CS 410 Project Progress Report

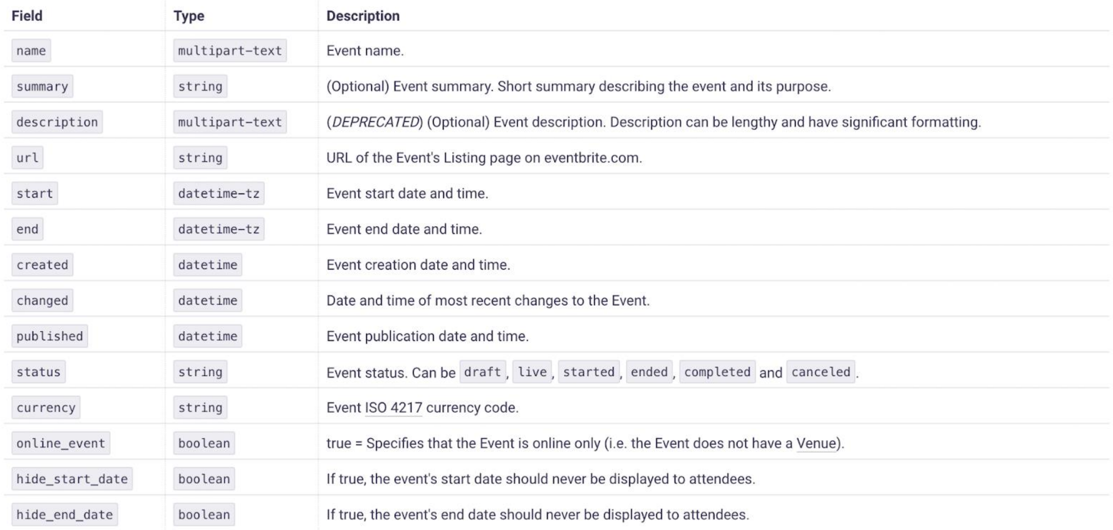
Team Name: Model Behavior
Team Members: gputcha2, aanil2, ksamant2, pp32

Which tasks have been completed?

The tasks we focused on in the initial phase of our project and that we have completed include Data Exploration & collection and creating the Project Workflow and coming up with the basic overview of the model and UI. We each looked at different websites and tried to figure out ways to scrape the text event data from there. While doing this we also faced a few challenges which have been listed in the last section.

To access public free Facebook event data, we found a few helpful links and API's eg: How to use Facebook Graph API and extract data using Python! | by Ravi Ranjan | Towards Data Science.) We also found https://pypi.org/project/facebook-crawler/ which can be leveraged to crawl event data. At the same time while looking at this data and events discovered some challenges such as Posting images instead of actual text.

We have also gained a basic understanding of the Eventbrite API. The API is REST-based and always returns responses in a JSON format. It consists of a number of objects but based on our research we will be primarily focusing on the "Event" object. There are several public fields available for use from the Event object.



Field	Type	Description
name	multipart-text	Event name.
summary	string	(Optional) Event summary. Short summary describing the event and its purpose.
description	multipart-text	(*DEPRECATED*) (Optional) Event description. Description can be lengthy and have significant formatting.
url	string	URL of the Event's Listing page on eventbrite.com.
start	datetime-tz	Event start date and time.
end	datetime-tz	Event end date and time.
created	datetime	Event creation date and time.
changed	datetime	Date and time of most recent changes to the Event.
published	datetime	Event publication date and time.
status	string	Event status. Can be draft, live, started, ended, completed and canceled.
currency	string	Event ISO 4217 currency code.
online_event	boolean	true = Specifies that the Event is online only (i.e. the Event does not have a Venue).
hide_start_date	boolean	If true, the event's start date should never be displayed to attendees.
hide_end_date	boolean	If true, the event's end date should never be displayed to attendees.

We expect that we will need to deserialize the JSON returned by this API using a JSON deserializer.

We applied for Academic Research access to the Twitter API. We are planning to use #events, #Chicago etc.. and other combinations based on the data requirements below to extract the data. We also found some helpful links such as https://towardsdatascience.com/hands-on-web-scraping-building-your-own-twitter-dataset-with-python-and-scrapy-8823fb7d0598

We also defined and narrowed down our dataset requirements:
Dataset requirements:

1. Source: Twitter, Facebook, Eventbrite
2. Data location restriction: Chicago, Seattle, Champaign, Arlington
3. Time frame: 1 week
4. Sample Data Format:
    a. Event name
    b. Event Datetime
    c. Event Location
    d. Event Description
    e. Event Link (ticket or to website)

