Clustering Twitter Feeds using Word Co-occurrence
CS784 Project Report

Tushar Khot
Computer Sciences Department
University of Wisconsin, Madison, WI
tushar@cs.wisc.edu

Abstract

For very large number of documents, normal clustering methods would take O($document^2$) time. When the number of documents are very large but short such as tweets, it may make sense to actually cluster the words. We present a method that clusters the words using the word co-occurrence as a similarity measure. We use spectral clustering for creating word clusters and do a “search” to get the actual documents. The resulting word clusters and tweets make sense most of the times.

1 Introduction

With the growth in popularity of micro-blogging sites such as Twitter and status updates on Google Buzz/Facebook, we now have access to a large corpus of very small documents.

Moreover, these documents may talk about events happening in real time and can be very useful source of information. The power of Twitter has already been shown in the last Iran elections and using trends from Twitter is becoming more and more common. But the trends tend to revolve around a single word i.e. they just indicate which word types are being talked about the most. What could be really useful is to have some idea of what ”story” is popular right now. Rather than looking at popular words such as ”Haiti”, ”Washington”, ”earthquake”, ”snowstorm”; we should be able to look at popular clusters - ”Haiti earthquake Richter scale 7.0”, ”Washington snowstorm 18 inches”.

We would need the system to be as real-time as possible and hence we can’t do full scale document clustering on all the Tweets. We can afford to ”forget” documents that are no more relevant as they are really old(say 24hrs in Twitter is old). Even within a small time period, we would have a large document corpus. Hence instead of clustering over the documents, we would instead cluster the word types by using the co-occurrence counts. As shown in Figure 1, as we look at more documents the number of word types actually flatten out.So as the number of documents increase, at some point we would have fewer word types than the number of documents.

Even if the word types are more than required, we could restrict to word types in English dictionary and Named Entities. Hence the approach that we plan to take for discovering news in Twitter feeds is as follows:

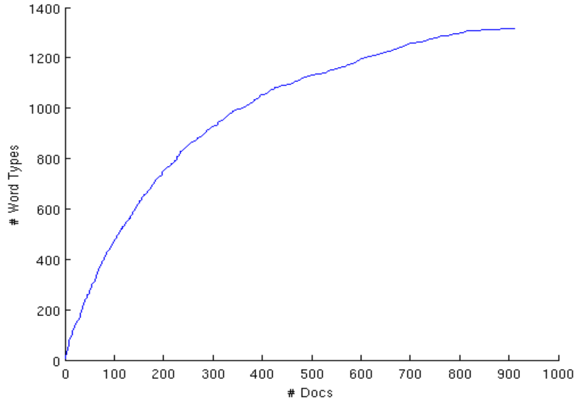


Figure 1: Number of Word Types Vs Documents

1. Maintain co-occurrence counts for every pair of word in each Tweet.
2. Maintain a last-updated time stamp for each pair of words and remove pairs that are not updated for some time or maybe use a decay rate.
3. Find clusters of words that have high co-occurrence counts.
4. Display these clusters as wordles.

2 Design

The overall system design is shown in Figure 2. My implementation does the following steps:

1. Selected eight news twitter feeds to fetch. This ensures that all tweets are in English, less noisy and more likely to form clusters. Twitter feeds used:
   - cnnbrk
   - nytimes
   - breaking news
   - fox
   - msnbc

