

# Application Programming Interfaces (APIs)

Yotam Shmargad  
Assistant Professor  
University of Arizona

As Deen Freelon said in his recent 2018 SICSS talk, we're quickly entering a "post-API age"

## Data access in the API age vs. post-API age

API age	post-API age
Reliable (or so we thought)	Tenuous
Broad	Circumscribed
Free/cheap	Expensive
Open access on demand	Subject to approval
TOS-compliant	Not TOS-compliant
Available	Gone

Full video: <https://www.youtube.com/watch?v=uHSCRDoJ0yM>

In the past, Twitter allowed anyone to use its API – recently, they added a developer approval process

**Twitter API**   
@TwitterAPI

Following 

As a developer platform, our first responsibility is our users: to provide a place that supports the health of conversation on Twitter.

To continue to prevent misuse of our platform, we are introducing a few new requirements for developers today.

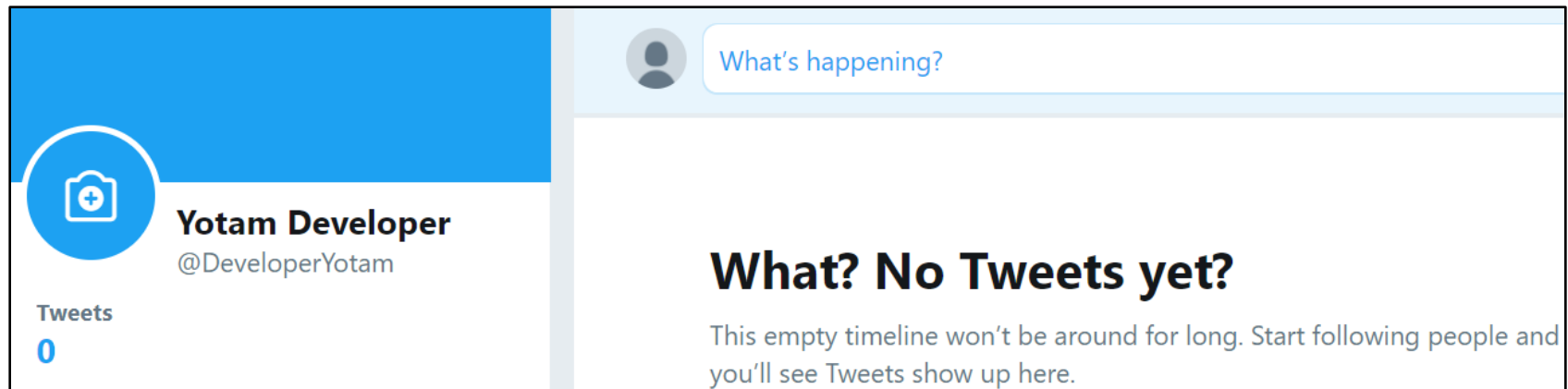


**New developer requirements to protect our platform**  
Recognizing the challenges facing Twitter and the public, we're taking steps to ensure that our developer platform works in service of the overall health of conversation...  
[blog.twitter.com](https://blog.twitter.com)

2:02 PM - 24 Jul 2018

To understand what it's like to apply for API access as a new developer, I created a new Twitter account

[www.twitter.com](https://www.twitter.com)



[apps.twitter.com](https://apps.twitter.com)

## Twitter Apps

As of July 2018, you must [apply for a Twitter developer account](#) and be approved before you may create new apps. Once approved, you will be able to create new apps from [developer.twitter.com](#).

For the near future, you can continue to manage existing apps here on [apps.twitter.com](#). However, we will soon retire this site and consolidate all developer tools, API access, and app management within the developer portal at [developer.twitter.com](#). You will be able to access and manage existing apps through that portal when we retire this site.

[Apply for a developer account](#)

# I went through Twitter's significantly more involved process a week ago and have still not heard back...

STATUS: IN PROGRESS


☒ User profile

☒ Account details

☒ **Use case details**

☒ Terms of service

☒ Email verification

 **Why the questions?**

We empower freedom of expression by providing a platform that protects the voices of our users — both on Twitter, and via our developer products. To help verify that all uses of Twitter data comply with our policies, we require additional information from developers signing up to use this service. Providing thorough answers will help us understand your use cases and will help expedite the evaluation of your application. Learn more about our [restricted use cases](#).

## Tell us about your project

**What use case(s) are you interested in?**

Select all that apply

☒ Academic research

☐ Advertising

☐ Audience analysis

☐ Chatbots and automation

☐ Consumer / end-user experience

☐ Engagement and customer service

☐ Publish and curate Tweets

☐ Student project / Learning to code

☐ Topic analysis

☐ Trend and event detection

☐ Other

**Describe in your own words what you are building**

In English, please describe your product - the more detailed the response, the easier it is to review and approve. Be sure to answer the following:

- What is the purpose of your product or service?
- What will you deliver to your users/customers?
- How do you intend to analyze Tweets, Twitter users, or their content?
- How is Twitter data displayed to users of your end product or service (e.g. will Tweets and content be displayed at row level or in aggregate)?

