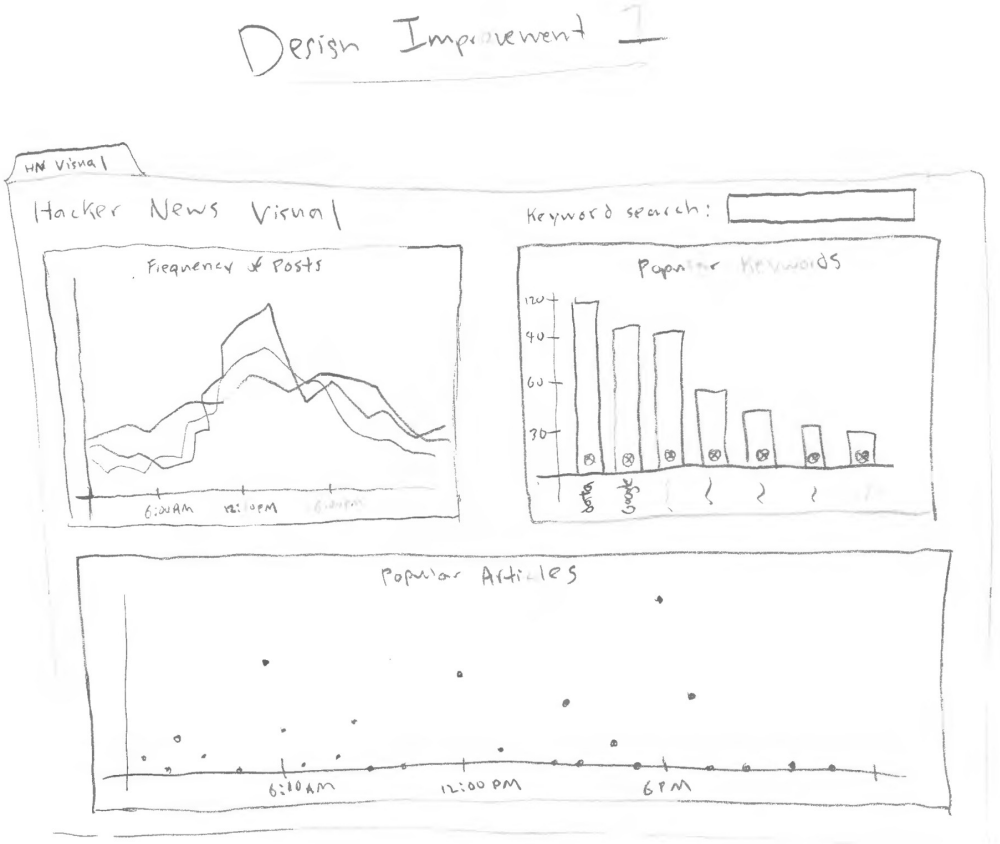
Preparation for Project 3

Scope of Improvement

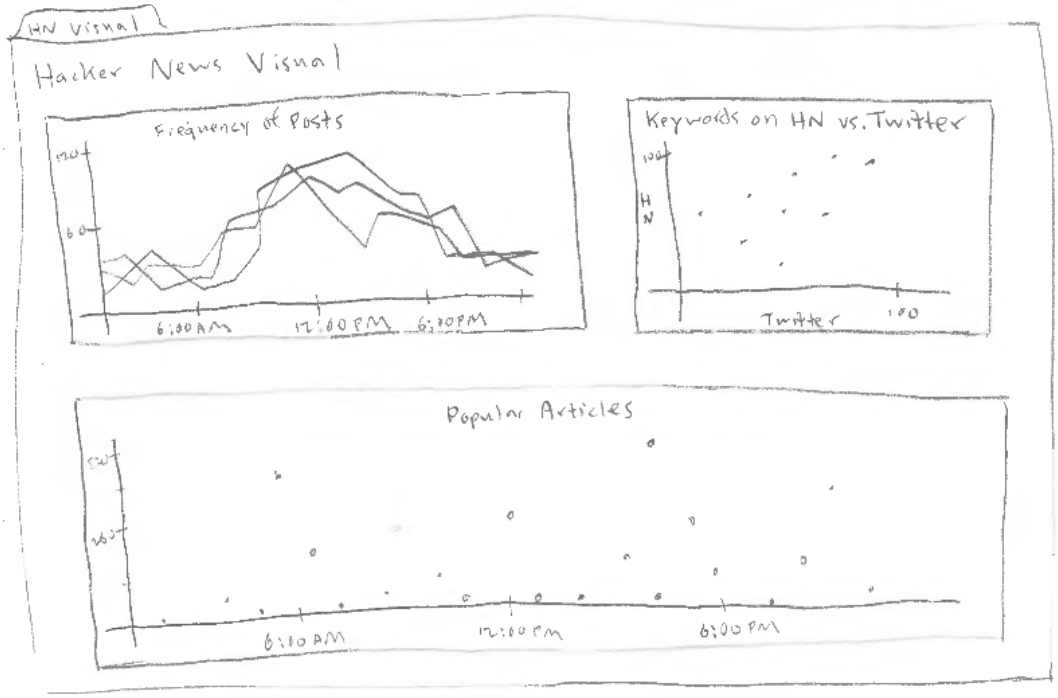
In addition to the bugs that have been fixed from project 2, we also want to make the visualization more cohesive. The links and connections between the different graphs are not immediately apparent. We want to organize our visualization so that it is visually appealing, but also so that it is easier to understand.

