INFORMATION RETRIEVAL PROJECT 4
CHATBOT
TEAM 8

Kartik Sehgal
Shalini Agarwal
Siddhant Gupta
Ann Konnyali

DEVELOPMENT STACK:

- **Frontend** - Vue.js framework with Vuetify library
- **Backend** - Python Flask app
- **Database** - Solr built on Apache Lucene
- **Visualisation** - Microsoft Power BI and Excel

APPLICATION FEATURES:

Our application has the following features –

1. When a user enters a query, the frontend fetches a response from the backend and displays it as a response to the query.
2. The user can press the send button, or press enter, to send the query. Empty queries won't be sent.
3. The user can choose any combination of the 5 topic filters.
    a. At least one topic filter must be selected.
    b. At all times, whether one filter is selected, or all are selected, the chitchat dataset is selected, and the app tries to analyse whether the query should hit the chitchat dataset or the reddit dataset.
    c. The user should click the Apply button to make sure the changes to the filters are stored in the backend.
    d. The Apply button is only available if there is a valid change in filters.
4. There are two buttons in the app bar linking to our data visualisation charts – the first includes the statistics for reddit comments and the second includes the statistics for reddit submissions. The reddit comments' chart further contains two pages giving multiple visualisation charts.

METHODOLOGY:

Data:

- We used the data collected in Project 1 of our IR course for data collection in the chatbot project.
- Since we had data from 4 different team members, we had to combine and create two separate files - one for submissions and one for the comments.
- In this data, we didn't require the subreddit submissions, hence we identified and removed all the subreddit submissions and their associated comments and only kept data from the topic wise submissions.
- Due to the combining of data it was prone to having duplicates. We identified and removed all duplicates using their unique ID.

- Additionally, we labelled each comment using their parent submission. As our dataset consisted of both - comments to submissions, and comments to comments - this required recursively searching through our corpus to find the original submission to which each comment belongs.
  The topic of each submission had already been added in Project 1.

Frontend:

For the frontend, we have used Vue.js framework along with Vuetify library. Using Vuetify made things easier as we did not have to create many custom components. Our webpage has the following structure:

1. App.vue is the base component
   a. In App.vue, we firstly have an app-bar that we use to display our logo and the links to our data visualisation charts.
   b. We then have a Grid.vue component which is used to give structure to our application. Within Grid.vue, we have two components
      1. TopicFilters.vue – This contains the topic filters.
      2. ChatBox.vue – This is our main chatting component where –
         a. The user sends messages
         b. The chatbot responds to the user
         c. The user types the messages
2. We also have a store to maintain the states.
3. For making API calls, we are using the “axios” library.

This frontend is hosted on a virtual machine (VM) on Google Cloud Platform (GCP).

Backend:

- The frontend communicates with the backend via REST API calls to a Flask app (in Python) which is also hosted on a VM on GCP.
- This includes the user input as well as the selected topic filters.
- The backend further processes the user input - removing special characters and detecting keywords which indicate “chitchat” query and formatting a query to the database. If

- For the information retrieval database, we continue to use Solr as in previous projects.
- The database index is implemented using BM25 similarity.
- We use dismax query parser as it works well for free text queries
- We query the ‘parent_body’ field with the user input. This identifies a comment which was a response to a comment/submission similar to the user input. Thus we can use the body of the search result as a response for the chatbot.

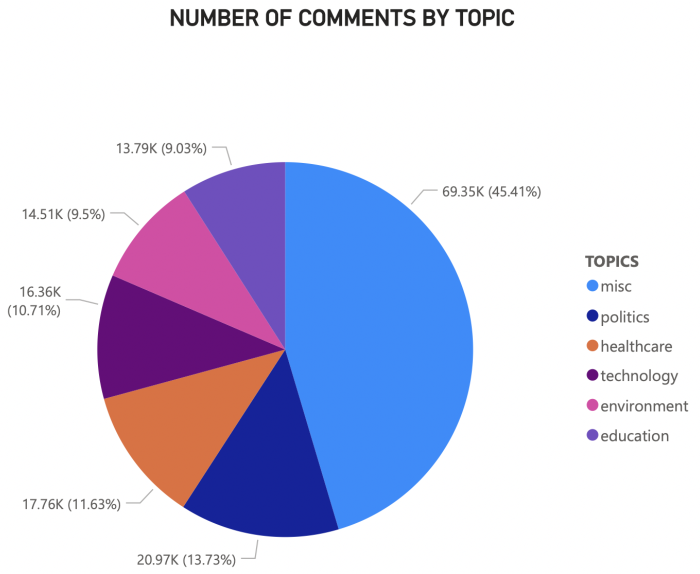
- Topic filters are implemented using the ‘fq’ query field, which allows us to efficiently filter the database to the requested topics, thus speeding up the query result.