This made us rethink and adjust the workflow accordingly.
Application Workflow

1. User Input (query, location, time)
    a. Query: contains keywords which user expects to see in the event name
    b. Location: chosen location
    c. Time : selected time frame by user
    d. Sort by (if time permits)
2. Sorting results based on ascending order of date, popularity (implicit feedback from users), relevance

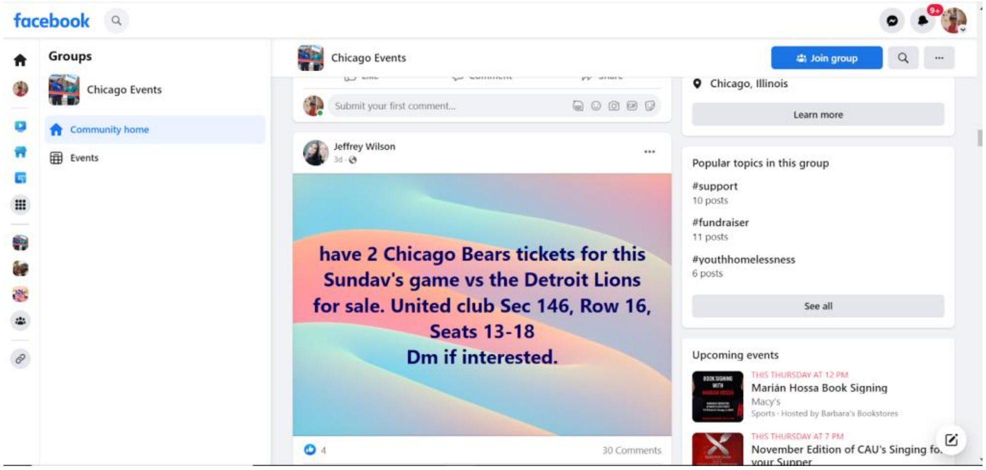
Which tasks are pending?

Based on the workflow created, our remaining tasks include:

1. Collecting the datasets, cleaning them and formatting them as mentioned above.
2. Implementing Frontend design (using React)
3. Implementing Backend (using Python)
    a. Build the model
        i. BM25 will be used
        ii. J. Mercer will be attempted and if better results, will be used.
4. Connecting frontend to backend
5. Train model using data collected
6. Testing and Evaluation to improve the model
7. Documentation and creating the submission video.

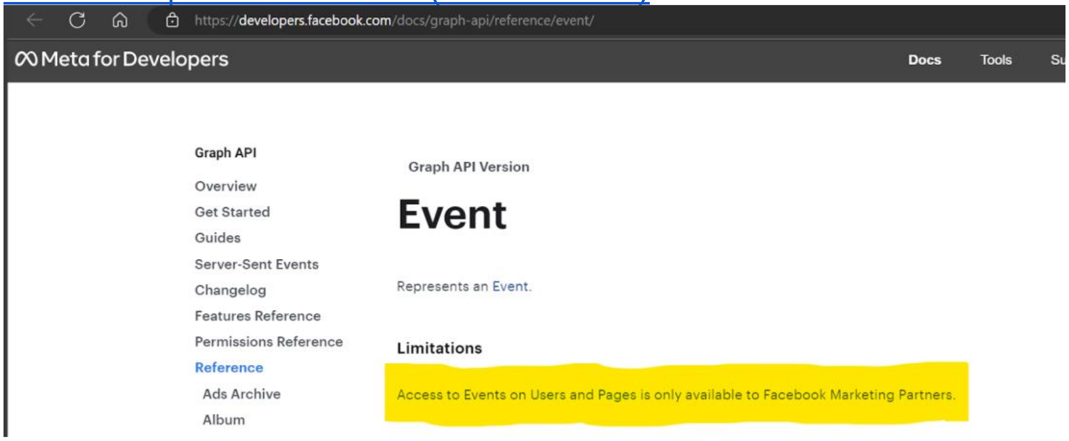
Are you facing any challenges?

1. Filtering through noise data
2. Some users post details of events in image instead of text.



have 2 Chicago Bears tickets for this Sundav's game vs the Detroit Lions for sale. United club Sec 146, Row 16, Seats 13-18
Dm if interested.

3. **Not all websites allow the scraping of data:**
   Some websites do allow the scraping of data, but many others do not. A few websites allow access. In this case, using API might be your only option. A good example is Facebook.
4. Access to a lot of the facebook events were not available publicly available:
5. Event - Graph API Reference (facebook.com)



Event

Represents an Event.

Limitations

Access to Events on Users and Pages is only available to Facebook Marketing Partners.

6. Evaluating if data collected is relevant. Based on feedback provided by the TA in our project proposal, we realized that explicit feedback will not be feasible as we won't have too many users so we had to think of different implicit feedback ways