SAND2015-9895C

Grandmaster: Interactive text-based analytics of social media.

Nathan Fabian, Warren Davis, Elaine M. Raybourn, Kiran Lakkaraju and Jon Whetzel
Sandia National Laboratories
New Mexico, USA
Email: ndfabia, wldavis, emraybo, klakkar, and jhwhetz@sandia.gov

***Abstract*—People use social media resources like Twitter, Facebook, forums etc. to share and discuss various activities or topics. By aggregating topic trends across many individuals using these services, we seek to construct a richer profile of a person's activities and interests as well as provide a broader context of those activities. This profile may then be used in a variety of ways to understand groups as a collection of interests and affinities and an individual's participation in those groups. Our approach considers that much of these data will be unstructured, free-form text. By analyzing free-form text directly, we may be able to gain an implicit grouping of individuals with shared interests based on shared conversation, and not on explicit social software linking them. In this paper, we discuss a proof-of-concept application called Grandmaster built to pull short sections of text, a person's comments or Twitter posts, together by analysis and visualization to allow a gestalt understanding of the full collection of all individuals: how groups are similar and how they differ, based on their text inputs.**

I. Introduction

People use social media resources like Twitter, Facebook, forums, etc. to share and discuss various activities or topics. By aggregating trends across many individuals using these services, we seek to construct a rich profile of a groups's activities and interests as well as provide a broader, group-based context to individuals as members of these groups.

Social media services provide API's to external applications through an opt-in procedure that allows external software to pull out an individual's data and process it. Project Grandmaster was developed to pull a person's social media data together with analysis allowing visual exploration, summarization and understanding of the data in aggregate. Using text analysis algorithms previously developed in Titan[2] and web-crawling technology developed in Avondale[1], we crawled the data, where it was not already available and processed the data into clusters of documents and clusters of individuals. To view and interact with these data, we developed custom visualizations to show these clusters, that allow an aggregate understanding of a group by reinforcing and amplifying patterns within identifying a stereotypical group identity.

It is important to note that although the data are from an individual, it is data in aggregate which is important to the results. We are not making a personality profile for an individual[20], [21], instead we consider only stereotypical behaviors for a group. The association of individuals with groups is probabilistic and can only infer qualities, not to guarantee their existence.

In the remainder of this paper we will discuss the implementation of the application, how we process the data, and the algorithms involved in doing so. We discuss our analysis using two sets of data. One is a bootstrap social media set consisting of approximately $10,000$ tweets of Twitter data taken from Senators, professional gamers, E-learning experts and Hollywood celebrities. We provide examples of the results for a full data set of $546,570$ tweets consisting of all data from that same group from when they first joined Twitter up to June 2013 when we stopped collecting. The other data set is a collection of comments in the forum of a massively multiplayer online game (MMOG) where people discuss the game at a high level as well as collaborate to develop strategies or organize trades. These data consist of approximately $149,921$ comments from nearly $3,836$ individuals.

II. Related Work

Much work on using Twitter or other social media data focused on the network structure, or link graph of associations between users. There is a rich set of data here, but there are already many existing approach to analyzing that graph [18], [4]. Instead our approach considers the unstructured, free-form content of the data. By analyzing this free-form text directly, we may be able to gain an implicit grouping of individuals through their semantics, rather than their explicit linking.

Work has also been done with regard to the semantic content of Twitter. Ritter et al. [16], seek to understand and tag conversations as they proceed through the dialogue acts that consitute a structure to human communication. This structure builds from the individual tweets into the larger back and forth communication occuring between individuals. While Smith et al. [18], are primarily analysing and visualizing the explicit structure, they also look at semantic sub-networks centered around individual topics or hashtags and the proponents of those topics, especially those groups polarized, and thus separated in the network, by a particular issue. The work of Abel et al, [3], explores the use of semantics to link Twitter user's posts with news articles in order to build a richer model of the content. While they do not construct an implicit linking between users based on this enrichment, we believe this approach could benefit the model we describe below and potentially improve the resulting clusters.

Both Golbeck et al. [11] and Quercia et al. [15] use semantic text analysis of the tweets belonging to an individual to predict traits the user would have in a personality test known as "The Big Five". By classifying users according to these personality traits links between individuals and groups of individuals may be found. The nature of links in our work differs and are based on the similarity of the content itself, and not as necessarily on the similarity of the personality expressed. However, this could also be a useful addition to improve our model.

