Available online at www.sciencedirect.com

ScienceDirect

Procedia Computer Science 36 (2014) 152 – 159

Procedia
Computer Science

Complex Adaptive Systems, Publication 4
Cihan H. Dagli, Editor in Chief
Conference Organized by Missouri University of Science and Technology
2014- Philadelphia, PA

Measuring the Influence of Mass Media on Opinion Segregation through Twitter

Omar ElTayeby[a]*, Peter Molnar[b], Roy George[b]

[a,b]*Clark Atlanta University, 223 James P. Brawely Dr. SW, Atlanta, 30318, GA, USA*

Abstract

During the US presidential elections the media played a major role in presenting the candidates' vision on several topics. Nevertheless, the diversity of opinions along with the political currents, one might notice segregation in opinions among some topics related to each other or a candidate. In the meanwhile, posting opinions on social media could be represented as a sentiment vector towards multiple issues. This gives us a ripe ground for clustering opinions to view tweets that hold similar opinions. In this paper we investigate the media's influence on segregating opinions by constructing an aspect-based opinion mining framework. Our main task is to detect the segregated groups of opinions by solving the proposed model using expectation maximization (EM) algorithm. We examined a corpus of tweets collected, which are related to famous political topics. We show interesting observations on the sentiment used for particular topics among the groups of opinions, and conclude the percentages of media influences among the segregated groups of opinions with respect to these topics.

Keywords: Expectation Maximization; Apriori; Aspect-based Opinion Mining; Sentiment Analysis; Trending Topics; Opinion Segregation

1. Introduction

The extensive use of social media in the past few years has given scientists in different research areas tremendous amount of data to analyze and explore insights according to their different goals. Given the substantial increase in the use of social media, there are many companies that are spending their time and money to engage on social media and create a significant amount of content for propaganda and advertisement purposes. However, they also aim for users' feedback. The outcome of such time spent on social media and the information being generated, businesses have taken notice and are attempting to leverage the power of social media to help them succeed in understanding customers' needs. In the context of improving the aggregation of users' responses and feedbacks, measuring the influence enables monitoring the efficiency of such engagement. Thus, we collected tweets during the US presidential elections in 2012 to measure the influence of the mainstream media on users' opinions about multiple issues during the elections. We came up with an aspect-based opinion mining framework [1] to provide such measurement.

* Corresponding author. Omar ElTayeby
E-mail address: otayeby@gmail.com.

doi:10.1016/j.procs.2014.09.062

Individuals who follow news and with particularly high levels of disassociation with the media will frequently experience feelings of dissonance [2], and then make individual media selections that align with their own views and support their own perspectives. At the macro-level; one can see ideological consistency throughout society and across media outlets.

Quantifying the influence through Twitter could help us find the factor at which the segregated opinions resulting from such ideological consistency. Thus, we comprise our research questions to discover how do opinions generally on Twitter are spread over the spectrum of sentiment for multiple political issues. And how can we view them as herds or clusters of similar opinions? Or how can we group people holding similar opinion among some topics? Which leads us to a more specific one; among those clusters of opinions, are there any segregated unidimensional opinions? And how can we detect them? And how did the mass media influence those segregated opinions during the elections? And how much probability this influence is true?

The rest of the paper is organized as the following. In section 2, we present similar previous work in the media theory, sentiment analysis and opinion mining context. In section 3, we define our goal from which we deduce the main task and framework. In section 4, we propose the framework and describe each step in details. Specifically in section 4.3, we describe the how we fit Expectation Maximization (EM) into our model. In section 5, we show the observations from the experiments after implementing the framework processes on a sample of corpora collected from Twitter.

2. Related work

In 2003, DeMarzo et al. [3] proposed a boundedly rational model of opinion formation in which individuals are subject to persuasion bias. They showed that the persuasion bias implies the phenomenon of unidimensional opinions, which is defined as individuals' opinions over a multidimensional set of issues converge to a single “left-right” spectrum. They explored the implications of their model in several natural settings, including political science and marketing, and obtained a number of novel empirical implications.