Visualisation:

In the data visualisation section, we have analysed data in our comments and submission data frame and have created some visualisation charts based on our findings. We used Power BI and Excel software for generating these interactive visuals.

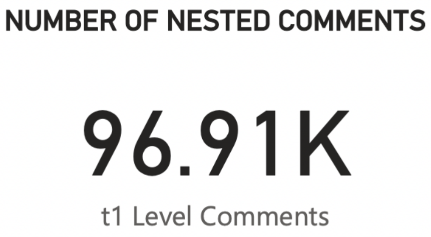
Comments Data Frame:-

1. Generated the topic wise comment count using pie chart in Power BI. This was created using the topic field in the comments data frame which gave the count of the number of comments. Depicted in figure.



A pie chart depicting the number of comments from each topic.

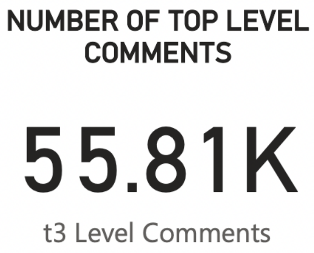
2. Generated the number of nested comments for the submissions.



NUMBER OF NESTED COMMENTS

96.91K

t1 Level Comments

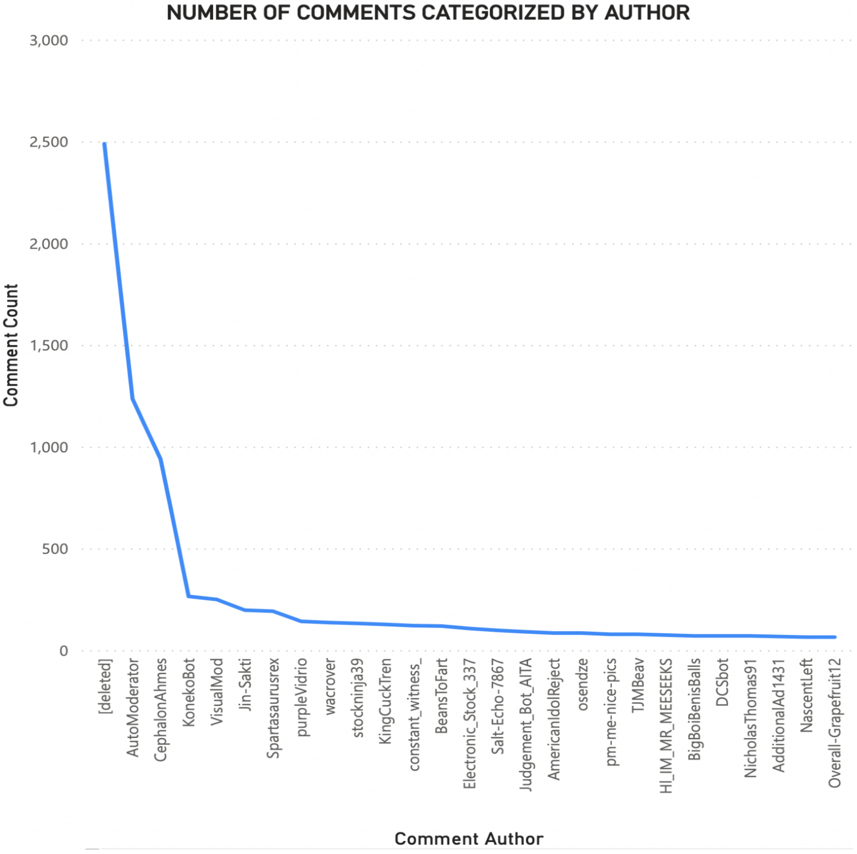


NUMBER OF TOP LEVEL COMMENTS

55.81K

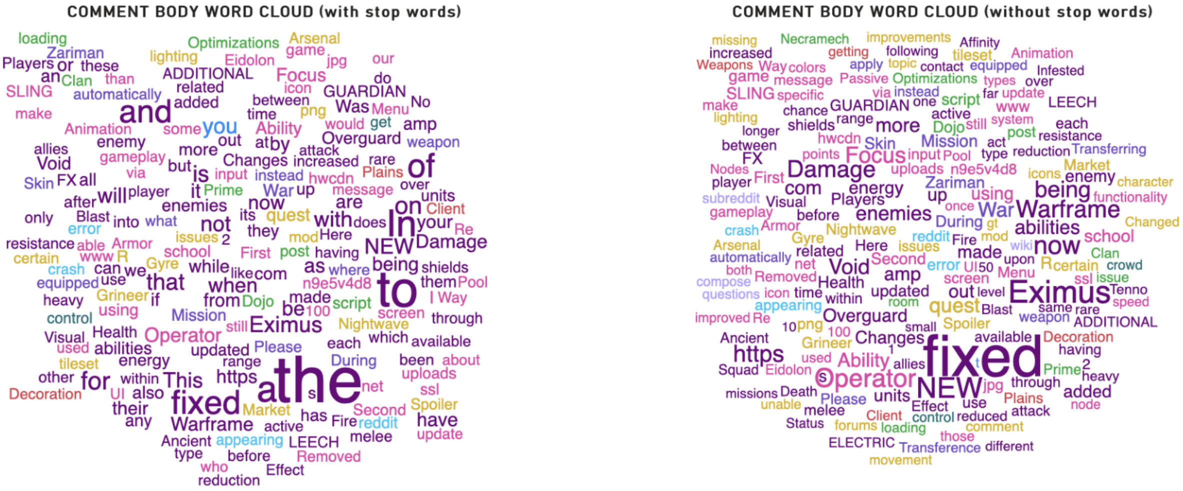
t3 Level Comments

3. Generated the line chart for segregating the number of comments by each author in our Reddit dataset. The X axis is represented by the author attributed and Y axis is represented by the comment id. Depicted in the figure.



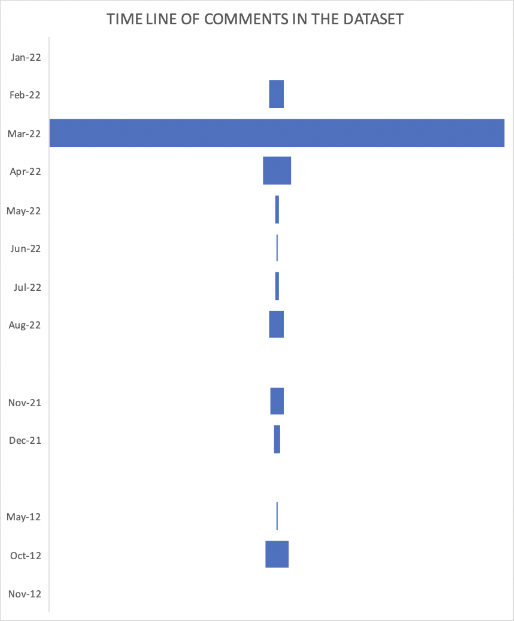
A line chart depicting the comment count by the author on Reddit in our dataset depicted in descending order

4. Generated the word cloud of most common words found in body attribute (comment body).



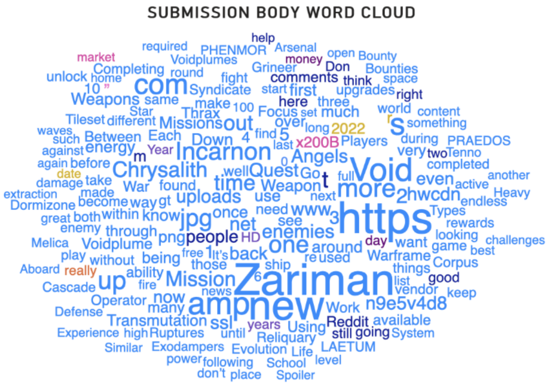
Word clouds depicting the comment body with and without stop words.

