3	Clusters of various categories for Default Message analysis	18
Figure 4	Overview of Users and Interactions for the Main Bot	19
Figure 5	Click Data count for the Menu Options	20
Figure 6	Trends for Users and Interactions for the Main Bot	20

List of Tables

		Page No.
Table 1	Traditional Data Store / Data Warehouse	7
Table 2	AWS / Elasticsearch	12

Chapter 1
Data Warehousing and Analytics for ChatBot

1.1 Introduction

Data Warehouse can be thought of as a data store that is capable of handling analytical tasks along with the traditional data transactional tasks. It is the process of collecting and managing data from a variety of sources to provide meaningful insights. It is used to connect and analyze data from heterogeneous sources. It can be considered a large amount of storage that can be used for analyzing data which can further improve the business. It is basically the heart of BI tools that can be used for data analysis.

The process of data warehousing includes an ETL process, an OLAP engine with the pre-existing OLTP framework, connection to several analytics tools and a visualization tool for visualizing the data and reporting it for business use-cases. A separate OLAP engine along with the existing OLTP engine enhances the analytics tasks on hand without affecting the transactional work-flow of the system. This provides enough structural stability for analytical reporting tasks, fetching translational data through ad-hoc query processing and decision making through stable visuals.

The main difference between Traditional Datastore and Data Warehouse is Datastores are designed to support transactional processes whereas on the other Data Warehouse is designed considering analytical tasks. There are few more other differences related to the processing of data such as Datastores supports faster INSERTs, UPDATEs AND DELETEs whereas Data Warehouse supports faster data retrieval even for large volumes of data.

Data is growing day by day and so it is difficult to understand. Most of the companies turn to BI tools that can help them analyze and present data in an understandable yet attractive manner. Out of a variety of tools available, it is difficult to select one that will satisfy all your needs. It is a bit easy to learn any visualization tool but selecting one is the actual task.

Table 1 : Traditional Data Store / Data Warehouse

Sr.No	Traditional DataStore	Data Warehouse
1	Designed to support high-end Transactional processes. (i.e. OLTP-driven)	Designed to support high-end Analytical Processes. (i.e. OLAP-driven)

2	Regular Updation of data	Data refresh to the warehouse a timely process, Data appended regularly but not updated.
3	Optimized for handling smaller and simpler operations like an 'INSERT' or a 'DELETE'.	Optimized for handling complex JOINS,AGGREGATE Functions and other Analytical operations.
4	Process-Oriented	Requirement-Oriented
5	Supports faster INSERTs, DELETEs and UPDATEs (Transactionality).	Supports faster Data Retrievals even for large volumes of data.