In the academic and commercial spaces work is done to collaboratively filter and recommend Twitter user's to follow, implying an implicit link that could be made explicit. As an example, the work of Hannon et al, [12], look at the content of the individual's tweets and those of who they follow and who follow them. Using text processing over this set of tweets, they make a recommendation for other Twitter users to follow. Our work also creates an implicit link between individuals that could be used to recommend followers. However, we create an additional clustering of the tweets themselves, rather than linking based on term frequency in the collection. This gives us better explanatory power over the reason behind the grouping of individuals, and allows us to understand what those groups are interested in talking about, e.g., Figures 7 and 8.

We believe our work is unique in constructing, from semantic content, a secondary network structure between users. The work of Smith et al. is perhaps most similar in finding a secondary network, but they build modifications as filterings of the explicit network. Our work is independent of the explicit network, generating the secondary network independently, and can as a result generalize beyond the available explicit linking to add an additional, implicit network structure or to build one where it did not exist previously, e.g., forums or article comments.

III. Implementation

Grandmaster is an open source, web-centric application available at https://github.com/sandialabs/grandmaster. The data collection and processing happen in the background and update a Mongo database. Once the processing is complete, the visualization is very low overhead and can be run from a laptop. The visualization code itself runs on the user's device as part of a webpage that queries new data each time the page refreshes and builds the visualizations.

The texts are pulled from Twitter and treated as a collection of separate documents one for each tweet. Similarly forum comments are pulled from the database and each comment in a thread is considered a separate document. Each of these separate documents remains associated with the individual who created it, whether "replying" or "retweeting". Therefore the final data representation for an individual is an array of documents. This allows us to cluster each of these documents independently for content. Using these document clusters we can determine an associated array for each individual of how they align with each document cluster, reflecting a person's diversity in categories of conversation. This is clarified in more detail in Section III-A discussing our motivation behind the analysis pipeline.

At a high level, Project Grandmaster is made up of two main parts: an analysis pipeline and a visualization suite. Additionally in the case of Twitter we use web crawling to collect the data and write it out to a Mongo database for further processing by the analysis pipeline. For an existing dataset like the forum data we simply insert the documents from their input form into the Mongo database as individual records. The analysis pipeline is handled in large part by the Titan Toolkit. We access Titan through a collection of Python scripts and store partial results at each step out to a Mongo database for display by the visualization code. The analysis pipeline is described in more detail in Section III-A. Finally, the visualizations were developed using D3[7], a JavaScript based visualization library, and are described in Section III-B.

A. The Data Analysis Pipeline

For the purposes of analysis we assume the documents to be processed are stored as a complete set of individual records in the database. Although it would be possible to process the data incrementally, at this stage we simply reprocess the entire collection each time we want to update the data set. The processing is broken up into four distinct algorithms: Latent Dirichlet Allocation (LDA), clustering of documents, word cloud preprocessing, and user clustering.

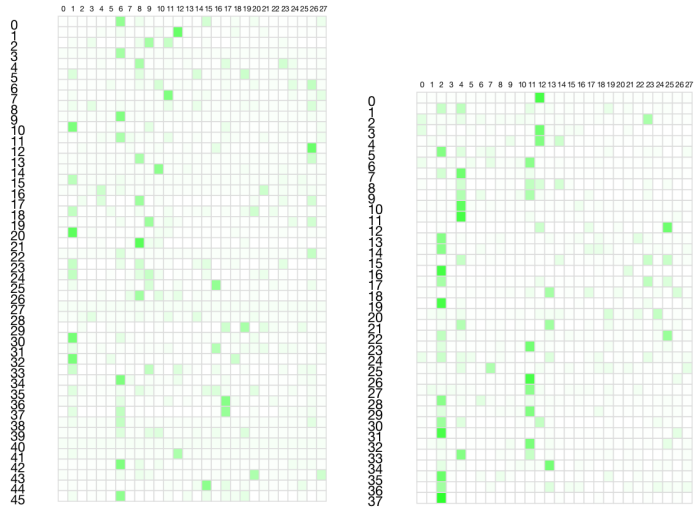

Fig. 1: Document distribution for a collection of documents assigned to two users, a gamer and a senator. Brighter green in a column means higher probability of that topic in the document at that row.