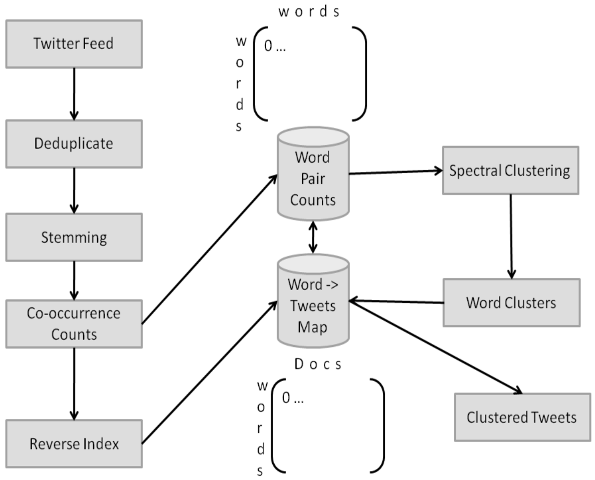


Figure 2: System Design

- foxnewsbrk
- abc
- wsjbreakingnews

2. Collected around 1000 tweets for 9 days spread over a 2 months period.
3. Sorted the tweets and removed tweets with at least first 60 characters in common. This is to avoid some news repeating their tweets again and again.
4. Put spaces around some punctuations such as , ; : - but not . ' as we don't want to split "U.S." or "don't".
5. Remove some basic stopwords(using Lingua::Stopwords in Perl) along with specific twitter stop words such as "alert", "breaking".
6. To help in clustering, stemmed the words using a Perl implementation of Porter stemmer. If we care about the word clusters making sense and maybe use them for search, we should avoid stemming.
7. Used the CMU toolkit to get a vocabulary of unigrams on this dataset and created features for every tweet.
8. Using these features, we created a word co-occurrence matrix, W. W_{ij} is set to n, if there are n tweets that contain both the features i and j.
9. Perform spectral clustering on the weight matrix, W to obtain clusters of words.
10. Use these words and a reverse index to obtain the cluster of tweets.

3 Spectral Clustering

To perform spectral clustering, we experimented with different settings to get decent word clusters.

Weight Matrix

Using the weight matrix, W without changes gave decent results but I also experimented with KNN and $\epsilon - NN$ (Zhu 2010).

KNN : We set the top K neighbours for every word to 1 and others to 0. If K is too small, the graph is almost disconnected and for a K too high, its almost same as using the original W matrix without the weights. K=100 really works well for our dataset

$\epsilon - NN$: In this method, we would set all the edges with weight less than ϵ to 0. This somehow doesn't work really well as even the smallest sensible ϵ(which would be 2 for our case) removes edges between words that occur together just once. Since most clusters are of size 2 or 3, this actually worsens the situation.

Laplacian

There are three possible Laplacians that we could use (Chung 1997):

$$\begin{aligned} L_{unn} &= D - W \\ L_{rw} &= I - D^{-1}W \\ L_{sym} &= I - D^{-1/2}WD^{-1/2} \end{aligned}$$

where D is a diagonal matrix such that $D_{ii} = \sum_{j=1}^{n} W_{ij}$.

We used L_{unn} and L_{sym} which seem to give similar results. L_{rw} tends to give complex values in the eigenvectors.

Clustering

We find C eigen vectors, ϕ_i corresponding to the smallest C eigen values. We arrange these eigen vectors as columns of a matrix, $Data = [\phi_1 \ \phi_2 \ ... \ \phi_C]$ of size $N * C$. N is the number of word types. We used the kmeans implementation in Matlab for clustering the words using the new $Data$ matrix into C clusters. Generally setting the number of clusters to 100 seems to work the best. The only issue with clustering is that it partitions the graph, but in our case the same word may belong to different clusters. For e.g., "Obama" may belong to different clusters such as "health care" and "drilling". Hence instead of actually using the clusters from kmeans, we used the centroids returned by kmeans also. We picked the top M words most similar to the centroid to obtain better clusters. In results, we shall call this M-nearest centroid clusters[M-NCC]. This is just for ease of naming.

4 Twitter Clusters

To obtain the actual tweets for the word clusters, we maintain a reverse index. The reverse index contains a map from the stemmed words to the documents(tweets) that contain these words. Given a word cluster, each word *votes* for the documents that they belong to. After going through all the words, the tweets that at least have n votes are selected. n here is set to be half of the number of words in the cluster.

5 Results

We are showing only the results that were obtained with the settings mentioned before. Tables 1 - 4 show the resulting clusters for various settings. The text in bold is the actual word cluster which is followed by the tweets. Figures 3 to 7 show a wordle for some sample clusters.

Due to lack of any ground truth, we don't present any precision recall numbers here. Some facts regarding the clustering :

Size of the vocabulary: 1374 words

Number of lines: 1009 lines (after deduplication)
Unnormalized/Symmetric Laplacian(200 clusters):
Time taken for word clustering: 20 seconds
Time taken to find tweets: 10 seconds
Dataset Location:
http://pages.cs.wisc.edu/ tushar/projects/cs769/twitter.txt

136th, churchil, derbi, down, kentucki, muddi, saver, super, track
Kentucky Derby. Live blogging from Churchill Downs via the Rail blog - http://bit.ly/crGUfj
Super Saver wins 136th running of Kentucky Derby in muddy conditions http://bit.ly/caJ3T3
Super Saver wins 136th running of Kentucky Derby on a muddy track http://bit.ly/ds3rP7
Super Saver wins Kentucky Derby on soupy, soggy day at Churchill Downs http://fxn.ws/9Pdjx5