In 2012, Seth Myers et al. [4] focused on both internal and external influence on social networks. In their model they distinguished between exposures and infections. Exposures to information lead to an infection. They developed an estimation technique from a given network and a set of node infection times. They infer the exposure curve that models the probability of infection as a function of the number of exposures of a node. They experimented with their model on Twitter and found that the occurrence external out-of-network events are detected accurately, and the exposure curve inferred from the model is often 50% more accurate than baseline methods.

Lars K. Hansen et al. [5] measured virality on Twitter in 2011. They hypothesized that negative news content is more likely to be retweeted, while for non-news tweets positive sentiments support virality. They analyzed three different corpora and concluded that the relation between affect and virality is more complex than expected based on the findings of Berger and Milkman [6], in 2012. They used the “AFINN” scoring list to analyze the sentiment, which we also used for our experiments. This list contains 2,477 words, which they rated with scores from -5 to +5.

3. Goal

In order to answer our research questions, we proposed a framework to measure how much probability the unidimensional opinions were affected by the mass media during the elections. This was simply achieved by calculating the percentage of news mentions in detected unidimensional opinions’ clusters. The method arise the question of how could we detect unidimensional opinions first. According to DeMarzo et al. [3] in 2003, they are individuals’ opinions over multidimensional set of issues which converge to a single “left-right” spectrum.

In order to detect the unidimensional opinion clusters we propose a probabilistic model, where the general case of opinion influence is considered to come from various sources, like friends, family, coworkers etc… The sources of influence are the latent factors (hidden variables) and the observed variable is the sentiment captured from the tweets (the incomplete data case). We use the EM algorithm to assign each tweet representing an opinion to a cluster that contains other opinions which are more likely to be similar and/or affected by the same popular sources. The clusters do not represent a special type of influenced opinions; instead we just use them to detect the unidimensional ones and calculate the percentage of news mentions in those clusters. We focus on the unidimensional clusters, since they are the groups that pay attention to some specific topics with extreme range of sentiment compared to other groups.

The undimensional clusters are isolated, on the sentiment scale, from the rest of the clusters, due to the convergence property they exhibit among the spectrum. Such isolation is detected from the EM's output by finding the nonintersecting clusters on the scale. The EM's output are the clusters' parameters, which are the mean and the standard deviation. Using both parameters we calculate the range's minimum and maximum of the sentiment spanned by each cluster with respect to each different topic.

The goal leads us to represent the tweets' sentiment in the form of a sentiment matrix, which is the input to the EM algorithm. The sentiment matrix's rows are the tweets and the columns are the topics. The sentiment assignment totally depends on the adjective used in the tweets towards different topics. The score is only assigned to the topics mentioned, while the others topics are assigned zero as neutral value.

4. Frame work

The framework we propose is composed of three main steps: finding trending topics, assigning the sentiment and clustering (EM) the sentiment matrix. Figure 1 shows the aspect-based opinion mining framework, where the first two steps aim for constructing the sentiment matrix to be ready as an input to the EM algorithm.

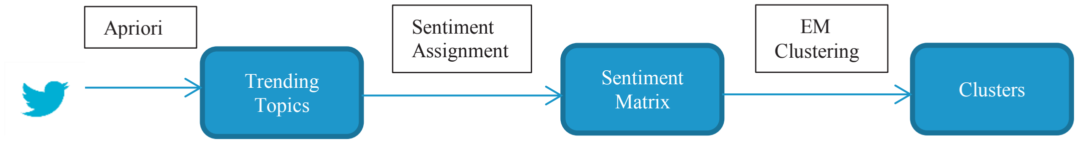

Fig. 1. The aspect-based opinion mining framework

Trending topics are presented by Twitter on the main Twitter.com site, which are tailored to each user according to the users' interests. The list is updated every few minutes as the new topics become popular. However, the popularity is mostly measured using the frequency of hashtags, which does not fulfill our requirements.

Our analysis towards the trending topics is only tailored to the collected tweets. We stream in tweets using certain political keywords. The list of the most frequent hashtags was then used to be the candidates for the trending topics. We used Apriori to find the association rules and the frequent itemsets between those hashtags. The Apriori algorithm is applied in different domains to mine the frequent itemsets and the association rules of items related to each other through transactions, like the market basket analysis problem. The most important advantage out of finding the frequent itemsets is reducing the sparsity of the sentiment matrix, in order to find more separated clusters. Each tweet is represented by a combination of the hashtags. The Apriori algorithm alternates between two steps, filtering and pruning, to find the frequent itemsets incrementally starting from one till the pruning step finds no more candidates for the next larger frequent itemset. The filtering step filters out candidates of frequency lower than the support count, while the pruning step prunes out the suggested candidates that has no credibility a priori to be in the future frequent itemset [7]. We calculated the support count using the average counts of the candidate hashtags.

