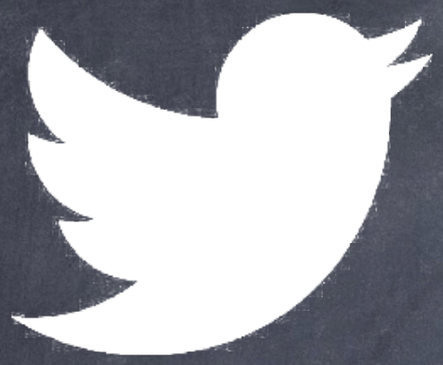


Milestone 1

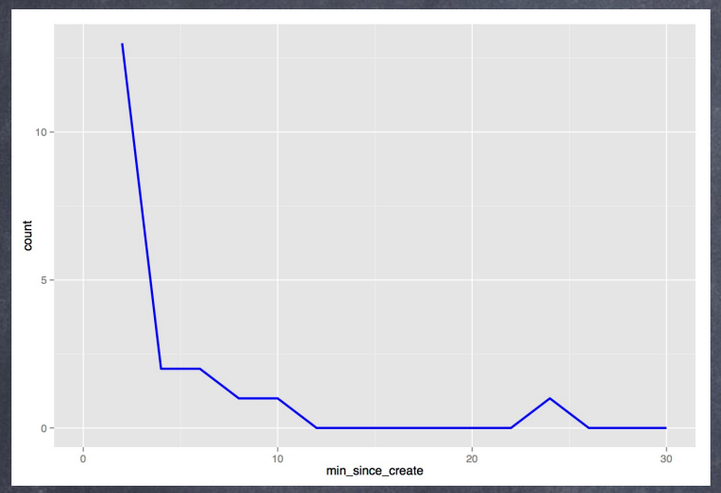
Tweet Lifecycle Analysis

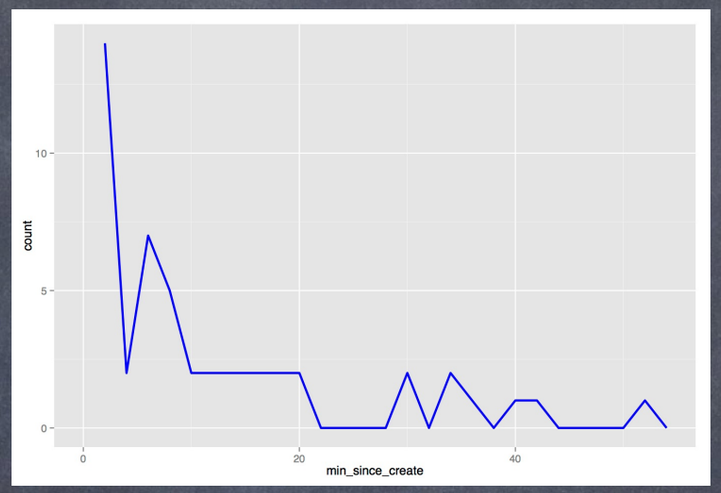
Presented by: Daiwei Lu, Yuan Gao

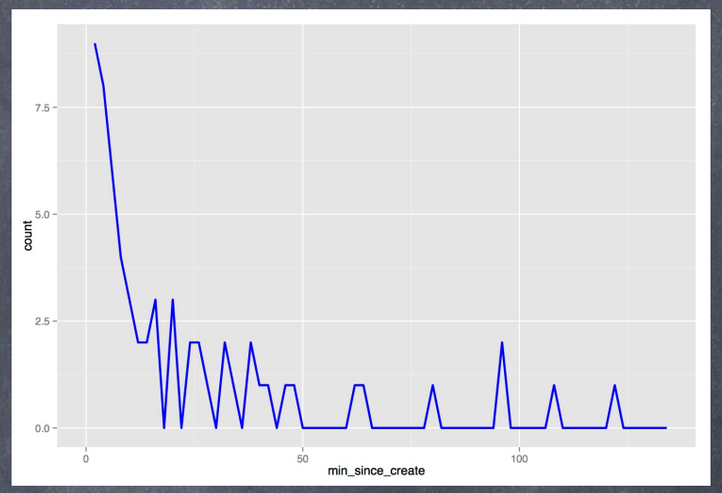
Research Question

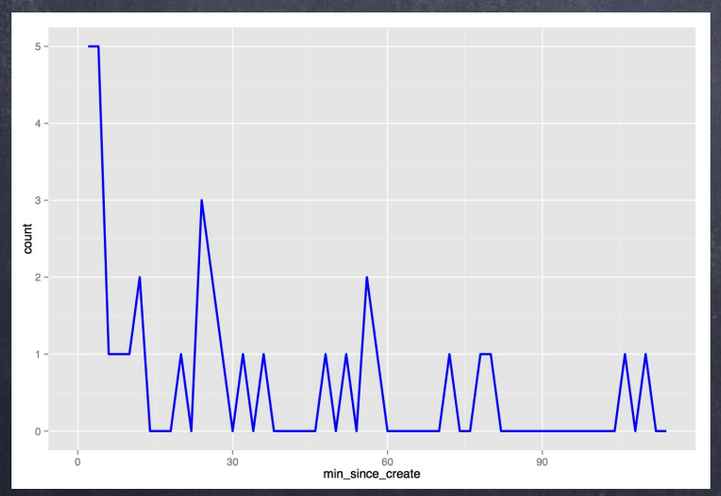
- What is the life cycle of a tweet? And the timing of each event in such life cycle?