I will be teaching participants of the Summer Institute in Computational Social Science at the University of Colorado,

?

I encourage you all to go through Twitter's process if you do not already have a developer account

Steps:

- 1) **Go to [www.twitter.com](https://www.twitter.com) and sign up for a Twitter account:** add a phone # and verify your email address
- 2) **Go to [apps.twitter.com](https://apps.twitter.com) and apply for a developer account:** check 'Student project/learning to code'
- 3) **Wait for approval...**
- 4) **Register a new app:** Chris Bail has a good tutorial [https://compsocialscience.github.io/summer-institute/2018/materials/day2-digital-trace-data/apis/rmarkdown/SICSS APIs markdown.html](https://compsocialscience.github.io/summer-institute/2018/materials/day2-digital-trace-data/apis/rmarkdown/SICSS_APIs_markdown.html)

Twitter's new approval process has made it difficult to teach students how to use the API in *real time*

## Alternatives

- 1) Have students go through Twitter's new developer application process well ahead of time
  - Unfortunately, there was not enough time to have Boulder SICSS participants (you!) go through the approval process
- 2) Teach students how to use Twitter's API using files that are generated ahead of time (through an existing developer account)
  - This is the alternative that I will attempt today

I'll first discuss a couple of technical issues when using APIs, then a specific application with Twitter

## The plan

- 1) Discuss a couple of technical issues when using APIs
- 2) Talk about the details of our specific application:  
Constructing Ego Networks from Retweets
- 3) Go through the code!
  - Those who already have a developer account can follow along using Twitter's API
  - Those without a developer account can download files from my Github repository: [github.com/thesickish/BoulderSICSS](https://github.com/thesickish/BoulderSICSS)



# There are two issues that often come up when using APIs

## Issues and solutions

- 1) Companies often impose limits on how much data one can collect from their servers in a single session
  - On Twitter, these limits refresh every 15 minutes
  - Tell python to sleep for 15 minutes if a limit is reached
- 2) Errors can arise when content or user accounts have been deleted
  - This can disrupt our code so that python stops working
  - Tell python to skip content/users that no longer exist

**Next: Constructing Ego Networks from Retweets**

# Twitter has become a central platform for anyone hoping to get information out to the public

## A tweet



**Retweets =**  
# of times  
tweet was  
shared  
*directly*

We can get a sense of how much exposure tweets are receiving by analyzing their **retweets**

### Advantages of analyzing retweets

- 1) **Measure dissemination:** one user's decision to share another user's message
- 2) **Timely:** since retweets are tied to specific posts, we can analyze a time period of our choosing
- 3) **Relatively rare:** users tend to have fewer retweets than followers, so we can analyze popular accounts
- 4) **Generate networks:** We can know if two retweeters of a user's tweets retweet each other

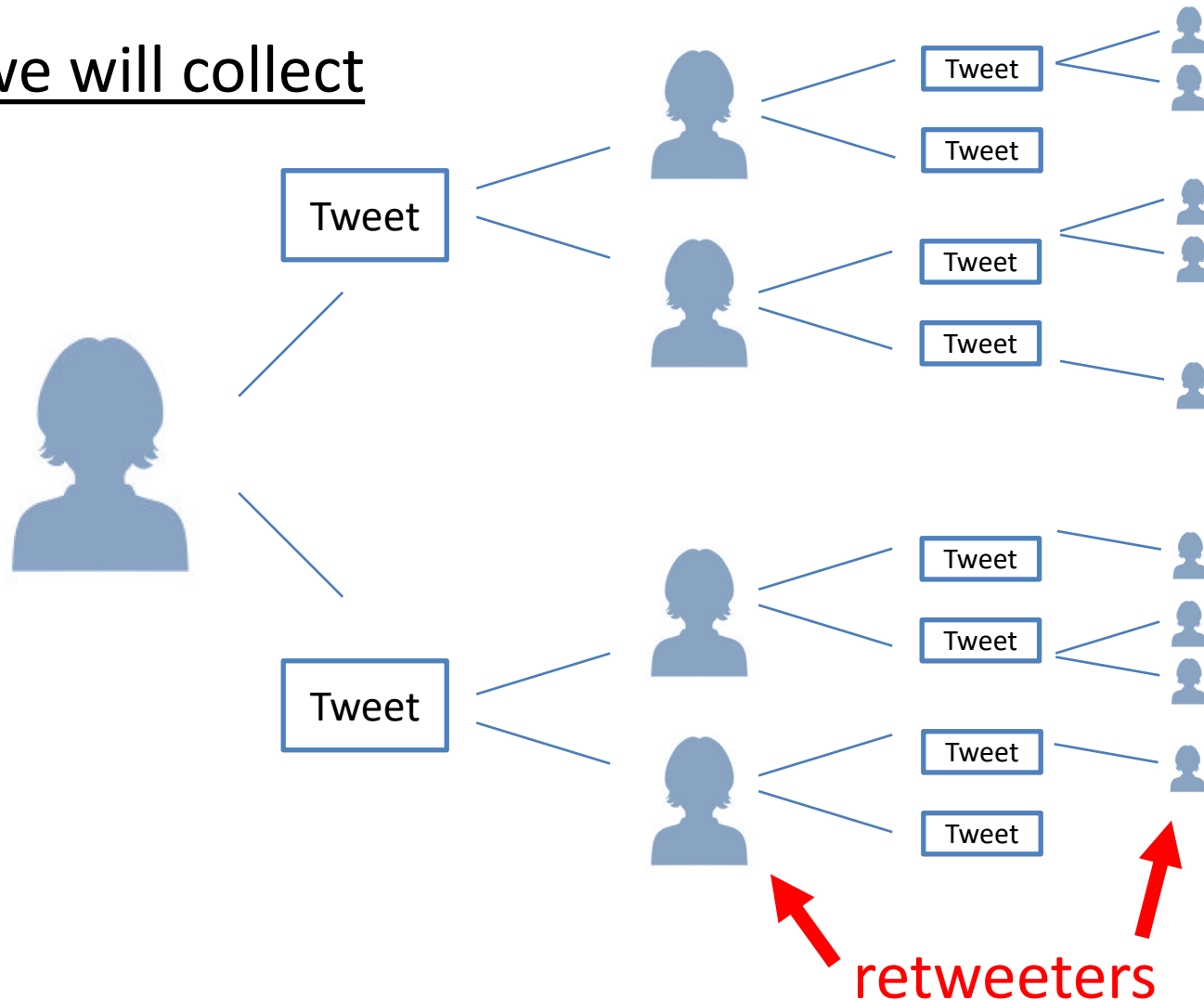
# Today, you will learn how to collect and organize Twitter data to build a network from retweets