We use a parallel version LDA, which we refer to as Parallel Latent Dirichlet Allocation (pLDA) [6], [8], [9]. In our case the corpus of documents are a collection of the atomic components, e.g., tweets or comments, with each document remaining identified to an individual, Figure 1. We wish to compare these individuals through the complete set of their documents. We could compare each individual's document

to another, i.e., a row in Figure 1 by assigning a numeric distance metric between the sets of document rows. However, this would result in $O(D^2)$ total comparisons where D is the number of documents. This would not scale well to the larger number, $D = 546,470$ tweets or $D = 149,921$ comments.

To counter this scaling issue, we aggregate the documents down to a single vector for each user. One approach, would be to average the term distributions across all documents for that user to create a single vector average. Then we could use a distance metric to compare just the final vectors. However, averaging tends to squash out a lot of the diversity contained in an individual's documents. We will show later the importance of maintaining these individualities in the examples shown in Section IV-B. Instead of averaging, our approach clusters all documents first, and then projects a user onto this new document cluster space, by projecting a weight of each of their documents into the space. Instead of squashing out the diversity this reinforces documents within a cluster and highlights the difference between documents in different clusters, giving a better view of the variety in the individual.

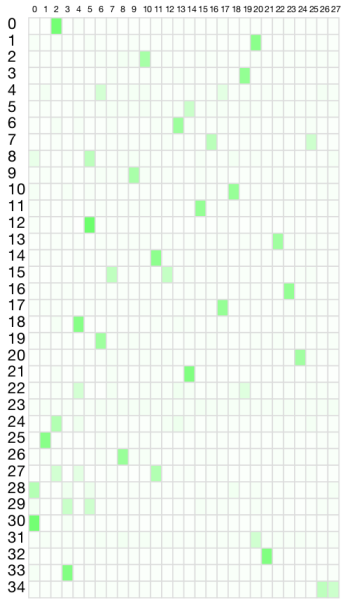

Fig. 2: Averaging the topic vectors over a clustered group of documents without squashing the vector as it would with averaging. Instead certain subsets of topics are highlighted and reinforced in each document cluster

.

Using cosine similarity as our distance metric between documents, we can use K-Means clustering [14] to group all documents (across all users) into a set of K distinct clusters. Figure 2 shows the averaged topic distribution for each document cluster. Note, how even though we are averaging the topic vector is reinforced by the similarity between documents in the cluster, instead of squashed across a broader diversity. We can use this to better understand the diversity of an individual when they project across these different clusters.

Now, we are able to project through these document clusters

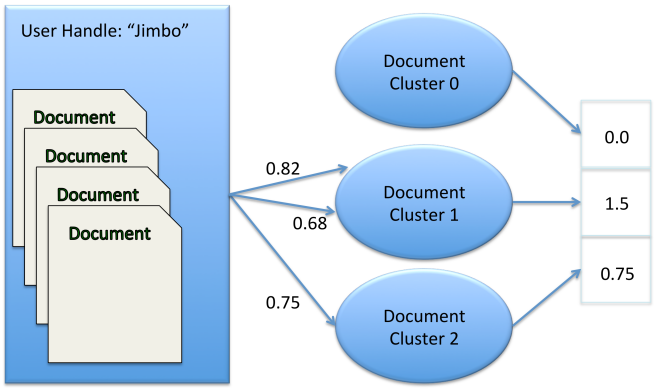

Fig. 3: The creation of a document vector for a user using the assignment of the user's documents into document clusters.

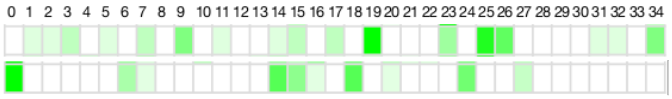

Fig. 4: User vectors produced by projecting a users documents through the document clusters as shown in Figure 3.

as vectors of weighted document counts for each person. For each document that is assigned to a particular person's cluster, that person's document vectors get a weighted increment in the associated index, Figure 3. The weighted increment is based on how close to the center the document is; documents will fall between 0.5 and 1.0 in this scale. Figure 4 shows the resulting document vector for the two individuals from Figure 1.

With this new vector, we finally have a useful single vector describing an individual and can now cluster individuals. Once again we use a distance metric, Euclidean here, and cluster the individuals again using K-means. Thus we are able to find groups of individuals with shared communication patterns.

For final visualization, we preprocess each document cluster to create a word cloud visualization[13]. A word cloud, also known as tag cloud, is a useful visual summarization of a collection of documents. It visually scales larger words which are frequently used in the collection and scales smaller words which are infrequently used. There are many variations of word clouds and many ways to sort. In our case, we use an arbitrary spatial arrangement and use only scale to communicate the frequency. The preprocessing step involves counting terms in each document cluster to determine the count for the visualization to use for later scaling. Note, that this frequency count is entirely independent of the topic weighting. We can use this as a separate verification of sense-making in the pLDA/K-means clustering processes.

