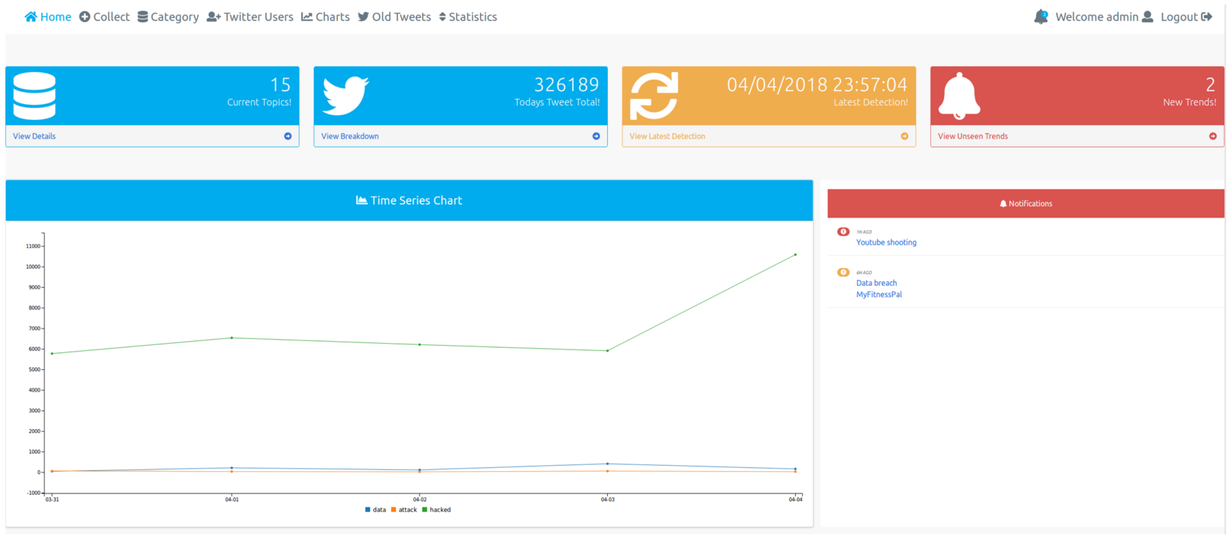
Semi-Autonomous Trend Detection using Live Twitter Data

Overview

Trend detection in social data is the attempt to detect what is suddenly becoming popular. This is important to a number of people including news organizations, retailers/advertisers, and first responders. Trends can tell us what people are currently interested in. For this project, I wanted to develop a web application that detects trends in Twitter data for the topics you are interested in. It aims to detect trends in real time, and it will pick up the key words that are relevant to the trend. For example it may find a trend on Netflix, but it would also pick up the keywords of outage, and down which suggests that maybe the website is down. Some of the features of the web app include: Adding the topics a user is interested in, auto-suggesting topics from Twitter users who the user follows, live notifications of trends as they happen, and statistics and charting of the topics for analysis over time. It will also have background tasks such as collecting historic tweet statistics for the topics so trends can be detected right away, and running the detection algorithm asynchronously in the background.

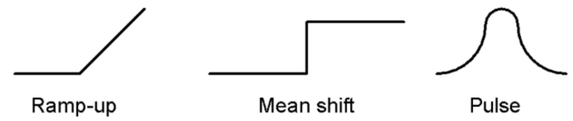


Different Types of Trends

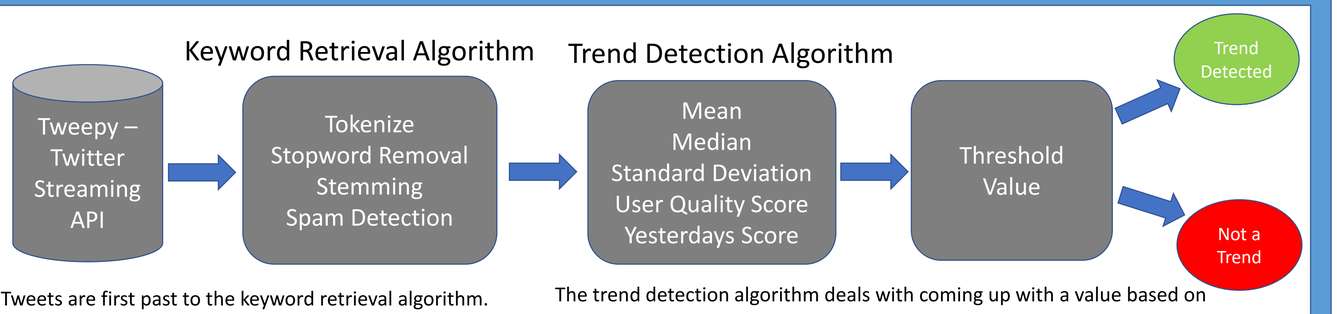
The different types of trends that are dealt with in Twitter data can be broken down into 3 different categories:

- ***Ramp-up:*** Ramp-up is a trend that starts off in a steady state, and continues to increase over time
- ***Mean shift:*** Mean shift is a trend that starts off in a steady state, and then abruptly changes to a significant value and stays there for a much longer time span.
- ***Pulse:*** Pulse is a trend that starts off in a steady state, and then increases significantly before returning to the previous steady state.

The trend detection algorithm is ran periodically in the background using Celery (Every 5 minutes for live detection, and then once more an hour to pick up slower moving trends). This is mostly for picking up Pulse trends, and then both Ramp-up, and Mean shift trends can be seen on time series graphs by graphing the data over time.



How the application detects trends



Tweets are first past to the keyword retrieval algorithm. This will tokenize each tweet (Break into individual words), remove stopwords (e.g. the, a, is), and also stem words together (e.g. Hacker, and hacked would become hack). Spam detection then filters out entries that are considered to be spam.

The trend detection algorithm deals with coming up with a value based on comparing previous tweet values stored in a database with the passed in keywords. It takes a combination of the mean, median, standard deviation, user quality score (Score given based on the reputation of the top users who have tweeted about the keyword), and also yesterdays score for that keyword. This then produces a value which is compared to a threshold which decides if it is a trend or not.

Technologies Used

python™ · django · elasticsearch · NumPy · D3 · jQuery · Celery · Tweepy · redis

Waterford Institute *of* Technology
INSTITIÚID TEICNEOLAÍOCHTA PHORT LÁIRGE

Danny O'Leary,
BSc(Hons) in Computer Forensics and Security,
Department of Computing and Mathematics, School of Science