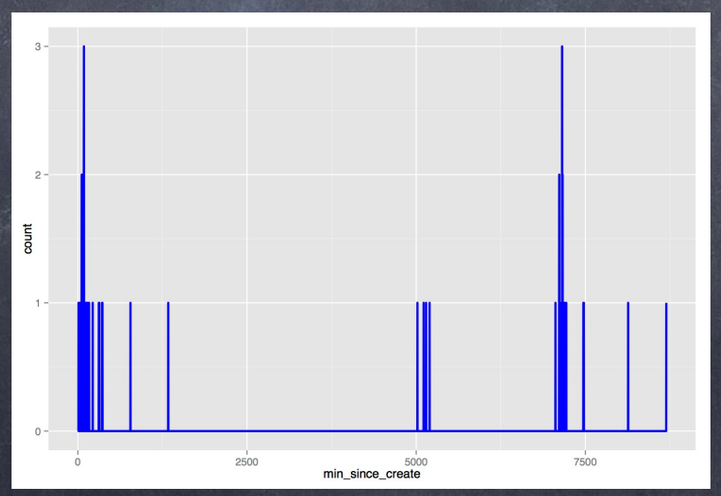
Preliminary Analysis

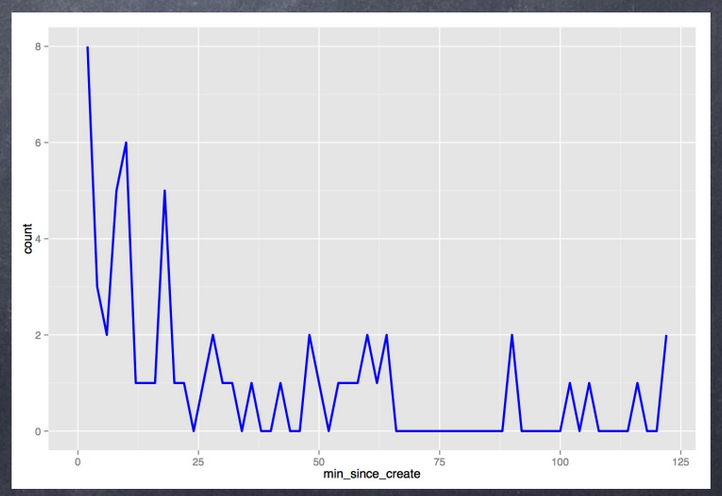




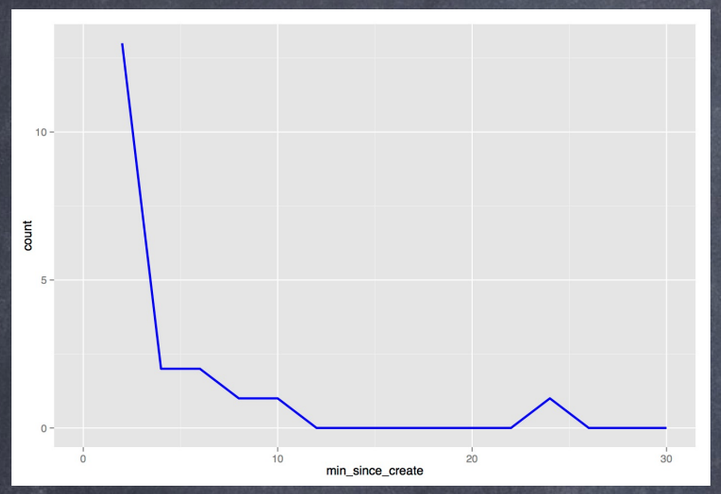


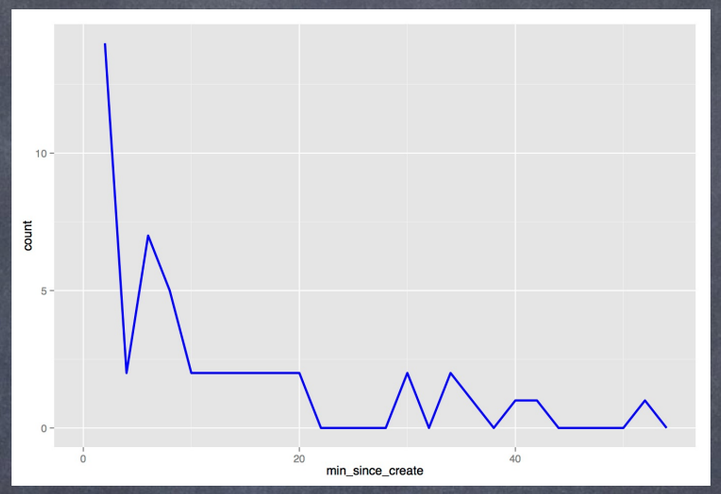


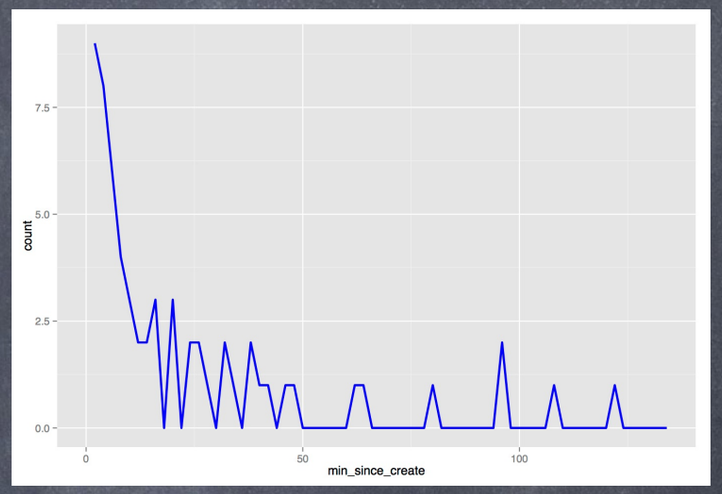


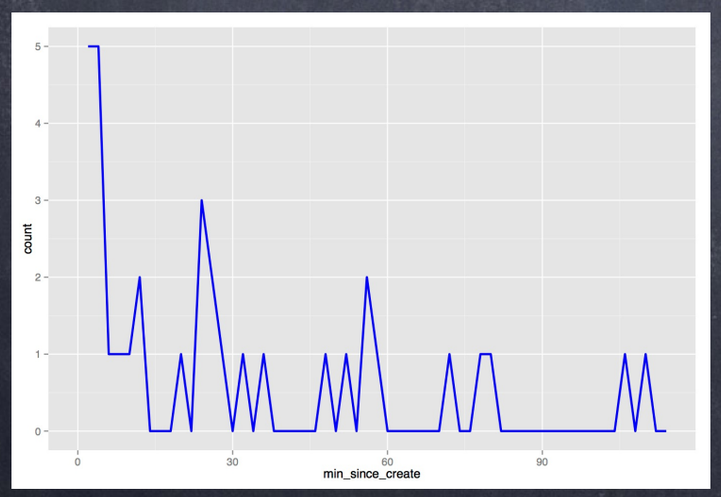


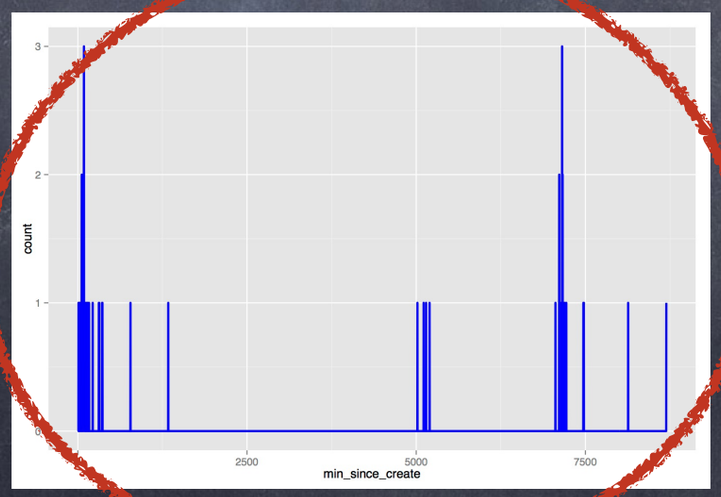
Preliminary Analysis

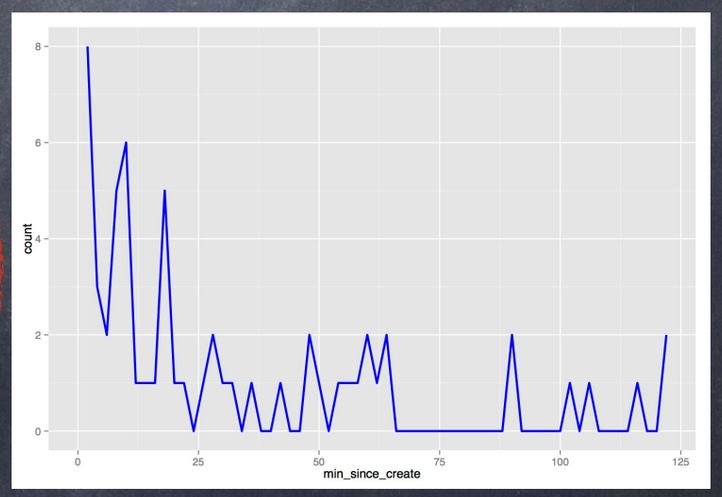




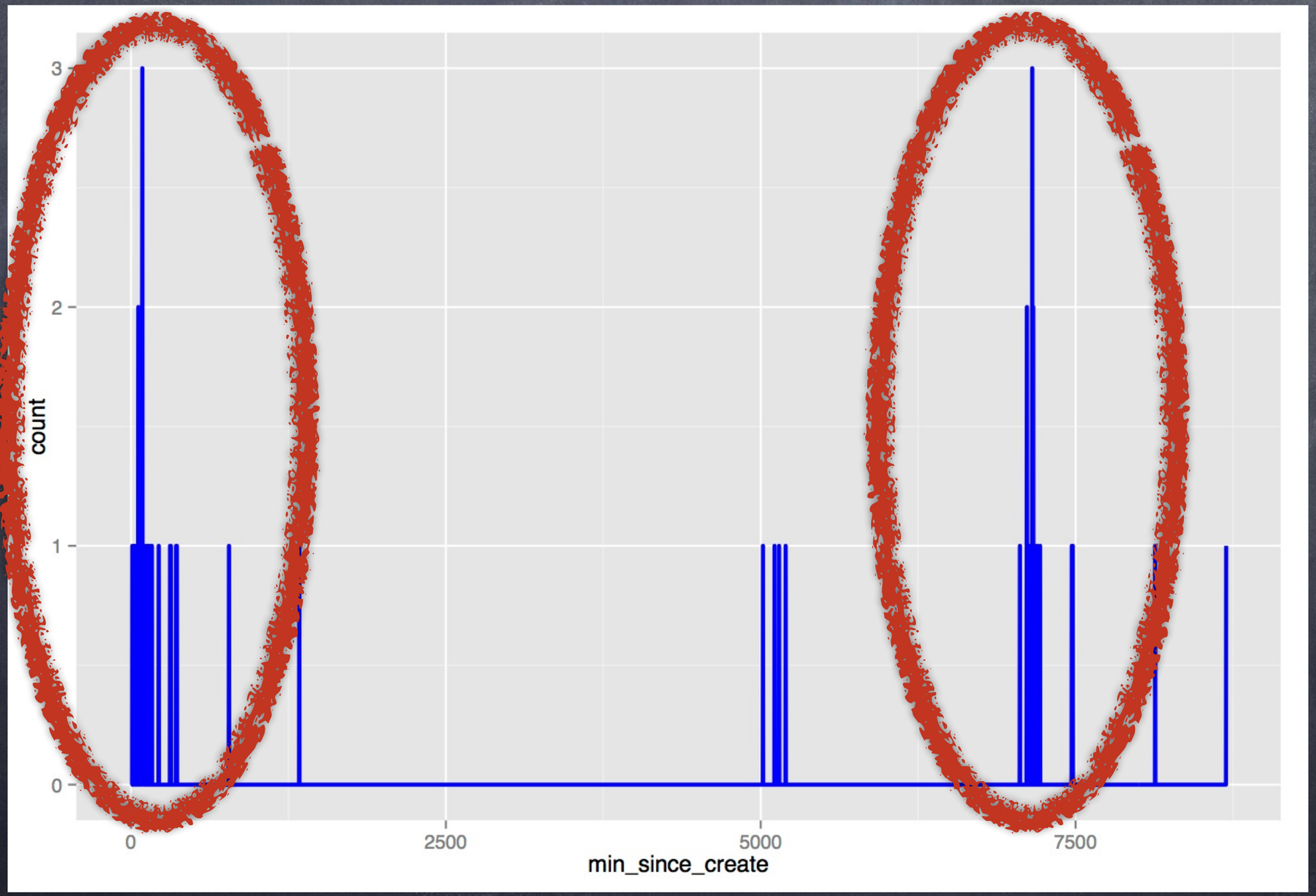








Preliminary Analysis



Review of the State of the Art

- Information Contagion: An Empirical Study of the Spread of News on Digg and Twitter (by Kristina Lerman and Rumi Ghosh)