1. Thus our first idea for design improvements largely focuses on the layout of the design. We want to avoid switching back and forth between the two visualizations (line graph and scatter plot) so we can make a dashboard that would include both on one page as shown below. We can fit all of the elements on one page by making both the line graph and bar charts smaller. The line graph can have a smaller width without losing much and the bar chart can be smaller by simply choosing to display fewer keywords (e.g. only 10 instead of 20). We also envision implementing a search feature where you could search for a specific keyword and we would only display the articles in the scatterplot that contain this keyword. We also thought about adding in the functionality of giving the user the ability to interactively remove keywords from the bar chart. This would add them to our “stop” list so the word would never be shown again.



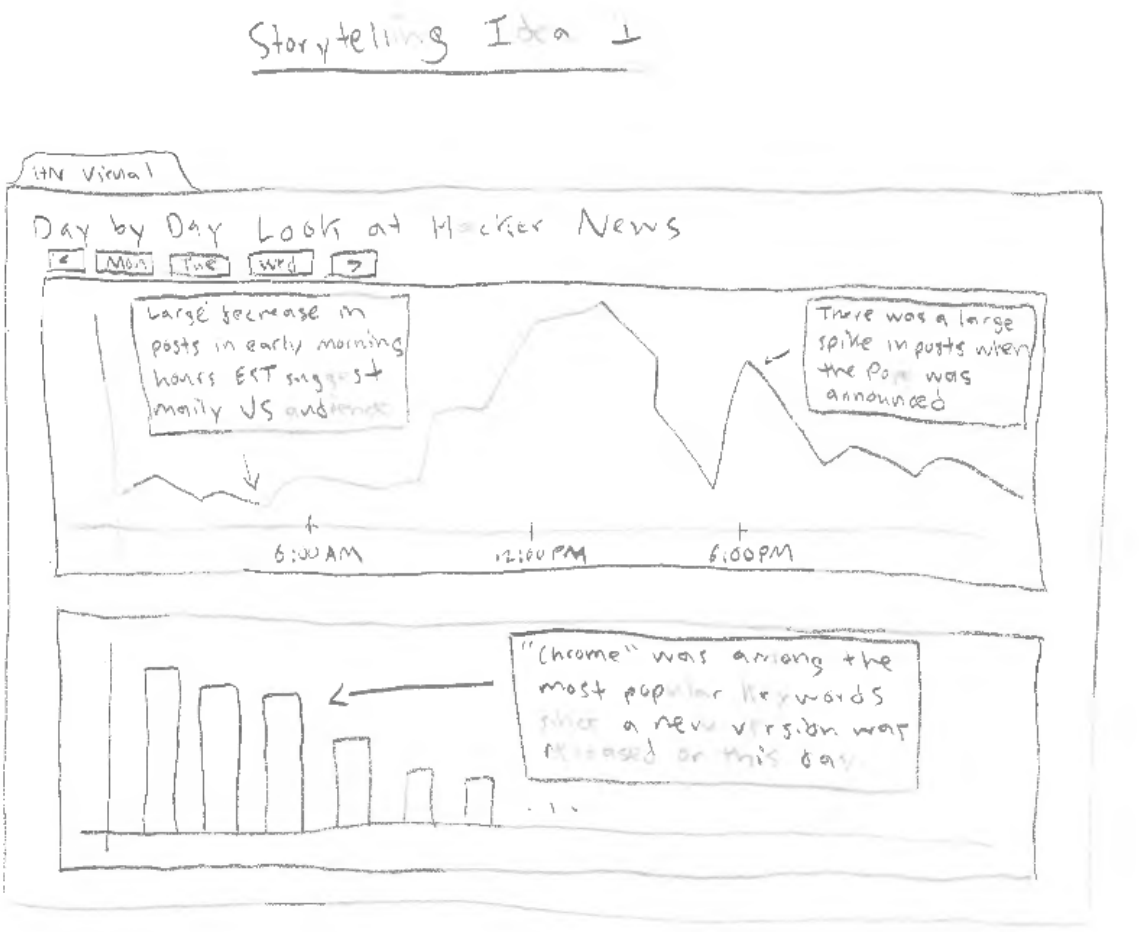
2. The second design improvement idea is similar but also incorporates a new feature by expanding our dataset. We think it might be interesting to compare the keywords we see on Hacker News to those of another source such as Twitter. Thus, we want to include a scatter plot that compares popularity on HN to popularity on Twitter. In this one, we decided to give the line graph a little more width, since the Twitter scatterplot would ideally be more of a square to prevent any bias.

