Hi Rob,

Unfortunately we were not able to get a polished working web app completely up and running by the deadline. I'm sorry to disappoint you Rob, but this has been a time of great transition for most of us. Three of us just got new jobs or were accepted into coding bootcamps within the last few weeks. I am currently in the middle of a very intensive and selective interview process for a position that I wasn't expecting but just kind of fell in my lap, so I will need to devote my attention to that fully over the next few weeks.

I was trying to use lambda to do the scheduling instead of Celery for a while before I have up, so lost quite a bit of time on that too. We ended up spending a lot of time doing research on all the different AWS, data pipeline and scheduling options out there, and I think we felt a bit paralyzed by so much choice.

These are the main next steps that remain to be completed:

1.Still need to get the scheduler to run while pulling in the tweets and put the real tweets themselves in the dynamodb (adding them together from search function, not from stream function because that was turning out to be much more difficult and didn't have time)

2.Put in the code to update the tables and add new tweet data to the old data without losing the old data

3a.Get the states parsed out of the location, either using querying in dynamodb, or using parsing code( I have some I wrote previously for the CSV file)

3b. Parsed out by keyword (can probably be done on dynamodb)

3c. Indexed (numbered), to match the earlier csv file

4. Figure out how to get keyword querying on website so users can choose which keywords they want (which will have already been collected, be pulled from the database and put through Vlad’s dataframe code somehow?)

5. Get this all configured within Django, using one coherent AWS account, email etc.

Some things that we did manage to do:

* Get a website up and running (thanks to Celestino)
* Create a dataflow diagram for the project, and a balsamiq mockup for the web app layout
* Pull in tweets on tweepy using both a stream and search API (We switched over to the search API later because it proved much simpler to use w/ Celery task scheduling)
* Parse out tweets by keyword, location and time (using python script in a csv file)
* Analyze tweets by keyword and state and output that analysis into dataframe charts
* Get a task scheduler running on Celery, pulling in tweets every 20 seconds
* Create a dynamodb table, inserting the tweets pulled in from Celery into the table (although the table is still not completely finished.

I think towards the end we have been getting a good idea of how things should fit together, a lot of the time we put into this project went towards the research part. It’s mostly a matter of tying it all together now and polishing it up, putting it all in Django, testing it etc.

We understand that the code is not necessarily up to an advanced level, in that it has not been tested, properly made use of modules and classes etc. Again, I think that we just did not have the time. We really did make an effort, but there were so many different parts to fit together, learning all about AWS, mockups etc. for the first time. As I mentioned earlier so many of us were unable to devote the time we would have liked to due to starting or interviewing for new positions.

We won’t be able to get a polished product to you in time for grading, but I know that for me at least, this is definitely something I would like to finish over the next couple months if possible, both to have for my portfolio and just because I find it an interesting, challenging project to work on.

Thanks Rob,

Kelly (and the team)