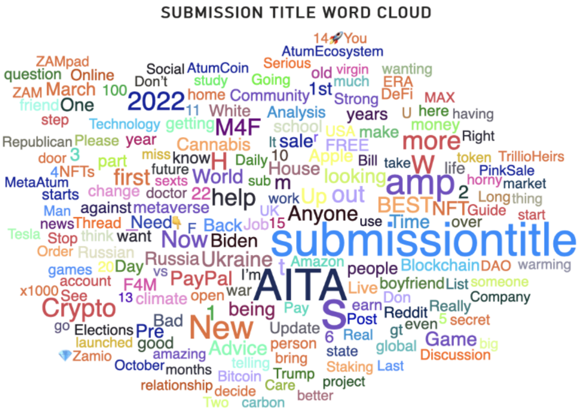
5. Timeline of comments in our dataset depicted using a funnel chart. We filtered the created_at field of the comments data frame and categorized it by month and date to create the chart.



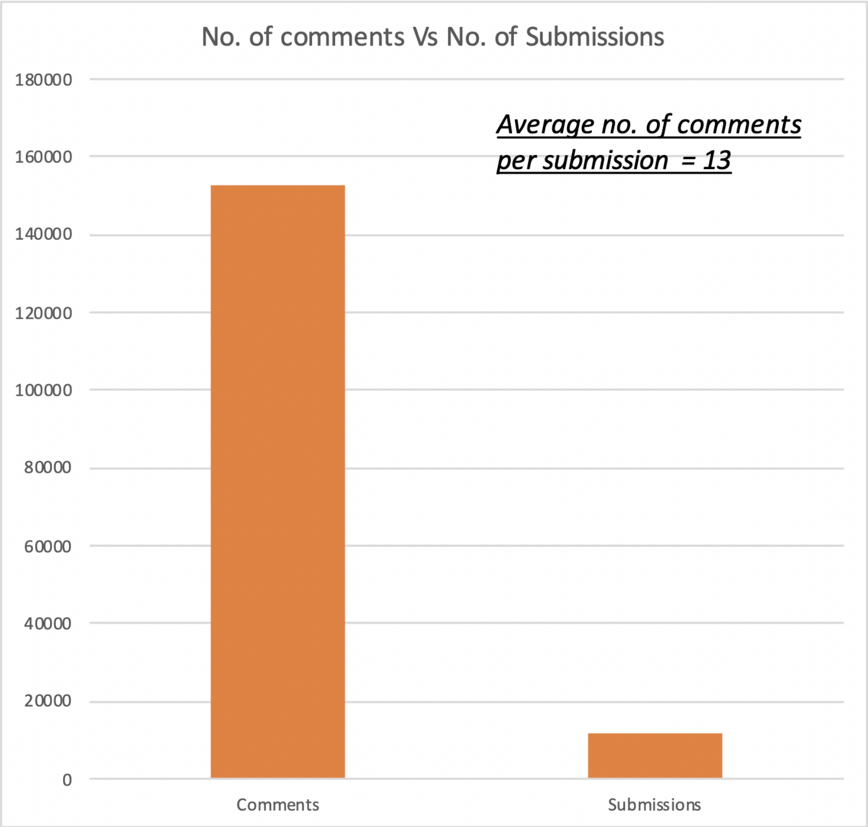
Funnel chart depicting the number of comments in the dataset categorized by the month and year.

6. Submissions Data Frame:-
   Generated a word cloud for the submission body and submissions title





Generated a bar chart to show the number of comments and number of submissions in our dataset.



7. Chit chat dataset:-

Word cloud for chit chat dataset with and without the stop words respectively.