B. Data Representations, Visualizations, and User Interactions

The two main data representations we need to visualize are the user clusters and document clusters described in Section III-A. This is represented by two sets of bubbles in the

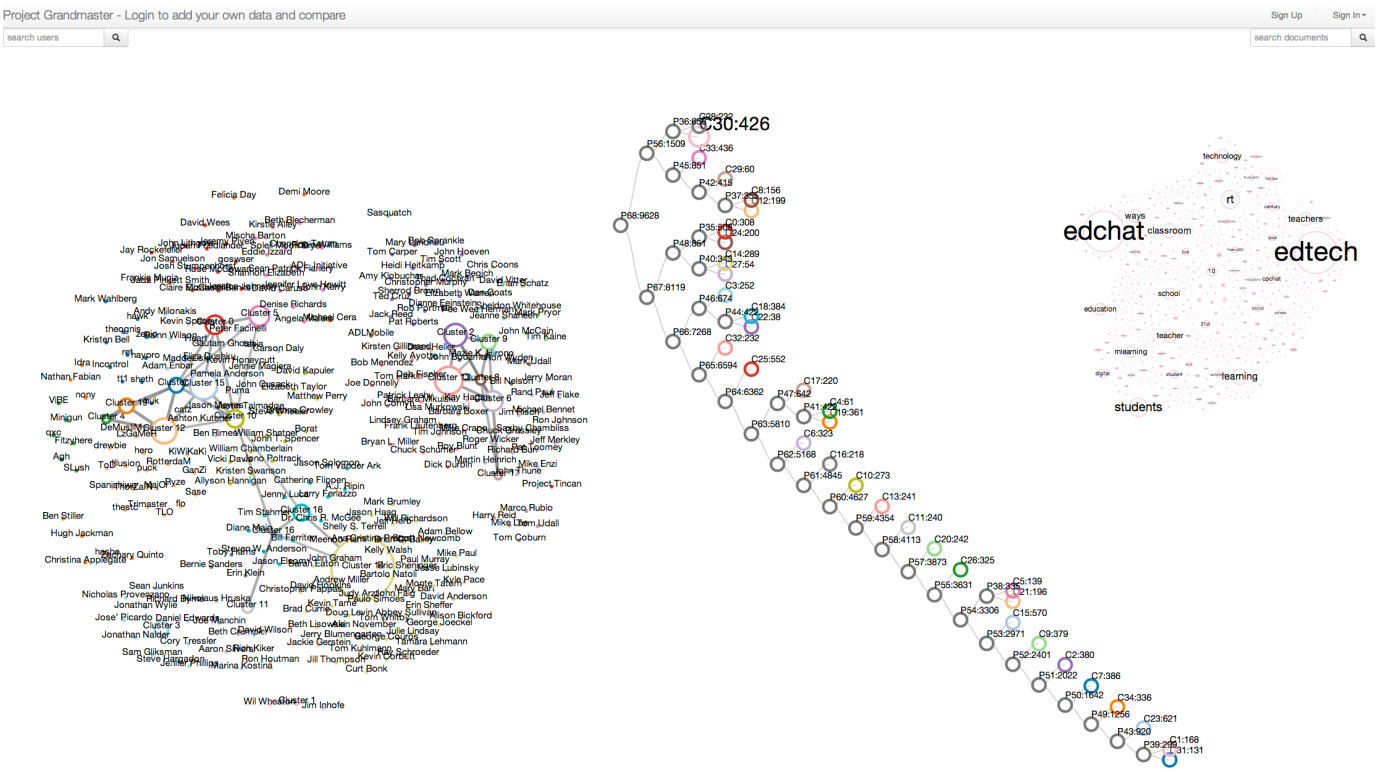

Fig. 5: Main view of Project Grandmaster. On the left are the user clusters. In the middle are the hierarchically organized document clusters. On the right is the word cloud associated with the selected document cluster.

main view of the application, Figure 5. The visualizations are developed through a web-based interface using the Javascript library, D3[7], to do the main part of the visual control. The cluster data is read from the Mongo database server-side and passed to client-side for rendering and interaction.