Table 1: Unnormalized Laplacian

america, aob5k7, atlant, drill, energi, expand, explor, offshor
Obama announces he's expanding offshore oil drilling along Atlantic coast so America can rely more on 'homegrown fuels and clean energy'
Obama energy plan would open up Gulf drilling http://on.cnn.com/aOb5k7
Obama unveils plans to open waters in the Atlantic and Gulf to drilling. http://on.cnn.com/aOb5k7

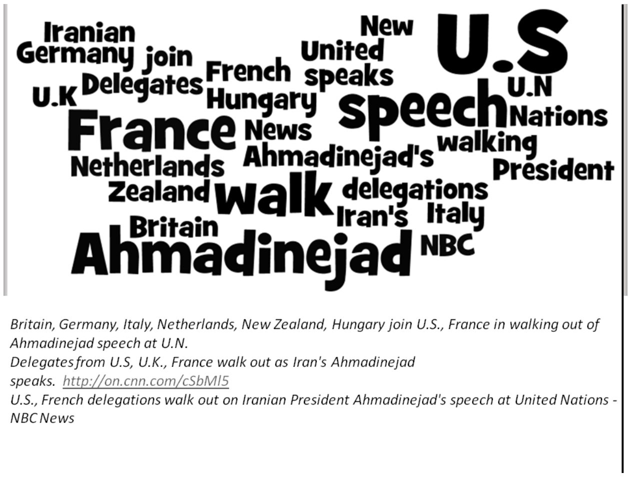
Table 2: Symmetric Laplacian

48, hour, travel, respons
Obama to visit Gulf of Mexico region in next 48 hours to check oil spill response, White House says. http://on.cnn.com/cHJLHp
President Obama is to travel to the U.S. Gulf Coast in next 48 hours in wake of oil spill, White House officials say - Reuters
President Obama will travel to Gulf coast in the next 48 hours in response to oil spill http://fxn.ws/bMWk9U

Table 3: Unnormalized Laplacian with 4-NCC

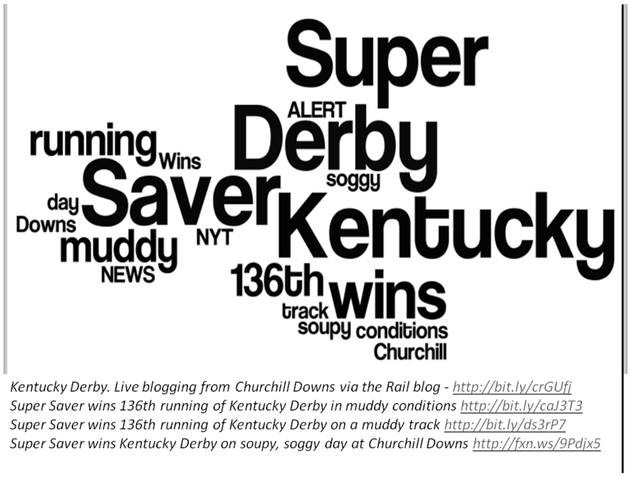
infantino, sling, recal, babi, link
More than 1 million Infantino baby slings are being recalled after being linked to 3 deaths by suffocation - AP
Sling Sting: Million Baby Holders Recalled: http://bit.ly/a3OkSD
Update: U.S. agency says parents should stop using Infantino SlingRider and Wendy Bellissimo slings for babies under 4 months

Table 4: Symmetric Laplacian with 5-NCC



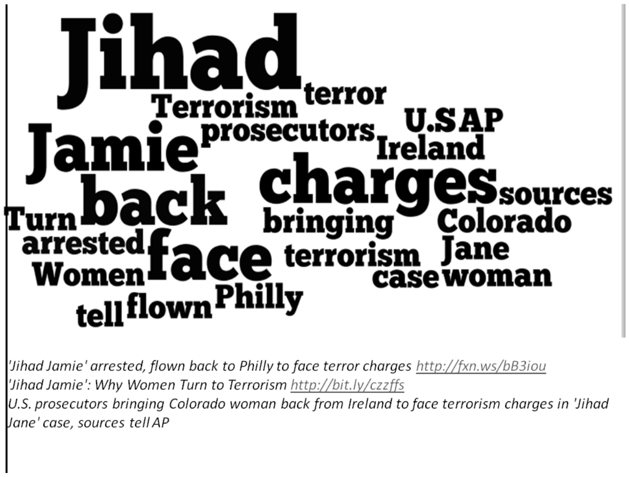
Britain, Germany, Italy, Netherlands, New Zealand, Hungary join U.S., France in walking out of Ahmadinejad speech at U.N.
Delegates from U.S, U.K., France walk out as Iran's Ahmadinejad speaks. http://on.cnn.com/cSbMl5
U.S., French delegations walk out on Iranian President Ahmadinejad's speech at United Nations - NBC News