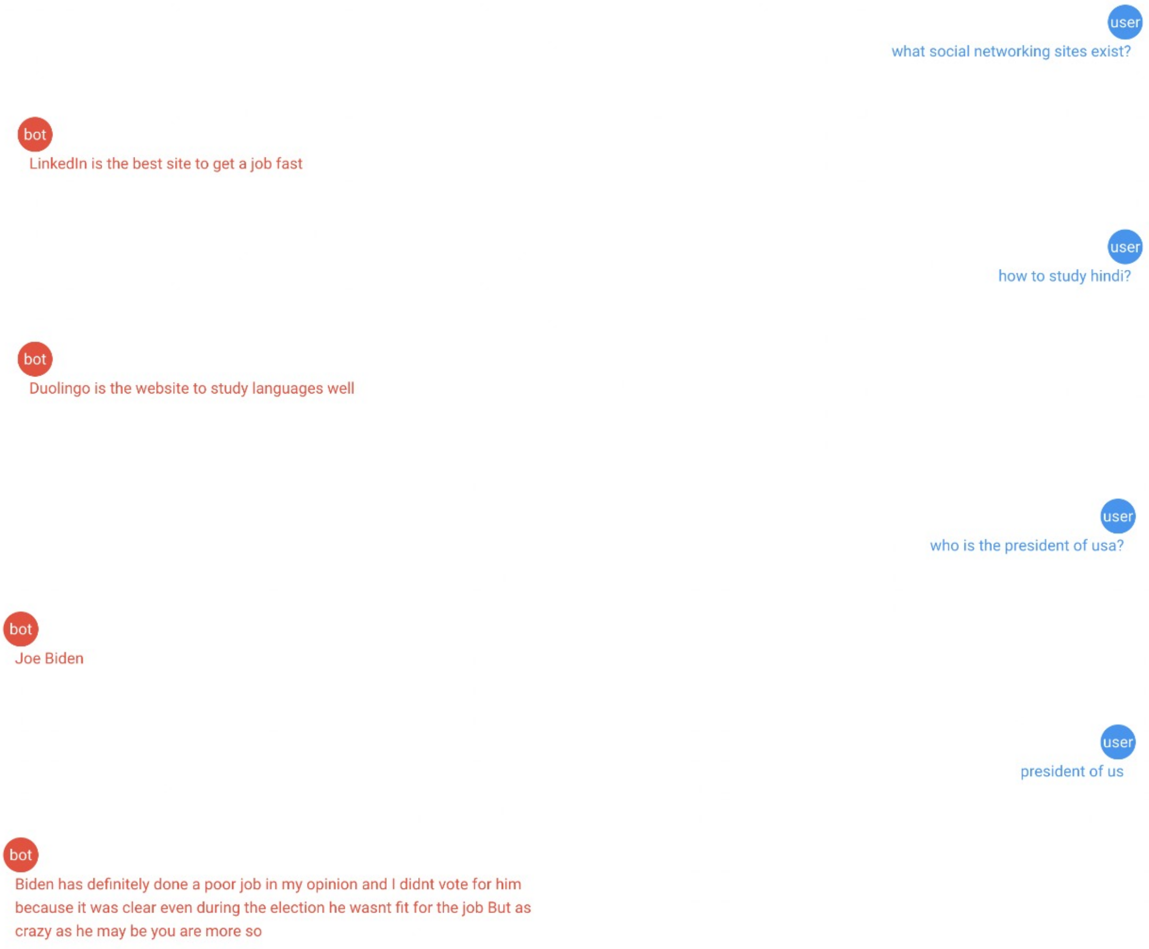
to
and
you
the
a
that
would
hello
have
are
for
of
is
it
in
but
there
this
like
be
was
good
can
just
dont
get
people
anyone
really
one
will
like
think
probably
going
hello
well
time
talk
chat
thanks
hey
yeah
thats
talking
see
want
day
know
maybe
cool
still
things
something

TEAM CONTRIBUTION:

- Kartik Sehgal: Entire frontend and hosting
- Shalini Agarwal: Data gathering and visualisation
- Siddhant Gupta: Backend and Solr query functionality
- Ann Konnayli: Backend API, Video demo

All member were part of the discussion and design strategy for the application

CHAT SAMPLES:



user
what social networking sites exist?

bot
LinkedIn is the best site to get a job fast

user
how to study hindi?