1.2 Organization of the Project

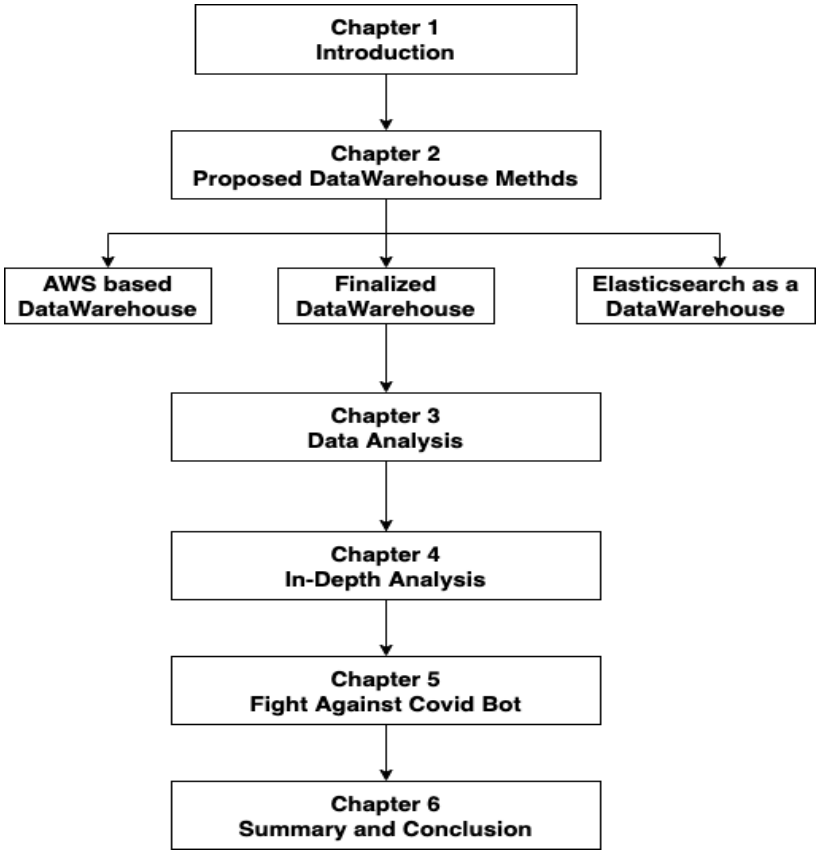


Fig 1 : Organization of the Project

Chapter 1 gives a brief introduction about Data Warehousing and also definition as well as scope of the project. In chapter 2 various Data Warehouse methods are proposed and finalized one data Warehouse. Chapter 3 describes the data analytics tasks that were carried out in order to improve the quality of Bot. Chapter 4 is all about in depth

analyzing tasks using NLP and machine learning algorithms. In chapter 5, a bot named as Fight Against Covid is discussed in detail along with a few of the screenshots of the dashboard. In chapter 6, I have summarized the entire project and also discussed few of the pending tasks.

1.3 Project Definition

Setting up a Warehouse over the traditional frameworks used and analyzing the data so as to develop dashboards, develop insights based on the tasks carried out and suggest improvement steps, with a valid POC to the development and the business team to act accordingly. The Warehouse hence setup, then be used as a data source for high-end business and development team weekly reporting and other OLAP tasks leading to improvement of bot quality.

The tasks also include building machine learning and NLP models based on the insights to improve performance and quality. Come with a quality index which is to be derived based on the analytics-tasks that are related to Usage of the bot to derive the quality of the bot.

1.4 Scope

To derive business insights from the data being captured, analyze bot performance and customer use by understanding their behavior patterns and build daily reports, weekly dashboards and periodic trends and insights.

To improve bot usability by deciding on a matrix that helps associate the bot's performance with the user feedback. Provide suggestions and assist the development team with several NLP tasks and improving faqs provided to the bot to enhance bot performance.

Chapter 2
Proposed Data Warehouse Approaches and Flows

2.1 AWS based Data Warehouse

Before I dive into explaining the pipeline of AWS based Data Warehouse, it seems mandatory to be brought to notice that the Chatbot data on cloud is hosted on AWS. Hence going with Redshift over Google BigQuery was a justified alternative. AWS charges for data inflow and outflow and use of its tools on cloud but it does not charge for the execution of query. While Google BigQuery charges for query execution along with the data inflow. Hence moving data out of existing AWS S3 buckets into Google's datastore would incur twice the cost just for creating an environment for analytics and reporting, hence AWS.

Data from both the sources ,i.e, Hadoop and the relational data are moved to S3 bucket using AWS pipeline. There is one more alternative to this is you can directly upload CSV/JSON files manually in the Amazon S3 Bucket. Once the data is in S3 Bucket, the second step is to move data from S3 to Data Warehouse Redshift. There is one service called AWS Glue that can handle this ETL process.
For this purpose AWS Glue is affiliated with Crawlers that can crawl the data files and create table schema without human intervention.

Once you have made a connection to the Redshift data cluster, you have to again go through the same process for warehouse tables and now the AWS Job helps us to transfer the data by selecting source and destination tables.