On the left side of the main view are the user clusters. These are organized into a force-directed layout[5] with the cluster centroids represented as rings and individuals represented as points. In a force-directed layout, all items have an implicit repelling force away from each other. There is only an attractive force between an individual's point and its cluster center. There is a secondary force between two cluster centers, represented with a semi-transparent gray line, if the two cluster centers are on average similar to each other above some threshold, i.e., the individuals contained in the cluster are similar to the individuals in the other cluster. This allows us to visualize a two-layer hierarchical arrangement between the cluster centers and the individuals.

In the middle view, the document clusters are organized into a full hierarchical agglomerative clustering (HAC), [10]. The initial clustering is done, as described in Section III-A, for all documents using K-means for a given K, here 35. Then the HAC procedure finds the nearest two clusters and merges them into a single parent cluster, replacing the two with this parent in the set. It then iterates this procedure, replacing the nearest two clusters, one or both of which could be a previously merged cluster, until only one cluster remains at the root. Using these hierarchical arrangements we can understand groupings of topics and sub-topics within the document collection. The labels assigned to the document clusters are either "P", representing parent, and a number or "C", representing original cluster, and a number. The number is a unique identifier for each cluster or parent. The number after the colon represents the number of documents contained in that cluster.

On the right hand side of the view is the bubble word cloud associated with the highlighted document cluster. As described previously, in Section III-A, the word cloud is generated by simply counting term frequencies in the document cluster. The terms that are easiest to see are the ones which are most frequent. There is no meaning to the spatial arrangement only to the size. The purpose of a wordcloud is to quickly summarize the content of potentially hundreds or thousands of documents contained in the selected cluster.

The view updates as a user of the application moves the mouse over various sections of the display. As he or she hovers over a user cluster, the other clusters fade out to emphasize the cluster in focus. Blue lines represent the communication pattern of the group and are connected from the user cluster center to the various document clusters the users in that cluster are talking about. The document cluster with the highest connection to that user cluster is highlighted and its word cloud is displayed. This is shown in the upper image of Figure 6.

In addition to hovering over user clusters, a user may also

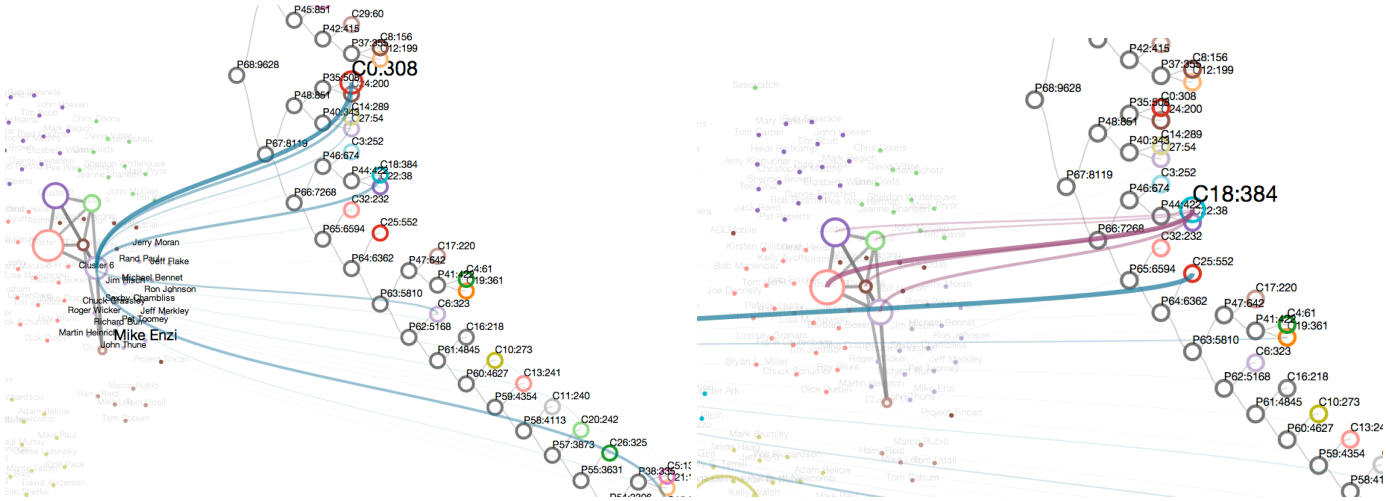

Fig. 6: The main view changes to highlight the group being hovered over with the mouse. The blue lines in the first figure represent the various document clusters the highlighted user cluster is talking about. As a user hovers over document clusters, in second image, the cluster is highlighted and the associated word cloud is shown. Purple lines are extended out from the document cluster to the user clusters talking about this set of documents.

hover over the document clusters directly. This will scale up the document cluster being hovered over and bring up its word cloud. It will also extend purple lines from the document cluster to each of the user clusters talking about this particular document collection. This is shown in the lower image of Figure 6. Note that neither the blue lines nor the purple lines disappear until a new bubble is hovered over. This is in order to maintain context as the user is moving around in the view.

