CS S-109A
Protopapas, Rader

Scope of Work for Data Science Final Project
Milestone #2

Prepared by Project Team #6
Eumar Assis
Andrew Caide
Mark Carlebach
Jiang Yusheng

Project Statement and Background

Most people who use twitter are aware of the possibility that tweets received are tweets generated by computer algorithms or 'bots'. There are concerns broadly that bots can cause societal damage by propagating 'fake news' that can influence people in a number of ways. The most prominent potential impact of 'fake news' is on how recipients of fake news vote in elections, both in the US and abroad.

The business risk of the above to Twitter is manifold:

- Regulatory risk if Twitter is not seen as having sufficient controls to mitigate fake news propagation.
- Reputational risk with advertisers who might not trust the size of Twitter traffic and Twitter user base.
- Reputational risk with the public who won't view Twitter as a reliable source of news and information.
- Financial risk to shareholders if stock price declines due to the above concerns.

One of Twitter's two strategies for dealing with this problem is 1) to grant 'verified' user status to authors of tweets who are known to twitter and 2) to use machine learning to detect suspicious accounts.

Our project fits in this context and has the **following goals**:

- We intend to answer the question whether the author of tweet is a human being or a computer bot. We will create a function that takes in a tweet and returns a 1-10 score showing the likelihood the author of the tweet is a human being or a bot.

- The score will be developed using input from both a classification algorithm and an NLP based algorithm.
    - The features of the classification algorithm are discussed below.
    - As for the NLP algorithm, we will analyze the textual component of the user's tweet history. This NLP analysis will yield additional attributes for each tweet that we can then use the classification algorithm. Textual complexity, textual style, sentiment, consistency, etc. could all contribute to helping identify human or automated authorship.
- We will show our scoring results and error rates for known real people (i.e., verified users on Twitter). We will also show our scoring results and error rates for known bots that we will find from a reliable sources online.

Literature Review

- Papers and References
    - http://www.icir.org/vern/papers/pam11.autotwit.pdf
    - https://theintercept.com/2018/03/16/twitter-bot-detector-software/
    - https://thenextweb.com/socialmedia/2017/11/02/tool-tells-youre-arguing-twitter-propaganda-bot/
    - https://medium.com/@robhat/identifying-propaganda-bots-on-twitter-5240e7cb81a9
    - https://botometer.iuni.iu.edu/#!/
    - https://dzone.com/articles/applying-nlp-to-tweets-with-python

- Known Twitter Accounts
    - Celebrity Accounts
        - http://profilerehab.com/twitter-help/celebrity_twitter_list
    - Known Bot Accounts
        - https://github.com/Plazmaz/Twitter-Bots-List/blob/master/list.txt
        - https://botwiki.org/bots/twitterbots/
        - http://meta-guide.com/bots-agents-assistants/100-best-twitter-bots

Available Resources

Our data will come from Twitter via the tweepy API (http://www.tweepy.org/) under the public, free, standard level of service. Among other things, this level of service does limit the amount of data we will be able to access (e.g., 7-day history only).

We will use two methods to obtain data from Twitter via tweepy:

- API.user_timeline (to obtain tweets and retweets per user id, retrospectively)
- API.statuses_lookup (to obtain full tweet details for each item on timeline)

The data we will obtain from Twitter are from the tweepy **Status, Tweet and User Objects**. As shown below, it will be a stretch goal to include in our analysis additional features that use the Twitter **Entities Object** (which contain more complex elements within a tweet, such as URLs, photos, polls, etc.)

We intend to store our data in mySQL tables through an Azure cloud service.

Our team is sharing documentation and code through a private GitHub account.

Preliminary EDA

- The following data will come directly from Twitter:
    - Related to **User Object**:
        - Date of creation
        - Location
        - URL
        - Verified status
        - Follower count
        - Friends count
        - Favorites count
        - Statuses count
        - Time zone
        - Language
    - Related to user's latest week's **Status/Tweet Object**:
        - Date of creation
        - Text
        - Source
        - Reply_count
        - Retweet_count
        - Favorite_count
        - Possibly_sensitive
    - If time permits, we will also look at storing URL information contained in tweepy's **Entity Object** (though we consider data related to hashtags, media, user mentions, symbols and polls strictly out-of-scope.
- Our intention is to engineer or enrich this data in the following manner:

    - We intend to create a separate table with tweet counts by user by "bucket of time". Our current thinking is each bucket should be the "minute of the day" (for 1,440 buckets per user). For performance reasons, we think it is better to pre-process and store these counts in each bucket rather than relying an "on-the-fly" query. We consider it only a stretch goal to create similar bucketing and distribution information for other user activities such as creating or receiving favorites or likes.
    - We intend to add classification fields to tweets and users based on an NLP analysis of each tweet's textual content. Depending on how rich our NLP models can be, we envision additional fields to include measures of text complexity, text consistency, text sentiment, text size, ratio of text information to other entity information (e.g., number of URLs, etc.)

We would like to collect the above for known real people (verified by twitter) and known bots (from online source). We will split the data we collect into a training set and testing set.

Our EDA will consist of visualizing the above features for both types of authors--which we will compare visually to see if a pattern is apparent. The following are examples of what we might see:

- Humans' tweet times are more spikey throughout the daytime whereas bots' tweet times are more steady at all hours
- Human tweets are retweeted less often than bot tweets.
- Human tweets contain text content that is more varied or complex than bot text.