“Applied Network Science: Social Media Networks” Project

Documentation to the analysis of the paper[*]

Samuel Anzalone
BS Computational Science

Alexander Timans
MS Statistics

This document contains the documentation of our code used in our analysis of the paper [Schöne et al.].

1 Introduction

The GitHub repository containing the scripts can be found here. The top-level `README.md` provides a description of the repository structure. Information about the paper and material used in our analysis:

1. Version of the paper [Schöne et al.]: link (preprint, version 2; if unavailable, see `cited_papers/Schöne et al., 2021.pdf` in our GitHub repository)

2. Supplementary material: link (if unavailable, see `paper_material/` in our GitHub repository)

As of the time of writing, the paper has been published and is available here.

Scripts The code to reproduce results and plots is divided into two scripts located in our GitHub repository in the directory `our_work/scripts/`:

- `Social_Networks_Analysis1.R`: R Script for statistical analysis written by Alex Timans
- `Social_Networks_Analysis2.ipynb`: Jupyter Notebook for exploratory data analysis written by Samuel Anzalone

[*] [Schöne et al.]

Data used The data used in the analysis corresponds to the original data published by [Schöne et al.] along with their paper and code files. See `paper_material/data/` in our GitHub repository.

2 Documentation of `Social_Networks_Analysis1.R`

This code covers reproducing some of the results reported by [Schöne et al.] in their paper and then analyses different aspects of their proposed LMM model for retweet predictions. The code mainly covers the topics in the presentation titled Insight #3.1, Insight #3.2, Insight #4.

The file is structured in named sections which can directly be navigated to using the *Jump To* menu at the bottom of the editor pane in RStudio. It is also annotated with comments to help clarify the code. To better understand the structure of individual model fits and the commands used in this context, it is recommended to have a look at the code published by the authors, in particular the file `analysis_manuscript.Rmd`.

In detail, the code covers the following sections:

1. Loading the data used, including a function to extract additional transformations of the data needed for subsequent analysis (function: `transform_cols`) thus returning augmented data frames.
2. A function `polit_aff` to improve political affiliation assignments based on thresholds, as introduced in the presentation in Insight #4.
3. Partially exploring the data including distribution and skewness of the retweet counts, counts for binary affiliation, distribution of follower counts and plots such as boxplots, stripcharts or interaction plots.
4. Fitting the original models proposed by the authors (`mod.trump`).
5. Fitting the models with retweets on the original scale.
6. Testing possible effects of the SentiStrength sentiment score rescaling done by the authors.
7. Residual analysis of the original model.
8. Model comparisons and model tests for single term deletions from the original model.
9. Testing the strength of the effect of follower count on model quality.
10. Encoding negative and positive sentiment scores as factors (nominal/ordinal).
11. Testing different models: LMM with polynomials, LMM with “optimal” normalization transform, GLMM with zero-inflated negative binomial.
12. Doing a small comparison of SentiStrength and VADER sentiment scores on a subsample of tweets.
13. Some plots for the presentation and resources used (web links).

3 Documentation of `Social_Networks_Analysis2.ipynb`

The Jupyter Notebook `Social_Networks_Analysis2.ipynb` was created to explore the Trump and Hillary tweet datasets. The Trump dataset is explored in the first part of the notebook while the Hillary dataset is explored in the second part in an analogous fashion.

The following Python packages must be installed in order to run the notebook: `numpy`, `pandas`, `matplotlib`, `wordcloud`. Furthermore, for the Trump tweet generator to work, Jupyter Notebook's `ipywidgets` must be installed and enabled.

There are Python comments and markdown cells spreaded around the code for documentation. In the following, I provide a quick high level overview of the different sections:

- Preprocessing sections
  - Sections A.–E. for Trump and A.–C. for the Hillary dataset are respobsible for the data preprocessing, e.g. they create the necessary DataFrames.
  - Section C. of the Trump dataset analysis is a tweet generator: you can set the desired range of retweets and valence and it will randomly sample a tweet with the corresponding properties. If that tweet has a twitter short link attached, it will also display the respective twitter URL linking to the attachment.
- Exploratory data analysis sections
  1. Analysis of originality of tweets: Analyses the amount of retweets w.r.t. the "original" tweets, visualizes the results through a pie and line chart
  2. Analysis of accumulated tweets, followers, likes, retweets per user: analyses per user statistics (aggregates data which correspond to the same user ID together), shows skewness of data, visualizes results in scatter and bubble plots
  3. Analysis of valence (only Trump): Analyses the valence scores , i.e. visualizes the positive and negative Sentistrength occurences through pie charts
  4. Word clouds (only Trump): Visualizes the "positive" and "negative" word clouds, i.e. the most frequent words of tweets with positive/negative Sentistrength scores

References

[Schöne et al.] *Negativity Spreads More than Positivity on Twitter after both Positive and Negative Political Situations*, Schöne et al. 2021