Finally there are two search boxes on the upper left and upper right. The search box on the upper left allows one to find users either by partially matching the name or by complete matching. Users are highlighted by creating a large blue dot to replace the normally small dot connected with individuals. This can also be seen in Figure 6. The other search box will highlight a document cluster, the one with the highest frequency occurrence of the term in the search box. Document term search is by exact match. As before, highlighting the document cluster also brings up the associated word cloud and the purple lines back to the user clusters.

All of these interactions allow a user of the application to search through all of the data in the collection and globally understand this data in terms of which groups the users have collected into and what sorts of things they are talking about. If the user has a targeted query, such as a particular person or term, he or she can search for that directly.

IV. Results

The data set we chose to bootstrap our proof-of-concept application with consists two sets of data. One set of the public Tweets of professional Starcraft players, United States senators, E-learning experts and Hollywood celebrities. Because they did not opt-in as we would normally operate, we chose them based on the public nature of their roles and selected only the data they explicitly made public. We have a collection of 257 individuals and 546,570 tweets. It consists of all tweets made by the individuals from when they first joined Twitter up through June 2013. The other set is an anonymized group of players from an anonymized online game. This collection consists of 3,836 individuals who made 149,921 comments in the game's forum.

A. Parameterization

In order to parameterize the processing, we began by experimentally finding a number of topics to pass to pLDA which produced topic distributions that made sense. We started with a smaller base data set of only 10,000 documents to boot strap the process. We have found through the experimentation that the system works better with as few topics as we can reasonably choose. In addition to the number of topics, we also have to choose values for α and β inputs to pLDA. For β we chose 0.01 as recommended by Steyvers and Griffiths[19]. However, while they recommend setting α to $50/t$ where t is number of topics, we found that because tweets are small they are less likely to be distributed over as many topics, we treat the numerator as a prior estimate on the number of topics per document, and use $2/t$. In the processing the final data, we maintained this parameterization of pLDA, $2/t$, but set the number of topics, t to 100.

Next we determined the proper number of clusters to use for the documents and the users. We did this by measuring the performance of the clustering for all values of K up to some reasonable maximum, finding a measure of average similarity between the elements of the clusters[22], [17]. We wanted to pick a K that is as small as possible to allow generalization without going too low in the measure. For the 10,000 document set, we chose a value of 35 using this method. Because this process was much more expensive to run for the final data set, we extrapolated and tried a few different

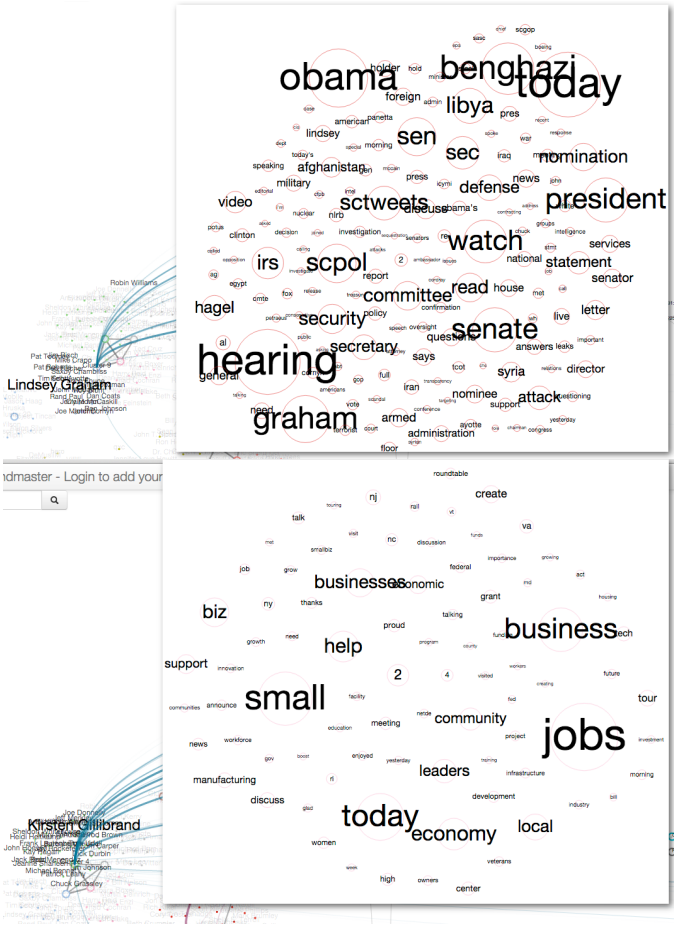

Fig. 7: A group of senators discussing the "Benghazi hearings" and a second group "small businesses".