## Outline

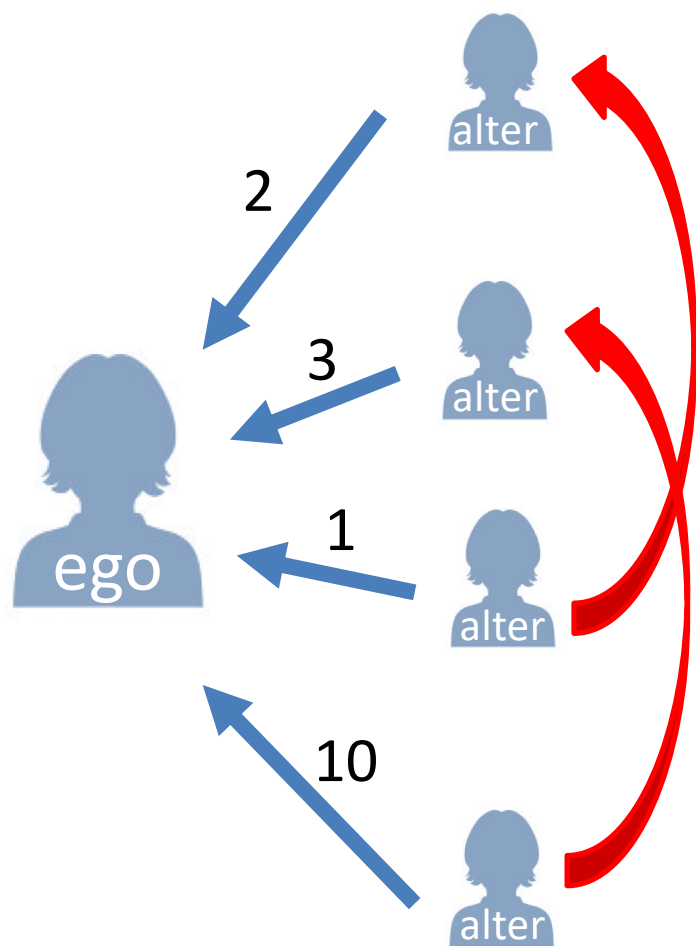
- ~~1) Create a Twitter developer account~~
- 2) Use the Tweepy package (in python) to collect tweet and retweeter data
- 3) Organize data and build a social network using the networkx package (in python)
- 4) Visualize the network

We will first collect an account's tweets, retweeters, their tweets, and their retweeters

Data we will collect



We will then use the data to build an **ego network**, where links capture retweets between users



Links in this network have **weights** – the number of one user's tweets that another user has retweeted

Since we know if retweeters retweeted each other's tweets, we can calculate measures of local **redundancy (or density)**

# I used this method to analyze Twitter accounts of candidates in the 2016 U.S. congressional elections

## Final dataset

- 1) Accounts of more than 400 candidates
- 2) Over 25,000 candidate tweets
- 3) Over 190,000 retweeters of candidate tweets
- 4) Almost 900,000 retweeter tweets
- 5) About 1.3 million retweeter retweeters
- ...
- 6) Campaign donations and election outcomes

(more at my talk this afternoon!)