Some of the issues that we have faced are as below :

1. **Cost** : The cost associated with Glue (crawlers & jobs) was a bit too high. Since for data analytics we need real time data or atleast daily so running the jobs equal to the number of tables was quite costly. One more alternative to this could be to develop a python script for transferring data from AWS S3 to Redshift but that will drive away the whole purpose of using Redshift as Glue was playing a major role here.
2. **QuickSight** : We have used QuickSight for the visualization purpose but found out some disadvantages that it has few visual options and that are not up to mark

2.2 Elasticsearch as Data Warehouse

Elasticsearch is a search engine developed on Apache Lucene. Being a search engine, it is very quick in retrieving data and can scan through millions of records in a second of time. That is one of the main reasons why it is used in industry for the search related requirements. So Elasticsearch is basically a NOSQL DB that is deployed on clusters. The number of clusters depend on the use cases and the requirement and also the cluster-size can be increased/decreased keeping in mind the performance. And on-top of it all it has a well-developed analytics engine to perform analytics tasks quickly.

The idea of using Elasticsearch as a Data Warehouse is not something that comes easily to mind, but there are few reasons that contributed to our choice of using Elasticsearch as Data Warehouse. One of the main reasons is that as it supports storing and retrieving data of almost all the schemas with ease, it can be used with ease when we have data from multiple sources and multiple formats. Several other reasons are also there such as schemaless storage, High throughput, low latency and real time data retrieval.

While discussing data warehousing, we do include the ETL pipeline as well as a stable visualization tool for the process of reporting and analyzing data pushed to the warehouse as well. Hence to use Elasticsearch as a DW, the suggested Pipeline was the well established and widely used ELK stack. Kibana is used for the analytics and reporting tasks.

Since I have already mentioned that most of the Chatbot data is in S3 Bucket, so to drive the data from S3 to Elastic we wrote Logstash scripts. Logstash is a pretty powerful translation and transformation tool. Logstash broadly consists of three components namely Input, Filters and Output, of which Filters are optional.

Once the data is in Elastic we have used an open source tool named Kibana for the visualization purpose but later on found that elasticsearch does not support complex joins directly instead a parent - child relationship should be established.
But since we have to deal with relational as well as Hadoop data, complex joins to figure out the required field from each table were required and could not be done without establishing a hierarchy.

2.3 Finalized Data Warehouse

The issues that we faced in the above two approaches made us think out of the box. We thought of expanding the current Maria DB setup such that it will work as DW by handling OLAP tasks along with handling the existing OLTP task it already handles. For this task, we aimed at setting up a machine for handling Maria DB and have a VM connect to the visualization tool, Superset.

But there was one change here so instead of pushing the data from two different sources, we thought of pushing this data to Apache Drill. Apache Drill, connected to the Hadoop Hdfs provides a SQL interface to query hadoop's big data that has both the interactions as well as the relational tables. We have used CronJobs and Python Scripts to load incremental data from time to time.

Apache drill can be connected to Apache Superset using a SQLAlchemy URI and now the data can be used to create the charts that can be added in the dashboard.
Superset was chosen over the remaining tools such as QuickSight, kibana as it satisfies all our needs. It is an open source tool, user friendly UI and also it supports complex join queries which we can fire to retrieve the data.

For the in-depth analysis of data, data is exported off any data source and pushed into python and now using different python libraries such as matplotlib, seaborn analytics task is carried out. For some of the analysis tasks, Machine Learning Algorithms such as KMeans was used.

Extending Maria DB is really an feasible solution for us and it overcomes the problems that we have faced in the above two approaches. It is cost free and also we can use superset as a visualization tool on the top of that. Screenshots to the Dashboards are not included here due to the Company's policy of Non-Disclosure.

