SUPERVISED PREDICTION OF TWITTER POSTS WITH UNSUPERVISED CLUSTERING

GREGORY HANDY, DOLAN ANTENUCCI, AKSHAY MODI, MILLER TINKERHESS

Abstract. Abstract goes here

1. Introduction

This is the start. This is how you cite stuff [1].

2. Clustering Technique

2.1. **Difficulties of Clustering.** Clustering is inherently a hard task, and clustering hashtags is no exception to this rule. There are a number of variables that make clustering hashtags very difficult. The first major difficulty is the fact that they are not required to be defined words. For example, the hashtag #p2 is a popular marker if a tweet contains politically progressive thoughts. In most cases, hashtags are a concatenation of words such as #iamthemob. There exist measures such as the Wu-Palmer distance and path distance similarity that are distances that are based on the synonyms of the words being examined. Unfortunately, this feature eliminates the possibility of using this measure as central component in a clustering algorithm, such as K-means. Our algorithm will still use this concept, but only in a very limited scope.

The fact that hashtags are usually concatenations also makes it difficult to apply another popular metric to compare words, known as Levenshtein distance (i.e. edit distance). As an example, this distance would say that the hashtags #iamblessed and #iamthemob are close to one another, when in reality, they are very different. We actually implemented a method centered on this distance, and the poor results backed this observation.

Further, the set of hashtags is constantly increasing in size, with new, trendy hashtags appear everyday. This fact forces us to create an algorithm that captures the essence of most hashtags used, while remain tractable. To account for this, we decided to focus our attention on the 2,000 most used hashtags, and clustered only over this set. This decision was a directly consequent of the final clustering algorithm, and will result in the clustering of more than 2,000 hashtags.

2.2. **Co-Occurrence Relation.** The paper written by J. Poschko developed the idea that two hashtags are similar if they co-occur in a tweet. Intuitively, this concept makes sense. Two hashtags are inherently similar if they are contained in tweets that discuss the same topic, and this fact could not be any stronger than if they appear in the same exact tweet. To back this intuition, the paper calculates the Wu-Palmer distance between hashtags that co-occur, and shows that this value is higher than the distance between two randomly chosen words. It was mentioned previously that the Wu-Palmer distance is a poor distance for hashtags as a whole, since there would be many unknown distances between hashtags that are not words. However, it does provide a good measure if you only consider the set of hashtags that is made up of well-defined words. From this fact, we make the assumption that if the Wu-Palmer distance is high for reoccurring hashtags that are real words, then the set of all reoccurring hashtags can be used as baseline for a similarity measure.

After creating the co-occurring lists for the 2,000 most used hashtags, we used the natural language toolkit (NLTK) in python to find the Wu-Palmer distance for each hashtag in each list. We then took the average over all of the lists. The final value is found in Table 1, along with the baselines found by Poschko.

INSERT TABLE

This table verifies that the claim by Poschko applies to our dataset. Clearly, the hashtags found in co-occurrence list are much more similar than random words.

2.3. **Minimizing the Level of Noise.** However, even though this applies to the list as a whole, it does not mean that there is not still a considerable about of noise present. For example, #photography is a popular hashtag one uses to denote a recently posted picture, and another hashtag is sometimes used to describe the place the picture is taken, such as #iran. However, as this example illustrates, the fact that these hashtags co-occur does not mean that these two hashtags are similar by our definition of similarity. To help minimize the effect this has on later stages, let n_{ij} be the total number of co-occurrences hashtag i has with hashtags j. Hashtags A and B are kept on the co-occurring list if and only if

$$min(\frac{n_{AB}}{\sum n_{Aj}}, \frac{n_{BA}}{\sum n_{Bj}}) > .05,$$

otherwise it is removed from the list.

Even after this step, it is still possible that many noisy relations exist; therefore we do another level of filtering, by comparing the contents of the respective co-occurring lists. Let m be the total number of hashtags that occur in both hashtag A and hashtag B co-occurrence lists. Further, let m_A be the total number of occurrences of these

hashtags in list A, and m_B be the total number of occurrences in list B. Then the relationship between hashtag A and hashtag B is maintained if an only if

$$min(\frac{m_A}{\sum n_{Aj}}, \frac{m_B}{\sum n_{Bj}}) > .2.$$

Note that by the first level of filtering, this minimum is at least .05. Comparing each list takes a considerable amount of time to run, and this bottleneck forced us to constrain our focus to only 2,000 hashtags. However, it is easily parallelizable, and when program efficiently, would allow a much greater number of hashtags to be considered.