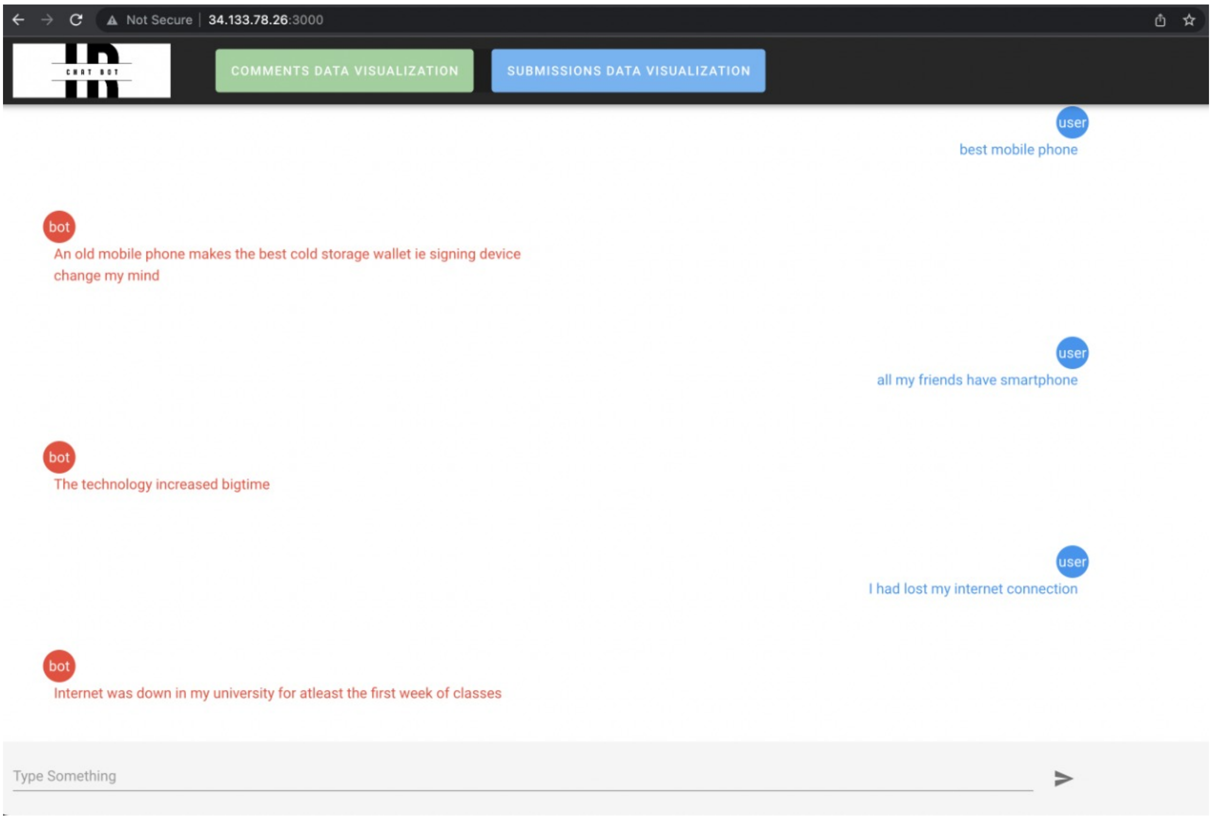
bot
Duolingo is the website to study languages well

user
who is the president of usa?

bot
Joe Biden

user
president of us

bot
Biden has definitely done a poor job in my opinion and I didnt vote for him because it was clear even during the election he wasnt fit for the job But as crazy as he may be you are more so