K settling on 125 as a reasonable tradeoff for performance and comprehension. Because the number of users did not change between the two data sets, we chose a value of 20 for the 10,000 document set and used that same number for the final set.

B. A few examples

Although really understanding the full data sets requires interacting with it through the tool, we provide a few examples here to give a gist for how the data appears in the final results. We show a few examples of the kinds of groupings we can achieve with this system. It is important to note that the nature of this data is probabilistic and there is some noise and potential error with some users in the system. In part this can depend on the amount of data we have available, error rates would go down as we increase the number of documents we include as well as the increasing the number of individuals.

In the examples in Figures 7, we show two different groups of senators on Twitter. We chose the number of user clusters as described in Section IV-A based on the quality of the clustering. However, while we maintain these clusters as distinct, for the purposes of visualization we also keep track of how close, in topic space, those user clusters are to each other. Those which are close above some threshold, in this case above 0.5 on a scale of 0 to 1, we draw a light gray line between them and apply some force in the visualization to keep them together. Thus it is possible to see two to three larger groups form out of the smaller clusters.

Having these larger meta groups can help in navigating each individual group as we see in Figure 7. The word clouds prominently show one group is talking about "Benghazi hearings" and the second group is talking about "small business" and "jobs." While both of these groups talk about multiple topics of discussion, represented by the blue lines extending from each group, they do so with differing distributions. Taken individually we might not assume we are seeing a group of senators, but when we start looking at the two sub groups, as well the other two linked with these, along with each of their discussion clusters, we see that these are all political talking points and can infer their affiliation.

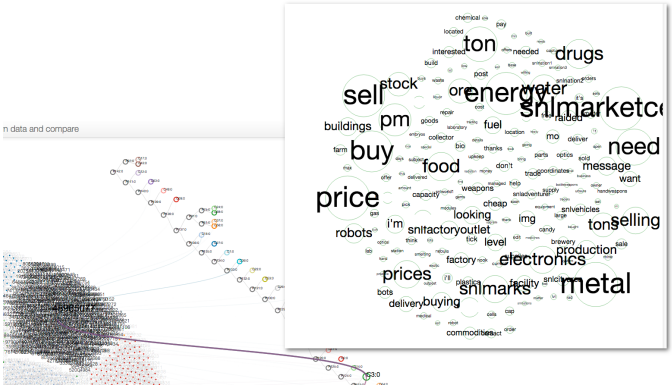

Fig. 8: Image shows an example from the game forum data set. It consists of a group entirely focused on discussing trades in resources of the game. They do not represent a strong interest in other aspects of the game such as wars, conflicts or game events.

In the second example, Figure 8, we see an example from the game forum data set. Here is a group of individuals interested in talking about trades and market information. The single blue line coming from this set is hidden by the purple line. We note that the distribution of purple lines shows that other groups talk about this information, but not to the degree of the main group, which is their exclusive topic as reflected by the single blue line. With this information we can find the affinity of groups and the distribution of topics that they are interested in, especially the difference in degrees between the distributions.

V. Conclusion

This work describes a proof-of-concept application for exploring the use of social media data to understand groups, individuals, and determine a group's affinities and interests as a collective. We have demonstrated a tool which is capable of collecting and processing social media free-form text and providing visualizations which aid a user to understand the whole corpus and the individuals in aggregate.

In future work we will explore the distribution of individuals among these groups to understand how individuals are distinct from the groups that they make up. We will also investigate the usefulness of an individual, the most central individual within a cluster. There is also the potential for understanding the temporality of the data. Here we have examined all the data over all time, but by looking at windows of time, we may see these data evolve and find the way topics, groups and individuals change and redistribute through the system.

VI. Acknowledgment

Funding for this work was provided by the Office of the Secretary of Defense through the Advanced Distributed Learning (ADL) Initiative. The views expressed are those of the authors and do not necessarily represent the view or policies of ADL.