Design Improvement 2

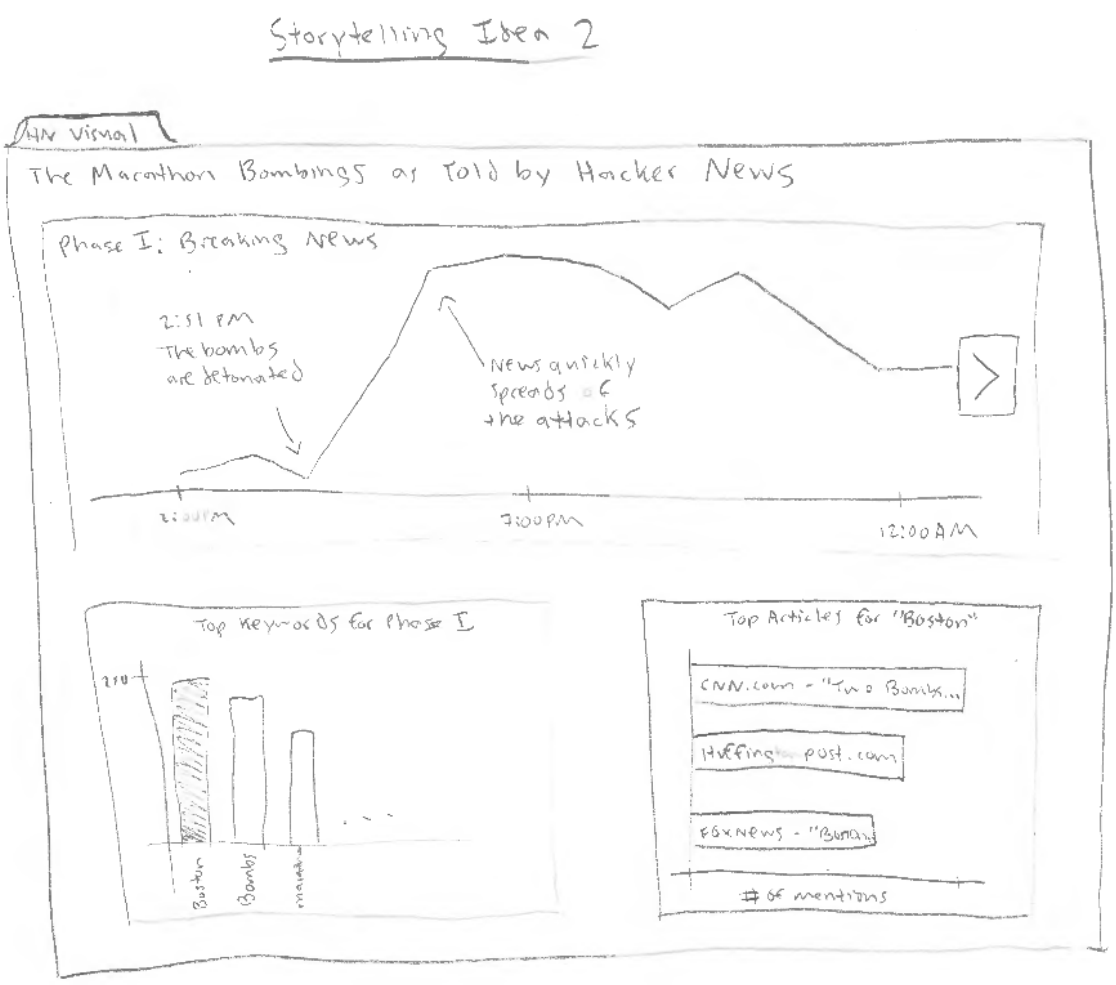


Scope of Story Telling

1. The first idea for storytelling would be a multi-step visualization (as used in Olympic Stars). This visualization would step through each day, annotating interesting finds for that day. For example, if we notice a sharp spike in the number of posts for a given hour we could explain why that was. Similarly, if there is a word such as “pope” that starts to appear frequently in articles, we could explain that this was because the new Pope was chosen on that day. After cycling through each day, the storyline of the week so to speak, you would arrive at the fully interactive visualization as depicted above.



2. A second idea of storytelling focuses on one specific event, The Marathon Bombings in Boston, and seeks to tell the story of that through Hacker News. We would selectively only include stories with keywords relating to Monday's events and tell the story that way. We would need to re-work our scraper so that it could collect data from Monday. I think we would break the story up into three phases and analyze what happened during that phase from a news perspective. For example, I imagine the first phase being the "breaking news" phase where the events were still transpiring and the story was developing. Next probably comes the outpouring of support for the victims and everyone involved. We would have to figure out the third phase by looking at the data, looking at the tendencies in the popular keywords in the articles.



Proposed Implementation Plan

I don't think it will be necessary to change technologies very much, but it depends on the design we end up choosing. If we decide to incorporate Twitter mentions, we would most likely use the Pattern API to grab that data from Twitter, but we would likely limit it to the top 10 keywords from each day. The Pros of this is that we are already familiar with Pattern and we are already using Python to scrape our data. Cons are that there are potentially better APIs out there, but this suits our needs.

The other first thing we would have to do, especially if we intend to do the Boston Marathon story line is to change our scraper so that we can collect data from that far back. Previously we had to run the scraper every 24 hours to get all the data on Hacker News but it has been more than 24 hours since the events on Monday.

Simultaneously, we could start looking at the data to figure out why certain things might be

happening that we could annotate for the story lines.

Timeline of Project Completion

4/18 – Update scraper and collect data if we are doing Marathon story