Not Secure | 34.133.78.26:3000
CHAT BOT
COMMENTS DATA VISUALIZATION
SUBMISSIONS DATA VISUALIZATION
Type Something

user
best mobile phone

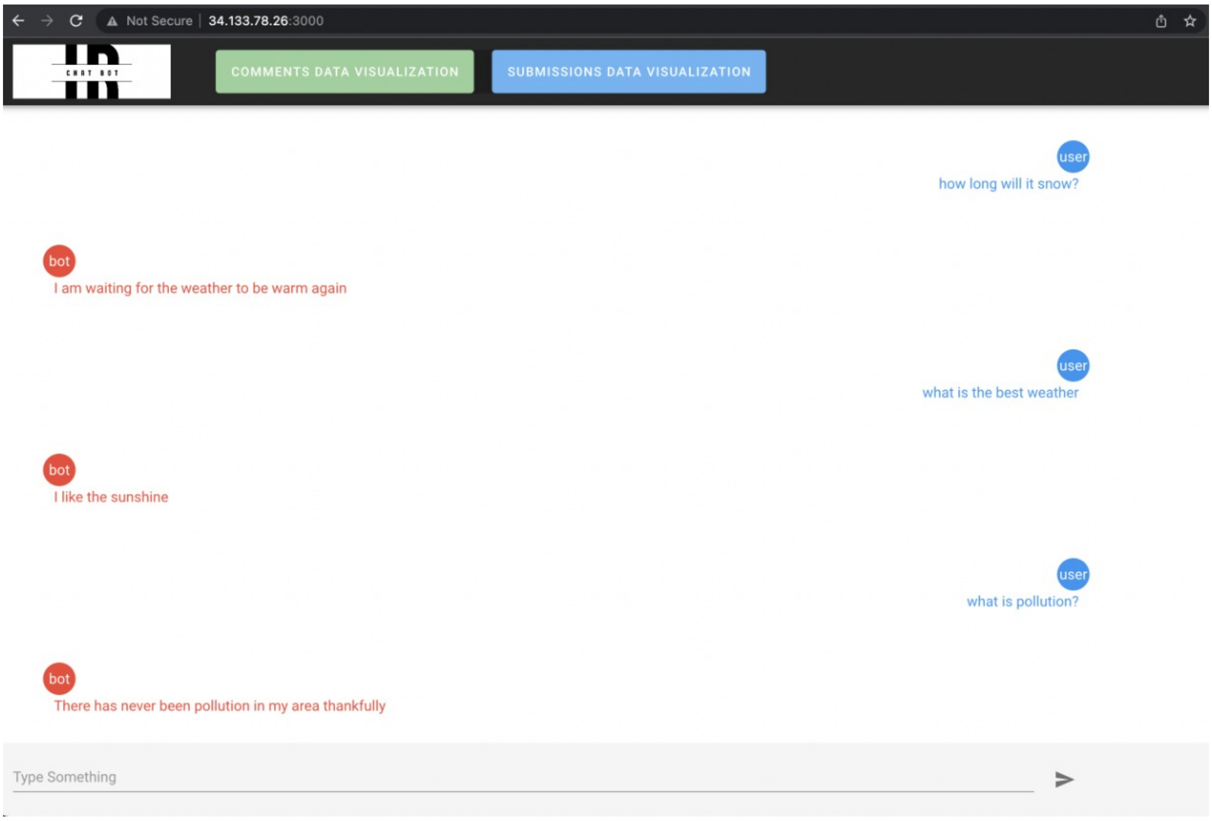
bot
An old mobile phone makes the best cold storage wallet ie signing device change my mind

user
all my friends have smartphone

bot
The technology increased bigtime

user
I had lost my internet connection

bot
Internet was down in my university for atleast the first week of classes


Not Secure | 34.133.78.26:3000
CHAT BOT
COMMENTS DATA VISUALIZATION
SUBMISSIONS DATA VISUALIZATION
Type Something

user
how long will it snow?