Sandia National Laboratories is a multi-program laboratory managed and operated by Sandia Corporation, a wholly owned subsidiary of Lockheed Martin Corporation, for the U.S. Department of Energy's National Nuclear Security Administration under contract DE-AC04-94AL85000.

References

[1] Avondale web crawler, sandia.gov.

[2] Titan toolkit, titan.sandia.gov.

[3] F. Abel, Q. Gao, G.-J. Houben, and K. Tao. Semantic enrichment of twitter posts for user profile construction on the social web. In *The Semanic Web: Research and Applications*, pages 375–389. Springer, 2011.

[4] E. Bakshy, J. M. Hofman, W. A. Mason, and D. J. Watts. Everyone's an influencer: Quantifying influence on twitter. In *Proceedings of the Fourth ACM International Conference on Web Search and Data Mining*, WSDM '11, pages 65–74, New York, NY, USA, 2011. ACM.

[5] G. D. Battista, P. Eades, R. Tamassia, and I. G. Tollis. *Graph drawing: algorithms for the visualization of graphs*. Prentice Hall PTR, 1998.

[6] D. Blei, A. Y. Ng, and M. I. Jordan. Latent dirichlet allocation. *JMLR*, 3:993–1022, 2003.

[7] M. Bostock, V. Ogievetsky, and J. Heer. D3: Data-driven documents. *IEEE Trans. Visualization & Comp. Graphics (Proc. InfoVis)*, 2011.

[8] P. J. Crossno, A. T. Wilson, D. M. Dunlavy, and T. M. Shead. Topicview: Understanding document relationships using latent dirichlet allocation models. In *Interactive Visual Text Analytics for Decision Making Workshop*, 2011.

[9] A. W. et al. Text analysis tools and techniques of the pubmed data using the titan scalable informatics toolkit, 2011.

[10] B. S. Everitt, S. Landau, and M. Leese. Clustering analysis. *Arnold, London*, 2001.

[11] J. Golbeck, C. Robles, M. Edmondson, and K. Turner. Predicting personality from twitter. In *Privacy, security, risk and trust (passat), 2011 ieee third international conference on and 2011 ieee third international conference on social computing (socialcom)*, pages 149–156. IEEE, 2011.

[12] J. Hannon, M. Bennett, and B. Smyth. Recommending twitter users to follow using content and collaborative filtering approaches. In *Proceedings of the fourth ACM conference on Recommender systems*, pages 199–206. ACM, 2010.

[13] O. Kaser and D. Lemire. Tag-cloud drawing: Algorithms for cloud visualization. *arXiv preprint cs/0703109*, 2007.

[14] J. MacQueen et al. Some methods for classification and analysis of multivariate observations. In *Proceedings of the fifth Berkeley symposium on mathematical statistics and probability*, volume 1, page 14. California, USA, 1967.

[15] D. Quercia, M. Kosinski, D. Stillwell, and J. Crowcroft. Our twitter profiles, our selves: Predicting personality with twitter. In *Privacy, Security, Risk and Trust (PASSAT) and 2011 IEEE Third Inernational Conference on Social Computing (SocialCom), 2011 IEEE Third International Conference on*, pages 180–185. IEEE, 2011.

[16] A. Ritter, C. Cherry, and B. Dolan. Unsupervised modeling of twitter conversations. 2010.

[17] S. Salvador and P. Chan. Determining the number of clusters/segments in hierarchical clustering/segmentation algorithms. Technical report, 2003.

[18] M. A. Smith, L. Rainie, B. Shneiderman, and I. Himelboim. Mapping twitter topic networks: From polarized crowds to community clusters. *Pew Research Center*, 2014.

[19] M. Steyvers and T. Griffiths. Probabilistic topic models. *Handbook of latent semantic analysis*, 427(7):424–440, 2007.

[20] C. Sumner, A. Byers, R. Boochever, and G. J. Park. Predicting dark triad personality traits from twitter usage and a linguistic analysis of tweets. In *Proceedings of the IEEE 11th International Conference on Machine Learning and Applications*, 2012.

[21] C. Sumner, A. Byers, and M. Shearing. Determining personality traits and privacy concerns from facebook activity. In *Proceedings of the Black Hat Briefings '11*, 2011.

[22] R. Tibshirani, G. Walther, and T. Hastie. Estimating the number of clusters in a dataset via the gap statistic. 63:411–423, 2000.