Figure 3: Sample Cluster



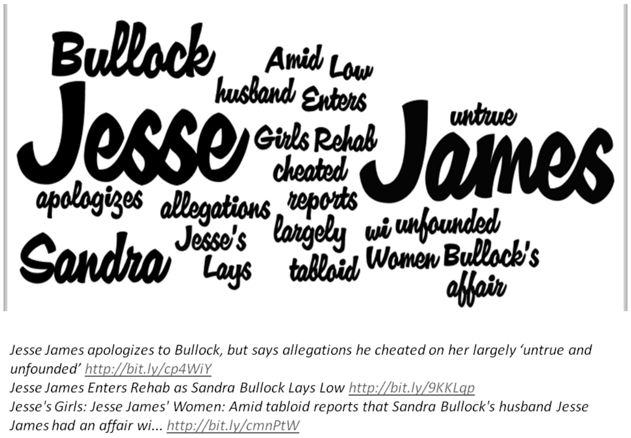
Kentucky Derby. Live blogging from Churchill Downs via the Rail blog - http://bit.ly/crGUfj
Super Saver wins 136th running of Kentucky Derby in muddy conditions http://bit.ly/caJ3T3
Super Saver wins 136th running of Kentucky Derby on a muddy track http://bit.ly/ds3rP7
Super Saver wins Kentucky Derby on soupy, soggy day at Churchill Downs http://fxn.ws/9Pdjx5

Figure 4: Sample Cluster



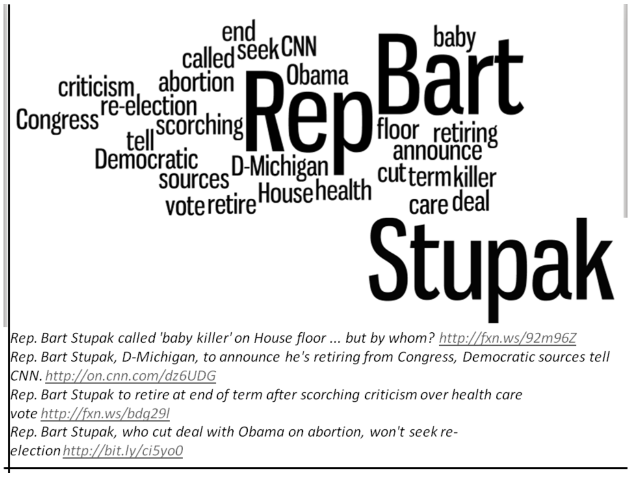
'Jihad Jamie' arrested, flown back to Philly to face terror charges http://fxn.ws/bB3iou
'Jihad Jamie': Why Women Turn to Terrorism http://bit.ly/czzffs
U.S. prosecutors bringing Colorado woman back from Ireland to face terrorism charges in 'Jihad Jane' case, sources tell AP

Figure 5: Sample Cluster



Jesse James apologizes to Bullock, but says allegations he cheated on her largely ‘untrue and unfounded’ http://bit.ly/cp4WiY
Jesse James Enters Rehab as Sandra Bullock Lays Low http://bit.ly/9KKLqp
Jesse's Girls: Jesse James' Women: Amid tabloid reports that Sandra Bullock's husband Jesse James had an affair wi... http://bit.ly/cmnPtW

Figure 6: Sample Cluster



Rep. Bart Stupak called 'baby killer' on House floor ... but by whom? http://fxn.ws/92m96Z
Rep. Bart Stupak, D-Michigan, to announce he's retiring from Congress, Democratic sources tell CNN. http://on.cnn.com/dz6UDG
Rep. Bart Stupak to retire at end of term after scorching criticism over health care vote http://fxn.ws/bdg29l
Rep. Bart Stupak, who cut deal with Obama on abortion, won't seek re-election http://bit.ly/ci5yo0

Figure 7: Sample Cluster

We also show various metrics computed for different C(number of clusters) and K(number of neighbours considered in KNN) for L_{sym} using 4-NCC for obtaining the clusters.