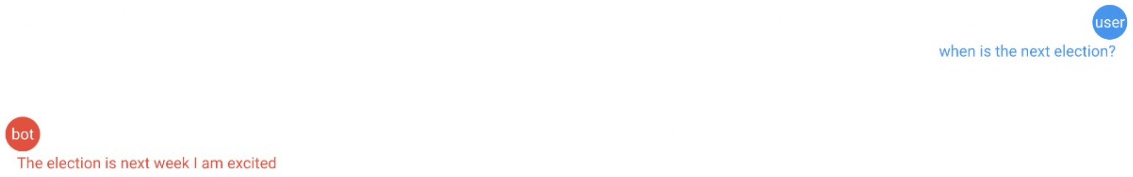
bot
I am waiting for the weather to be warm again

user
what is the best weather

bot
I like the sunshine

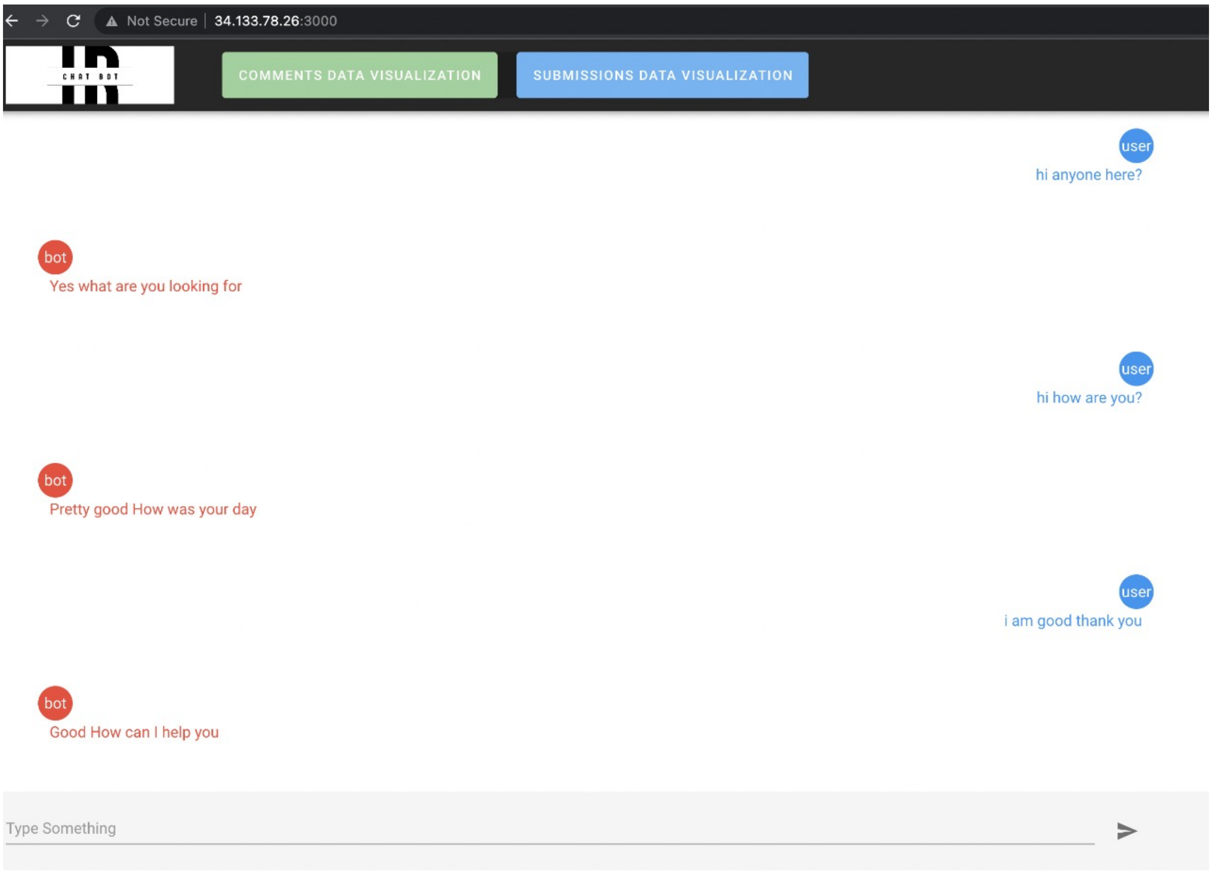
user
what is pollution?

bot
There has never been pollution in my area thankfully



when is the next election?

The election is next week I am excited



hi anyone here?

Yes what are you looking for

hi how are you?

Pretty good How was your day

i am good thank you

Good How can I help you