4/22 – Complete dashboard layout of visualization, add Twitter search to scraper, continually run scraper this week to collect new data

4/25 –Add search feature of keywords, add ability to remove keywords from results

4/28 – Complete multi-step layout with visualization at end and analyze data to add annotations to storyline

5/1 – Finalize viz, make it look pretty, make sure process book is in shape, create video, submit

Summary of TF Meeting

After meeting with Sofia, we discussed many things about improving our project two. We realized that we need a cleaner design and a better picture of how everything links together. We also discussed the need for more brushing where we should be able to select a sub-set of data and link that new selection in another view. We also discussed the possibility of helping our storytelling with a sort of animation in the beginning of our visualization. Based on this feedback, we have a clear idea of what we want to focus on.

Final Design

Our new design (sketched below) introduces the new element of a streamgraph. The design is more focused around keywords now and the graphs will be linked and brushed with respect to these keywords. At the top, you will be able to select one of our pre-curated “storylines” based on keywords, which you will be able to explore through the different graphs. For example, one of our main storylines will be the Boston Marathon bombings and ensuing events. We will select the top keywords from these articles so when you click on the storyline you will be able to see the top keywords based on frequency in the bar graph in the top left. On the top right will be the scatterplot of the articles relating to these keywords based on time they were published and their popularity based on number of upvotes/points they received. If you select one of the keywords in the bar graph, the scatterplot will only display articles that have that keyword (fulfilling brushing/linking). The streamgraph on the bottom, will display the popularity of keywords, as they change over time. This should give us a good idea of when certain keywords are popular. We can annotate these observations to help with our storyline element.

We have already gathered the necessary data for this project by updating our scraper and running it. Other than that, we will use D3 to implement the streamgraph (example here). We also plan to implement a search bar, where you can search for specific keywords and the graphs will update accordingly.

If we have extra time, we will implement a new data structure for our backend with the goal of making our page run faster.