Table 2 : AWS based Data Warehouse / Elasticsearch as a Data Warehouse

Parameters	**AWS based DataWarehouse**	**Elasticsearch as a DataWarehouse**
Primary Database Model	Relational DBMS	Search Engine
Licensing	Commercial	Open Source
Cloud - Based	Yes	No

Schema	Yes	Schemaless
Query Language	SQL	Query DSL
Complex Joins	Yes	No (Parent Child Relationship mandatory)
ETL Pipeline	AWS Glue Crawlers and Jobs for ETL process	Logstash with Grok Patterns
Visualization Tool	QuickSight, not appealing to eye	Kibana, better than QuickSight
Notification Module	No such module available to trace sudden change or draw attention.	Notification Module available to trace sudden changes

Chapter 3
Data Analytics

3.1 Introduction

Once the data is in DW, now comes the most important thing! Analysis of data in order to enhance the bot performance. These requirements are relatively decided by a team consisting of members of the business team, developing team and the analytics team keeping the idea of improving the chatbot platform, usability of the bots already deployed and the quality of the responses provided by the bot.

Along with the above requirements, one more thing that we need to consider is customers. These analytics tasks are also intended at improving the way
the customers use the bots they have set up and deployed, report feedback provided by the end users of the bots, suggest improvements for improving bot efficiency and providing them with the required knowledge so as to improve their businesses, because their development would in turn lead to the company's overall development in terms of new customers and retaining older customers.

3.2 Dimension for Analysis

After finally deciding the analytics tasks, they were broadly distributed keeping a set of dimensions in mind, which are as follows:

a) On-Plan Customers v/s Non-Paying Customers
b) Interactions with LiveChats Agents v/s Interactions with bots
c) Bot Usage on Path vs Bot Usage on FAQs
d) Bot Deployment and User Usage By Geography
e) Bot Deployment and User Usage By Language
f) Usage Time

First dimension deals with the company's customers. So the chatbot platform has a detailed bifurcation of the services that it provides within each plan. The main divisions are Professional, Enterprise and Business which include various features depending upon the pricing. Also apart from the paying customers, we also have to consider non-paying customers that can potentially turn to paying if they are served well and to satisfaction.

Second dimension talks about Interactions with LiveChat Agents ,i.e, instead of bot replying to the customers, live agents as in humans will respond to the bots. So if we have more number of LiveChat requests that shows the number of times the bot is failing to respond to the user and manpower is required to address the customer's queries.

Third dimension talks about Path as well as FAQ interactions. So the bot gives us some flows out of which the user can select one so basically those types of interactions are called Path interactions. On the other hand, the flows that are set up may be not sufficient enough so the user might ask some questions apart from the flows, those are called as FAQ interactions and how well is bot trained to answer those requests.

The chatbot platform has bots deployed over various geographical areas and so analyzing the users based on their geography, usage of the bot, fall-off rate, preference of path over FAQ interactions, LiveChat requests over bot responding to the customers can help the customers to improve the bot performance.

Same is the case with languages, if the users of one language engage well but for the same bot users of another language drop off, so one of the possible reasons is that there might be issues with the language translation.

Chapter 4
In-Depth Analysis

4.1 Measures of Analysis

There are two types of analysis basically, one that can be done by firing some queries and the second one where you have to use python libraries, ML algorithms for getting insights from the data. The analytics tasks are broadly measured on the following tasks:

a) Average Session Duration and Frequency of Interaction

b) Similar Queries (Queries that have similar responses)

c) Default Message Analysis

4.1.1 Session Duration and Frequency of Interaction

There might be cases where for a particular bot, count of Total Interactions might be too high but the same bot might not be used in the last two months at all. Multiple reasons might create such a scenario, one of the reasons can be that the bot might be unable to answer the queries asked by users and so the user is not satisfied by the bot performance and hence the interactions count might drop off.

This gives us an insight that the customer might churn despite having high Interactions count. The second thing that one should observe is the amount of time the bot is being used for a particular session. Session Duration can help us in knowing whether a particular bot is utilized enough or not.