2.4. **Defining the Similarity Measure.** Using the Wu-Palmer distance and filtering methods described, we are confident the lists of co-occurring hashtags that remain imply some level of similarity between hashtags in these lists. As a result, we define the following similarity measure between hashtags A and hashtags B:

$$S(A, B) = (\frac{n_{AB}}{\sum n_{Aj}} + \frac{n_{BA}}{\sum n_{Bj}})/2.$$

This function fits the axioms of a similarity measure. S(A,B) = S(B,A), and S(A,B) $\geq$ 0, and based on filtering process it will actually be greater than .05. It should be noted that several similarity measures were tested, such as multiplying the two ratios together, but this proved to be the best measure.

Figure 1 illustrates the algorithm in a simple case.

References

[1] M. Shiels, *Twitter co-founder Jack Dorsey rejoins company*, BBC News (28 March 2011), available at `http://www.fermentas.com/techinfo/nucleicacids/maplambda.htm`.

[2] Alexa (17 November 2011), available at `http://www.alexa.com/topsites`.

[3] Twitter.com, *Your World, More Connected* (2011), available at `http://blog.twitter.com/2011/08/your-world-more-connected.html`.

[4] Bollen H. Mao and X. Zeng, *Twitter mood predicts the stock market*, Journal of Computational Science **2** (March 2011), no. 1, 1-8.

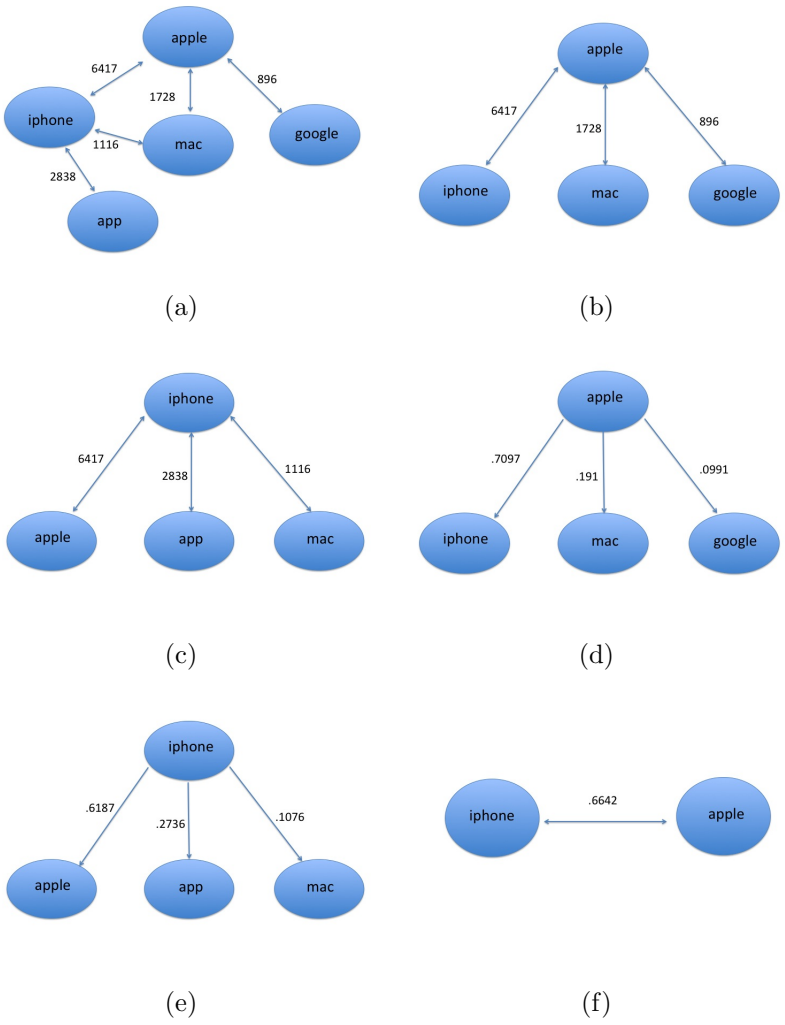

FIGURE 1. Figure (a) represents the origial undirected graph proposed by the co-occurring lists. Figures (b) and (c) consists of only the neighbors of apple and iphone respectively (the graph remains undirected). Figures (d) and (e) convert the undirect graphs into an directed graph by weighing the edges. The filter process examines these directed graphs, and after passing the final undirected graph with the similarity measure is created.