

Final Project Proposal in Visualization Techniques

“Life in a Day at Facebook”

Nitish Dhakal

11/07/2016

Introduction

There has been significant rise in number of users in social networks over the time. These users have different types of interactions in the social network. If we consider Facebook, likes, comments, posts are most common modes of the interactions. In twitter people tweet, retweet, follow, hashtag etc. There is no doubt that people spend quite significant amount of their time per day in these social networks.

So we thought of visualizing the time spend by the users in social network in terms of interactions. This project would be titled “**Life in a day at Facebook**” or **LIADAF**. The main goal of the project would be to highlight the interaction traffic in Facebook within a day through visual analytics. We would highlight 6 time frames within a day namely, midnight to 4 am, 4 am to 8 am, 8 am to 12 pm, 12 pm to 16 pm, 16 pm to 20 pm and finally 20 pm to midnight.

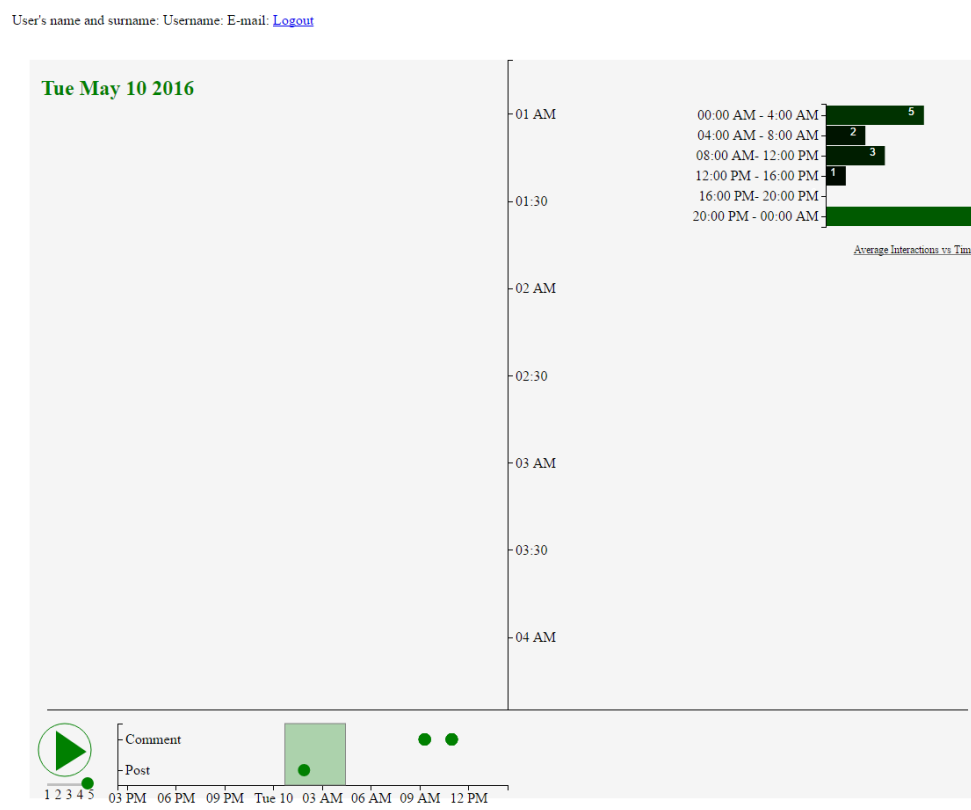


Figure 1: Preliminary design of Life in a Day at Facebook

We would focus our visualization in Facebook and if time permits we would like to extend the same in Twitter as well. Figure 1 shows the sample made for **LIADAF**. We would target our application to the Facebook and Twitter users (16-32 years) who would like to view and perhaps decide how have they usually spend their time in social medias.

Objective

Our main objectives are:

1. To find out how user spend their time in Facebook or Twitter daily
2. To find out the average latency in interactions done in user Facebook or Twitter page.
3. To view the activities in Facebook or Twitter in a moving timeline at our own speed.

Methodology

To do this project we will use an MVC based python web framework called “Django”. We will use Mongo dB as our backend and Django REST framework for creating our APIs. The server side would use **MVC pattern** while the client side scripting would leverage the closure and **singleton pattern** as a design pattern. The codes would be open source and would follow the necessary coding standards and put into GitHub. Figure 2 shows the general application flow of Life in a day at Facebook.