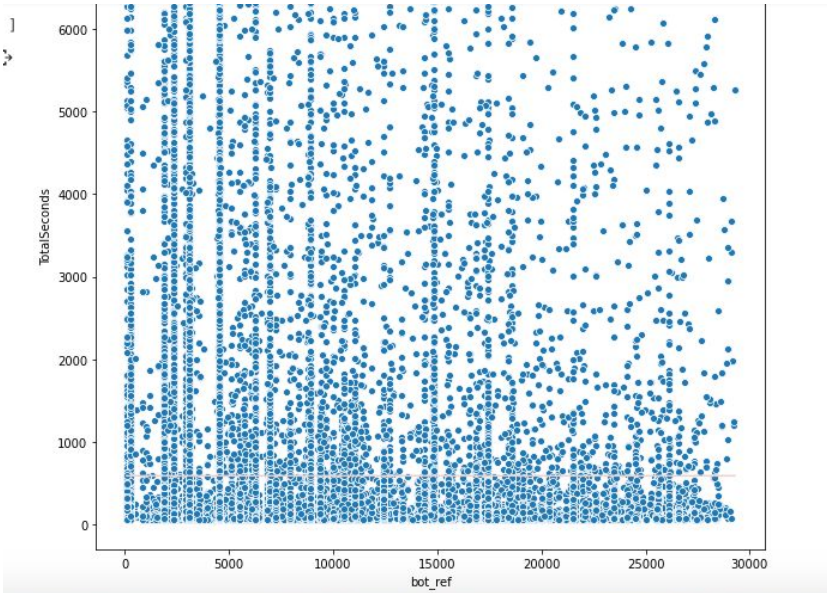


Fig 2 Session Duration For a Particular Bot

4.1.2 Same Queries, Similar Queries, Queries with similar response

It is mandatory to know whether bot has been trained well and is responding to same and similar queries in a definite manner. If the questions that are asked to bot are the same then the bot should reply with the same response in order to maintain consistency in the response delivery throughout the system.

For example let's consider that one college is using this bot then the questions asked related to the courses should be answered in a consistent fashion and is expected for a bot to provide the same response for the similar queries.
If the bot is trained as expected then it should respond the same to all these questions such as can you let me know the fees for Mtech? or What are the fees for Mtech or How much do I have to pay if I opt for an Mtech Course? and sometimes this can be a really tricky task. For this thing we have to use some of the characteristics of NLP which will be discussed below:

Removal of Punctuation and StopWords : When we talk about Grammar, stopwords and Punctuation plays an important role but while doing analysis these things do not have much importance. In Fact we should remove them otherwise they may show up as different phrases and might also add to the cluster. Hence it is desirable to remove them.

Tokenization : The process of splitting the text segments into small words which are called as tokens is called Tokenization. Tokens can be considered as the heart of the system and the first step for any text based system. These tokens are then further used in stemming and lemmatization. NLTK permits tokenization for words (splitting a sentence into a set of words) as well as sentences (splitting a complex sentence into root sentences) using sent_tokenize and word_tokenize.
The main reason behind using the sent_tokenize is that most of the FAQ we have used to train bot are full sentences. Now tokens are extracted from these sentences that are further compared with the query to generate a score that will determine the output that the bot should display for the query that is asked.

Stemming and Lemmatization : These terms are very similar in meaning yet different in operation. Stemming means to get to the root word of the provided token by normalizing them into their simplest forms. Comparison of text in their root decreases loss of similar words which would otherwise be lost due to different verb forms of the same word. Lemmatization on the other hand helps bring to the root word of the lowest degree of the word, hence adds a contextual

dimension to the process. Lemmatization is, hence the method of grouping different forms of the same word so that they can be considered as a single item in analysis. For instance if a word better is present in the sentence, it would be converted to good, which is the root word of better's lowest degree good. We used the PorterStemmer as it removes morphological prefixes and suffixes from the words making stemming more effective. Also this algorithm provides the highest word-stem accuracy amongst many nltk provides.

4.1.3 Default Message Analysis

There might be some of the cases where the bot is unable to respond to the questions asked by the users and so it will just display one default message saying "Sorry, I am unable to understand your query". This dimension is a really important one because it can help us to know how well the bot is trained.