In the sentiment analysis step we downloaded the "AFINN" list from[a] and used the Natural Language Processing Toolkit (NLTK)[b] for tokenizing and tagging the sentences which takes place in this method when the adjectives are compared with the scoring list. The score of the adjective is assigned to the topics mentioned in the vector. The score of the adjective assigned refers to all of the topics mentioned.

4.1. Clustering opinions

[a] Finn Nielsen. DTU Compute. http://www2.imm.dtu.dk/pubdb/views/publication_details.php?id=6010 (accessed February 15, 2014).
[b] Natural Processing Toolkit (NLTK). http://www.nltk.org/ (accessed July 24, 2014).

The goal of the EM clustering step is to assign each tweet to cluster that contains other tweets of partially similar sentiment towards the same topics. Those assignments conclude that tweets of the same cluster have also been affected by the same anonymous mixed sources of news or opinions. In the next paragraph we explain how our probabilistic model is solved by EM algorithm using a simple example, and then modify this example into the actual model.

4.2. Opinion tracking example

Consider a simple opinion tracking experiment for two Twitter pages of two news channels A and B with unknown biases θ_A and θ_B respectively, towards a single topic. This opinion tracking example is a modified version of the coin-flipping example that was given in [8] to fit our field of application and explain how we fit the EM algorithm to solve the model proposed. We define θ_A as the probability that channel A on its Twitter page would use positive sentiment towards a certain topic t. Our goal is to estimate $\theta = (\theta_A, \theta_B)$, which can be easily obtained by repeating the following steps five times, so the entire procedure involves a total of 50 tweets analyzed (Table 1):

1. Randomly choose one of the two channels
2. Perform ten independent sentiment analyses on the tweets from the chosen channel about a single topic t.

Table 1. The complete case of the opinion tracking experiment.

Channel ID	Sentiment of 10 tweets	Channel A's sentiment counts	Channel B's sentiment counts
B	+, -, -, -, +, +, -, +, -, +		5 +, 5 -
A	+, +, +, +, -, +, +, +, +, +	9 +, 1 -	
A	+, -, +, +, +, +, +, -, +, +	8 +, 2 -	
B	+, -, +, -, -, -, +, +, -, -		4 +, 6 -
A	-, +, +, +, -, +, +, +,-, +	7 +, 3 -	
Total sentiment counts		24 +, 6 -	9 +, 11 -

Thus during the experiment we keep track of two vectors representing the "Complete dataset case". Vector $x = (x_1, x_2, \dots, x_5)$ is the number of positive sentiment observed during the i_{th} set of tweets, where $x_i \in \{0,1,\dots,10\}$. While vector $z = (z_1, z_2, \dots, z_5)$ is the identity of the channel used in i_{th} set of tweets analysed. Parameter estimation in this setting is known as the complete data case, where the values of all relevant variables in this model (the sentiment towards the topic and the composer for each set of tweets) are known. A simple way to estimate θ_A and θ_B is to return the observed proportions of positive sentiment for each channel, using the estimators $\hat{\theta}_A$ and $\hat{\theta}_B$:

$$\hat{\theta}_{A\ or\ B} = \frac{\#\ of\ poistive\ sentiment\ posted\ by\ channel\ A\ or\ B}{total\ \#\ of\ tweets\ posted\ by\ channel\ A\ or\ B\ about\ topic\ t} \qquad (1)$$

