Personality identification using online social network data

William Santos - Supervised by: Dr. Raja Akram
Information Security Group, Smart Card and IoT Security Centre

The Smart Card and Internet of Things Security Centre

Objective

The objective is to mine data ("posts" and other user-generated content) surrounding a specific group of categories/topics from an online social network platform. The dataset will large: >70GB. The data will be used to build/train machine learning models that are able to identify and cluster similar people/users/personalities.

Specifically:
- Mine posts – and relevant users – about certain topics from Twitter
- Store the data in a way that permits easy and fast querying of posts and related posts/users
- Use unsupervised machine learning and clustering techniques to identify groups of similar users

Introduction

The project consists of two key stages:

- **data acquisition**: the collection – mining – and storage of posts, users, and media from an online social network platform. I will be mining data from Twitter, and I will be using Twitter's search and live-stream API to collect posts (tweets) relevant to specific search queries.

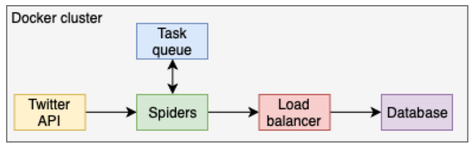
- **data analysis**: querying the dataset and using clustering algorithms to group similar users. I will create multiple models – combining different attributes – to identify similarities between users across many dimensions, including:
  - Language/word usage
  - Hashtag usage
  - Sentiment distribution
  - Activity distribution
  - Emoticon usage

Once users have been clustered, it is possible to identify and extract more specific personal details and various personality traits, such as: gender, average age, location, profile activity, and shared interests.

Data acquisition

Multiple servers were used to scrape post and user data from the Twitter API. Crawl tasks were distributed across the cluster using a message queue.

Server cluster configuration
- **13 servers** – 3 for storage, 10 for scraping
- **Docker Swarm** – for container orchestration
- **Dgraph** – primary database
- **HAProxy** – load-balances writes to database servers
- **RabbitMQ** – for task queuing
- **Grafana + Graphite** – for microservice monitoring
- **Twitter API spiders** – written in Golang

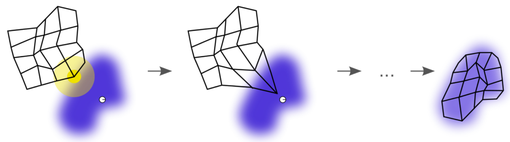


All spiders listened for crawl tasks on a message queue – and thus could be controlled remotely.
The spiders pulled data from the Twitter API for two weeks. The live-stream API was used to track tweets for search queries in real-time.
An average of 750,000 tweets were saved into the database every 25 hours.

Data analysis

Users are clustered based on tweets they have made or have re-tweeted from other users – thus users are considered similar if they post similar content. Features are extracted from posts and vectorised – to be used as inputs to the machine learning algorithms.

Self-organising feature maps [1]
Simple clustering algorithms like k-means and k-nearest would require an enormous amount of memory – due to the size of the dataset. I decided to use self-organising feature maps to encode the relationships between similar inputs as a matrix.



Single class support-vector machines [2]
Utilising single class support vector machines, I trained a model to distinguish between ordinary users and brands / celebrities. This allowed me to exclude advertisements and other irrelevant content from my training datasets.

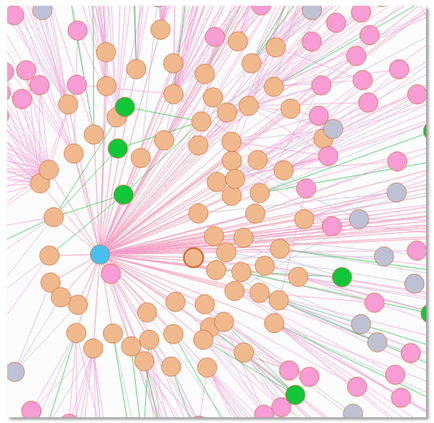
The final dataset

The final dataset consists of tweets (and associated users) relevant to the following topics:

Topics scraped
- Blogging, writing, publishing, poetry
- Photography, film photography
- Digital art, drawing

The dataset contains over 10 million tweets and 1.5 million users. The database is spread across multiple servers; the total disk space utilised is ~75GB.

Key dataset statistics
- Tweets: ~10,000,000
- Users: ~1,500,000
- Hashtags: ~3,000,000
- Total size on disk: ~75GB



The entire dataset is stored as a graph. There are many complex relationships between posts, users, and hashtags; the graph structure allows for easy insertion and querying of such relationships.

Results and limitations

Important results and findings
- It is very much possible to identify and cluster similar people using only publicly available OSN data.
- Identifying individual personalities requires manually inspecting a cluster of similar people – and extracting commonalities.
- User clustering and personality identification can be used for:
  - Targeted marketing
  - Recommendation systems

Key limitations
- **Twitter API rate limiting** – Twitter limits the number of API calls that can be made to their servers.
- **Stability of Dgraph** – the primary database experienced multiple crashes – which corrupted data and caused delays.
- **Sheer size of Twitter** – there are simply too many tweets and users to scrape in any reasonable timeframe – especially with the aforementioned API limitations.

References

[1] Wikipedia contributors. "Self-organizing map." Wikipedia, The Free Encyclopedia. Wikipedia, The Free Encyclopedia, 29 Jul. 2019. Web. 25 Aug. 2019.
[2] Wikipedia contributors. "Support-vector machine." Wikipedia, The Free Encyclopedia. Wikipedia, The Free Encyclopedia, 19 Aug. 2019. Web. 25 Aug. 2019.

Acknowledgements

Clement Petit – suggested clustering users based on emoticon usage

Contact Information

- Web: https://wsantos.io
- Email: william.santos@rhul.ac.uk
- Phone: +44 7783 791678