Once we have the list of the questions, we can have those questions included in the FAQ and can train bot on those questions as well in order to increase the quality of the bot and these can also help customers to increase the bot usage.
We have used the K-Means Algorithm in order to broadly categorize the questions so that we can have a look at the categories and train bot accordingly.
These categories can be referred to as clusters in terms of K-Means.

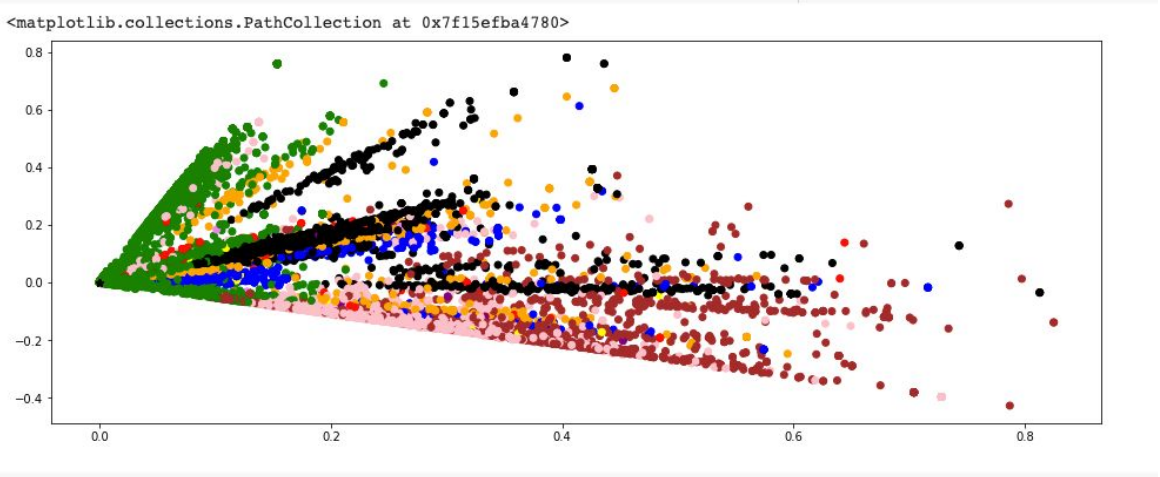


Figure 3 : Clusters of various categories of default message analysis

Chapter 5
Fight Against Covid Bot

5.1 Covid Bot

Company decided to build, train and deploy a Covid bot that is free to use and that can really help people with the right information and contribute a bit to help the world in this pandemic.

This bot is divided into two phases based on languages. Multilingual bot is the main bot that has English, Hindi, Deustch, Arabic and Italic. The other one is the regional bot that has English, Gujarati, Tamil, Telugu and Bengali to help the local audience to know more about the Covid bot. It has different flows such as General Information, Latest News, Myths, Precautions, Help centres, I am bored and similar such flows.

The Faqs for the bot included questions on the following Categories for which the bot was trained:

- General Information
- Precaution and Prevention
- Questions on Lockdown and Prevention Strategies
- Myths
- Impact on Society, Organizations and Businesses
- Testing

We have set up two different dashboards ,i.e, one for the Multilingual bot and the other for the regional one so that the company can analyze the charts and get information related to the bot.

Finally, I will share few of the screenshots of the dashboard that are as follows:

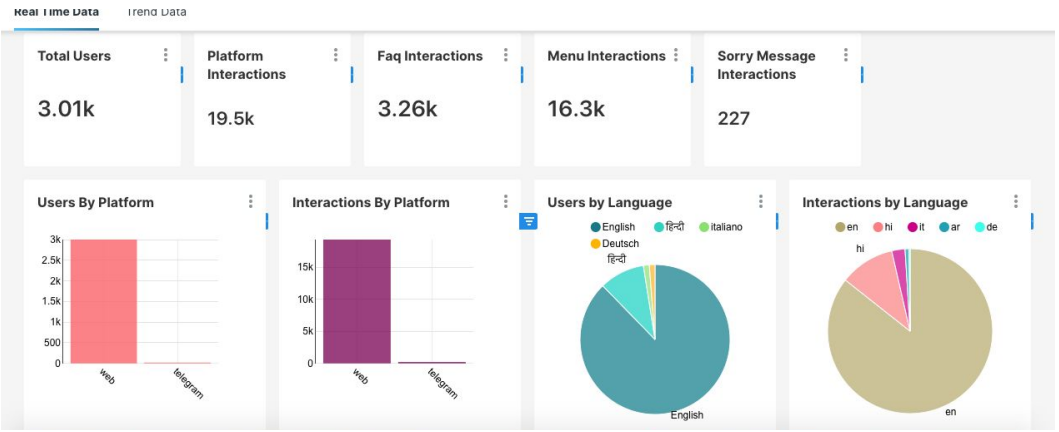


Fig 4 : Overview of Users and Interactions for the Main Bot

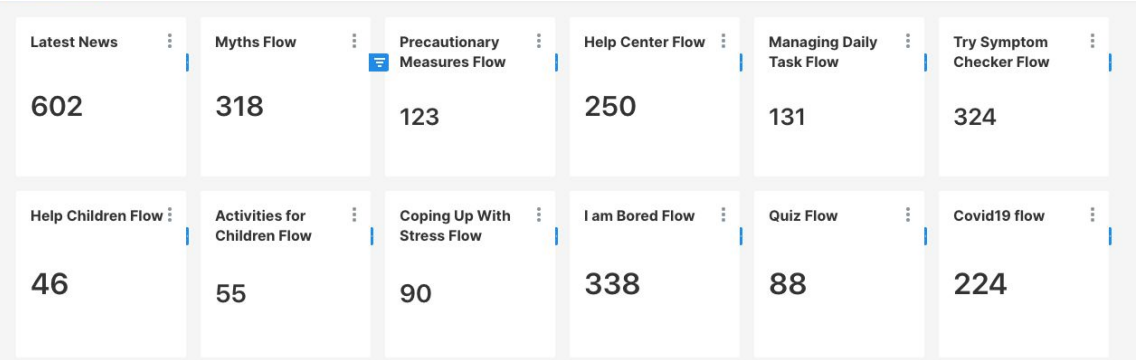


Fig 5 : Click Data Count on Menu Options

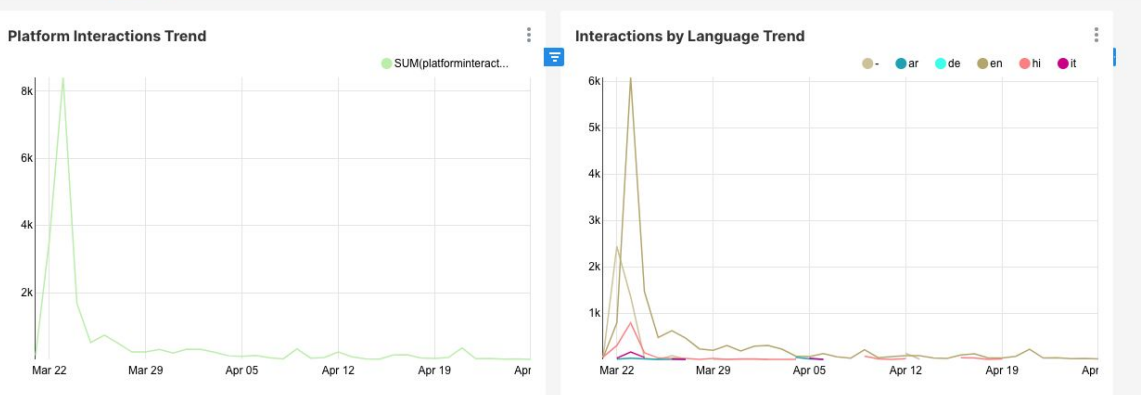


Fig 6 : Interaction Trends for the Main Bot

Chapter 6
Summary and Conclusion

6.1 Summary and Conclusion

While dealing with Chatbots, it is necessary to maintain and avail data from all different sources to be able to analyze the bot, it's performance and similarly the performance of the platform that hosts it. To make data reach the analytics cluster , with minimal or no data loss and to be sure of not affecting the production data sources, it was necessary to set up a data pipeline and an entire warehousing framework to host data for the OLTP tasks well.