- Want to be Retweet? Large Scale Analytics on Factors Impacting Retweet in Twitter (by Bongwon Suh, Lichan Hong, Peter Pirolli and Ed H. Chi)
- Predicting Information Spreading in Twitter (by Tauhid R. Zaman, Ralf Herbrich, Jurgen van Gael and David Stern)
- The Pulse of News in Social Media: Forecasting Popularity (by Roja Bandari, Sitaram Asur and Bernardo A. Huberman)

Dataset Collection & Cleansing

```ruby
require 'twitter'
require 'csv'
require 'time'

# Config a Twitter client
$client = Twitter::REST::Client.new do |config|
  config.consumer_key    = 'KaP6dQMgDQypv7fVvQ7Q'
  config.consumer_secret = 'Spy6Ei4BVttpSJjYMTubto79xV9Sfo1H1E843xpP8'
end

# Target tweets' id that we want to analysis retweets on
targets = [517101582534852608, 517101577631694848, 517101563438178305,
  517101561269723137, 517101546111533056, 517101437713932289]

targets.each do |id|
  # Get original tweets
  tweet = $client.status(id)

  # Get headers for CSV output
  headers = tweet.attrs.map {|k, v| k}

  # Combine original tweets with all the retweet tweet into one array
  # sort them based on created time
  arr = ([tweet] + $client.retweets(id, count: 100)).sort do |a, b|
    Time.parse(a.attrs[:created_at]) <=> Time.parse(b.attrs[:created_at])
  end

  # Create a new CSV file and write all records
  CSV.open("./retweets_of_#{id}_new.csv", 'w') do |csv|
    csv << headers # Output header of CSV
    arr.each do |tweet|
      csv << tweet.attrs
        .select {|key, val| headers.include? key} # Only output a value with its key in headers defined earlier
        .map {|key, val| val.class == Hash ? nil : val.to_s} # Ignore values with nested attributes, we don't need them for now
    end
  end
end
```

Other Contributions

Use #Hashtag

Use popular tweets

Using streaming API to collect data

Questions?

Thanks