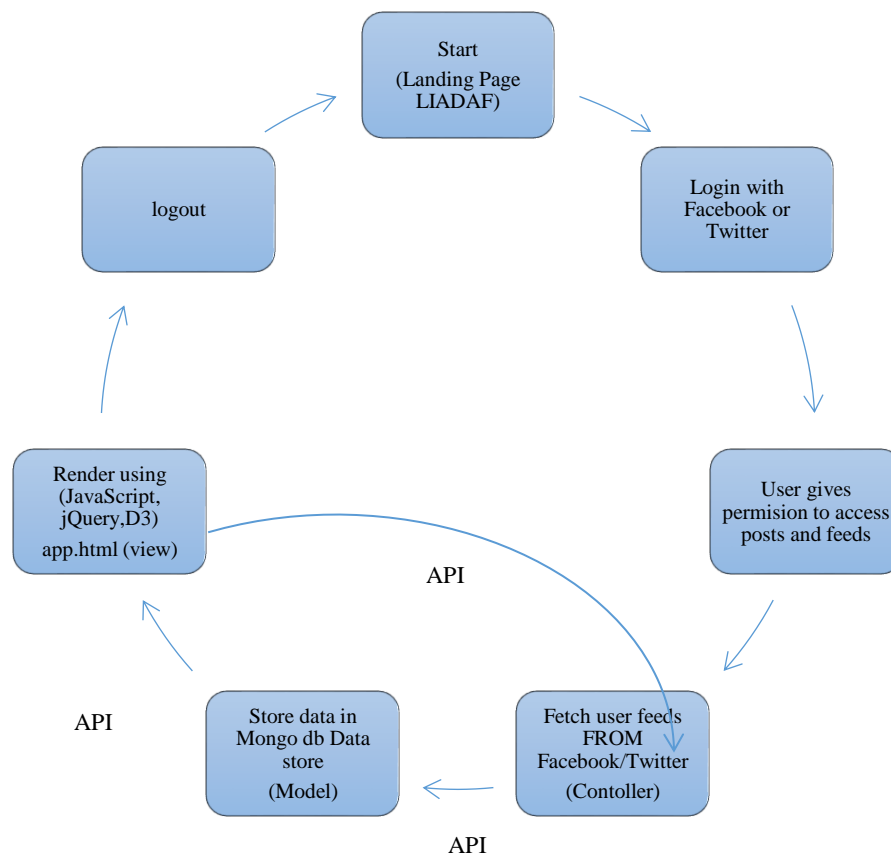


Figure 2: Application framework of “Life in a Day at Facebook”

Application flow

1. User goes into our web page and chose whether he wants to login through Facebook or Twitter
2. After login in S/he would have to give the necessary credentials and “use post” permission in Facebook or Twitter. This is the necessary step to produce data. If user does not provide the access to the user feed, then the application would not work.
3. After step 2, the application would fetch the 2 month old profile information of the user and save it in Mongo DB data store and format the data and send it to the view to render.
4. The view would comprise of the app.html file, which would use the native JavaScript, d3 framework and jQuery to render the visualization.
5. The visualizer would have 2 different chart components, main Chart and navigation chart. Main chart would contain the vertical moving time scale along with the beautiful components showing the user post and comments information over time. The navigation chart would provide the same information in a smaller chart but with panning and zoom facility. There would also be play, pause and forward button to provide feature for user interactions with the chart. There would also be a feature to select the data within the particular range. If user selects the data in different date range, then the API would call the Facebook or Twitter server for additional information and again store data in Mongo Db data store and then send the data to the view to render.
6. There would be additional components in the visualizer which would show more depth about the interactions.

Features

1. Main Moving Chart
This would be the heart of the application as well as the main area of user focus. This would contain the vertical time line with user posts and comments on either side of the timeline. The user post and comment would appear along the selected date range.
2. Play, Pause and Forward
This would be another addition to the main moving chart which would provide dynamism with user control. User can play pause or forward the moving chart with the slider button.
3. Navigation Chart
This sub chart allows the user to navigate across the timeline in horizontal fashion. This chart also allows the user to know the instance of posts and comments which appears sequentially according to the date of creation. Navigation chart also allows brushing selection across time as well as scaling and zooming options

4. Calendar Widget

User can load arbitrary historic ranges of Facebook or Twitter data and view their interactions across the main chart and navigation chart. To load these data, user need to be logged in to either Facebook or Twitter.

5. Interaction Widget across different timeframes

This widget provides the percentage of time spend by the user in Facebook or Twitter through the dynamic bar chart. The time frame here is divided into 6 different partitions as mentioned in the introduction section.

6. Interaction Instance Widget

This widget is essential in giving user high level knowledge about the data. The main chart and navigation chart has to be moved until end to see all the occurring of interaction data. However, with this widget, user can view the interaction instances between the selected date range. There would also be a horizontal moving bar to show help make user better understand the data occurrence.