This task was achieved after several proof of concepts , and trying out several pipelines and warehouses starting from a cloud based AWS Redshift, to a Search Database in ElasticSearch and Finally to Hadoop. Hadoop being affiliated with a distributed File System a Yarn Framework proves highly effective to handle big data from bot interactions and relational tables from MySQL. Drill developed on Hadoop helps use a SQL interface to query the data and a SQLAlchemy URI
is used to connect the same to Apache Superset to visualize and dashboard. The same tool is used for reporting while lengthier analytics tasks will be done by exporting data from Superset to Python.

Performing Analytics Tasks not only helps evaluate the current position the bot and the bot platform are in, in terms of the dimensions laid out but also helps provide several insights that can be used to improve the position of the platform, the bot and hence in turn the customer using the bot,working in the favour of developing healthier relationships with the customers.

We intend to develop a rating for each bot based on several parameters and provide specific bot specific insights to be able to improve the bot and its rating. We are on course of completing these tasks and intend to complete them shortly.

Along with business, the company's attitude towards contributing their bit to help the people by setting up and launching a bot on COVID19, and have us work on it has made us appreciate the effort and devotion towards serving not only our customers, but reaching to the masses without thinking of money matters.

6.2 Pending Tasks

The DataWarehouse setup is complete and we intend to roll out based on additional requirements to the Business team and the Development Team as and when required, along with an update weekly.

Analytics Tasks are also on the course of completion and once the measures are clearly analyzed and laid out, it will be easier to enumerate methods of assigning ratings to the bots based on these measures.

Build up a few models based on the insights that can help us further to retain customers as well as improve the quality of the bot.

References

[1] Engati, The best free chatbot platform to integrate your bot, 2019. Accessed on: May 02, 2020. [Online]. Available: https://www.engati.com/

[2] Oracle, Data Warehousing Guide, 2002. Accessed on : May 02, 2020. [Online]. Available: https://docs.oracle.com/cd/B10500_01/server.920/a96520/concept.htm

[3] AWS, What is AWS Glue. Accessed on : May 02, 2020. [Online]. Available : https://docs.aws.amazon.com/glue/latest/dg/what-is-glue.html

[4] AWS, Amazon Redshift Documentation. Accessed on : May 03, 2020 [Online]. Available : https://docs.aws.amazon.com/redshift/?id=docs_gateway

[5] AWS, Amazon Simple Storage Service Documentation. Accessed on : May 03, 2020. [Online]. Available : https://docs.aws.amazon.com/s3/?id=docs_gateway

[6] AWS, Amazon QuickSight Documentation. Accessed on : May 03, 2020 [Online]. Available : https://docs.aws.amazon.com/quicksight/?id=docs_gateway

[7] The Apache Software Foundation, 2019. Accessed on : May, 04, 2020 [Online]. Available : https://superset.incubator.apache.org/installation.html#aws-athena

[8] Get Started with Elasticsearch. Accessed on : May 05, 2020 [Online]. Available : https://www.elastic.co/

[9] Eduonix Learning Solutions, April 23, 2019, Accessed on : May 08, 2020 [Online]. Available : https://blog.eduonix.com/artificial-intelligence/clustering-similar-sentences-together-using-machine-learning/

[10] Logstash Reference, Accessed on : May 09, 2020 [Online]. Available : https://www.elastic.co/guide/en/logstash/7.6/index.html