Final project proposal: Influence ranker

Team members

juan4@illinois.edu

Topic

Apply Latent Semantic Analysis to determine whether the Natural Language used by two different individuals is influenced by the same Language Model.

Tasks (30 hours estimate)

1. Provide a User Interface to input the name of a person in Social Media and a time frame in which similarity is to be discovered (I assume that language evolves over time and similarity is best highlighted on a specific window of time). **5 hours**.
2. Crawl the public space of the searched person to retrieve its post history in order to build its Language Model. **2 hours**.
3. Crawl the public space of the searched person to retrieve a list of other individuals from whom he/she shares content (e.g. retweets their posts) or vice-versa (e.g. they retweet her/his posts). **2 hours**.
4. Crawl the post history of the list of individuals retrieved in the last step and use it as evidence when determining, via PLSA, whether the model that explains their language is a mixture of a background model (English), the searched person's Language Model derived from the second step and a third unknown model. **10 hours**.
5. If many of these individuals' posts can be explained to a certain degree (e.g. a threshold) by the searched person's model then it is assumed that they are influenced by similar factors and they are filtered in.
6. Return the list of individuals having a similar language ranked by their model's similarity, including their location.
7. Print each person from the returned list highlighting their location and the level of influence similarity (e.g. a green dot on the map whose opacity is determined by the similarity factor). **6 hours.**
8. Evaluate the accuracy of my system using a normalized Discounted Cumulative Gain score (explained below). **5 hours.**

Relevance of the application

Being able to determine whether a group of people speak similarly could mean that they share similar interests or have similar opinions. This can help a targeted ad system guess what someone might like or dislike based on the public profile of other similar people. Further extensions to this system could theorize, based on time, causal-effect relationships between

people so we can get a picture of who's influencing whom. Being able to find "influencers" in social media can be very interesting to advertising companies.

Tools

For the front-end I will be creating a webpage (possibly in React) to allow users to enter input and visualize the output of the ranking algorithm in a map.
For the backend I'm planning to use a web server (possibly in C#) with access to a public social media API (Twitter or Facebook) to crawl for data and then use the PLSA algorithm to calculate the similarity between the Unigram Language Models of the searched person and his/her possible matches.

Datasets

In order to explain a person's Language by a background model and the model of an "influencer/influenced" person, I will be requiring the datasets of the last two. The list of posts from the influencer (or influenced) will be retrieved online while doing the search but the background model can be preloaded using the English language dataset as a reference. However, given how people in social media have, in general, a very particular Language Model already when compared with overall English, it could perhaps be a better idea to shape the background model using data crawled directly from the social media app.

Evaluation

In order to score the effectiveness of my ranking application, I propose a system in which a human judge reads 10-30 posts of a given person to help her/him determine the unique traits of their language. Then my system will return a list of **N** people, including 10-30 posts of their own, believed to use a similar language so the judge can rank (using a scale of 1-5 for example) how similar they are indeed.
Given the feedback provided by the human judge, I will use nDCG score to grade my application.