We show the clustering time over different C and K values in Figure 8. There is an unexpected peak at C=550 and K=350. This is because Matlab sometimes prints warnings if it can't find good eigen vectors which can be very long. But otherwise the time taken to cluster increases very slowly.

Figure 9 shows the time taken to search for matching tweets for every cluster. As one would expect, the search time increases linearly with the number of clusters. This motivates the need for better search techniques using databases.

Figure 10 shows the number of 'good' clusters(clusters with more than 2 tweets) that are obtained for different C and K values. The number of clusters don't seem to have any direct correlation with the C and K values. This just indicates that choosing a larger C or K value wouldn't guarantee more or even better clusters. Some of the clusters are repeated don't even make sense if there are too many of them. If there are too few, we miss important clusters.

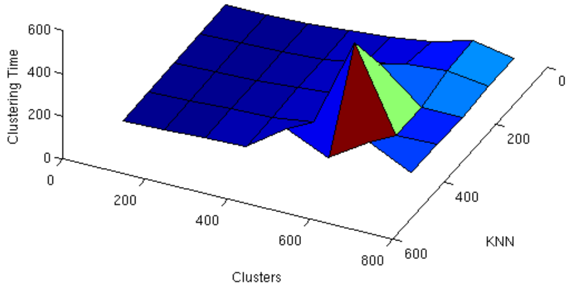


Figure 8: Clustering Time

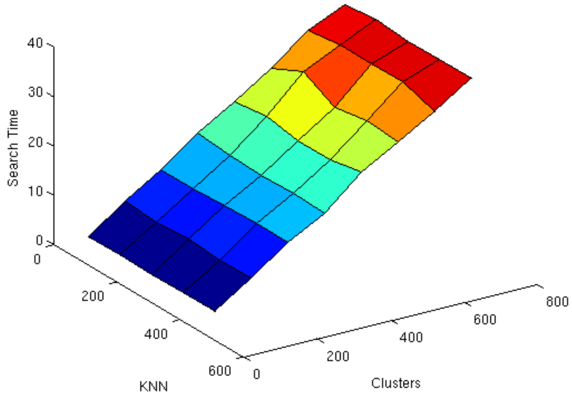


Figure 9: Search Time

6 Future Work

As mentioned before, the main motivation of this work was to scale this over many documents. So the next steps would be to run this over more than 1000 documents and cluster in real-time. This may involve using a database for storing the word co-occurrence counts. With more documents, we may have to do some kind of filtering to reduce the size of the weight matrix and remove outdated edges as mentioned before. There might be other clustering methods that can be explored to reduce the time taken for clustering.

7 Conclusion

Spectral clustering using word co-occurrence is an interesting option for faster clustering over a large set of documents. But it may not be faster than other heuristics which are used for clustering without computing distances between every

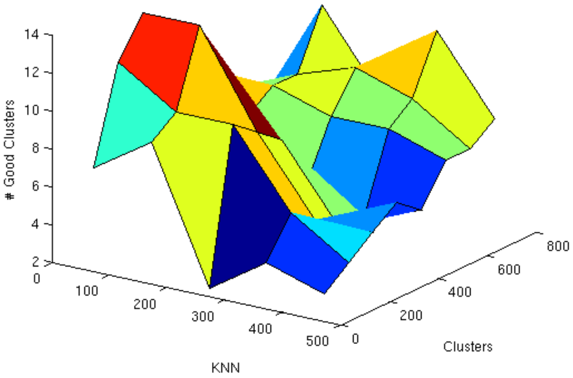


Figure 10: Good Clusters

pair of documents. There is an additional lookup time involved to get the documents from the clusters, but there have been many optimizations for improving search. Also once we have the clusters, it is possible to get new documents that belong to the cluster in real time[www.twitter.com/search].

References

[Chung 1997] Chung, F. *Spectral graph theory*. In Vol. 92 of the CBMS Regional Conference Series in Mathematics,1997.

[Zhu 2010] Xiaojin Zhu. *Clustering*. In CS769 - Spring 2010 - Advanced Natural Language Processing Course Notes.