In statistical literature, this intuitive guess is known as "Maximum Likelihood Estimation (MLE)". This method values the quality of a statistical model based on the probability it assigns to the observed data. Since the intuitive guess is limited to the sample of tweets collected, we want to find the best estimation as if we have an infinite sample (mathematically), which is practically equivalent to all of the population's opinions. Thus, the model is suitable to be applied on tweets, which is a limited to a very small sample of the whole populations' opinions that is impossible to collect and not explicitly accessible by the public (except on Twitter). Later, we will discuss this actual model. From this small sample we maximize $logP(x, z; \theta)$, where $P(x, z; \theta)$ is the joint probability for obtaining any particular vector from the observed positive sentiment counts x of the source channel identity z, to find the parameters $\hat{\theta} = (\hat{\theta}_A, \hat{\theta}_B)$ which are solved by the above formulas. For the example given in table 1 the parameters are as following:

$$\hat{\theta}_A = \frac{24}{24+6} = 0.80, \qquad \hat{\theta}_B = \frac{9}{9+11} = 0.45$$

Now consider a more challenging variant of the parameter estimation problem in which we are given the recorded positive sentiment counts x but not the identities z of the channels that posted each set of the tweets. We refer to z as hidden variables or latent factors, which in our model represent the source of sentiment that we want to reveal. Parameter estimation in this new setting is known as the incomplete data case. This time, computing proportions of positive sentiment for each channel is no longer possible, because in this setting we assume do not know the source of the tweet. However, if we had some way of completing the data (guessing correctly which channel posted in each set of the tweets), then we could reduce parameter estimation for this problem with incomplete data to maximum likelihood estimation with complete data.

One iterative scheme for obtaining completions could work as follows: starting from some initial parameters, $\hat{\theta}^{(t)} = \left(\hat{\theta}_A^{(t)}, \hat{\theta}_B^{(t)}\right)$ determine for each of the five sets whether channel A or channel B was more likely to have posted the observed tweets (using the current parameter estimates). Then, assume these completions (guessed channel assignments) to be correct, and apply the regular maximum likelihood estimation procedure to get $\hat{\theta}^{(t+1)}$. Finally, repeat these two steps until convergence. As the estimated model improves, so too will the quality of the resulting completions.

The EM algorithm is a refinement on this basic idea. Rather than picking the single most likely completion of the missing channel assignments on each iteration, the EM algorithm computes probabilities for each possible completion of the missing data, using the current parameters $\hat{\theta}^{(t)}$. These probabilities are used to create a weighted training set consisting of all possible completions of the data. A modified version of maximum likelihood estimation that deals with weighted training examples provides new parameter estimates, $\hat{\theta}^{(t+1)}$. By using weighted training examples rather than choosing the single best completion, the EM algorithm accounts for the confidence of the model in each completion of the data (Figure 2).

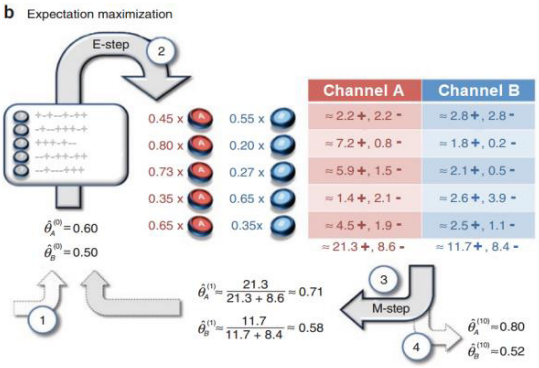

Figure 2. The incomplete case of the opinion tracking experiment.

4.3. Actual model

In our actual model, we modify the previous simple example to emulate the real influence, according to the following assumptions:

- The tweets tested are assumed to be influenced by multiple news channels and other sources (i.e. friends, family, coworkers, etc…), instead of testing the Twitter pages of direct sentiment from two news channels only
- The sentiment is range of values instead of binary values (positive and negative) only
- The sentiment captured for each tweet is towards multiple different topics instead of one only

This leads us to represent the unknown biases as θ_{ks}; the probability that influencer k influenced a user to be biased towards a topic t by a sentiment s. Here we consider influencer k as a mixture of influencers who lead to such constructive opinion represented in the tweet towards the T number of topics tested. Since the sentiment range is not binary now, there are S numbers of sentiment values that can be assigned to a tweet.

If we repeat the same simple experiment with the new settings we have now, then we should keep track of 2 vectors: $x_s = (x_{s_1}, x_{s_2}, \dots, x_{s_5})$, which builds up a matrix for all different counts of tweets with sentiment s, where also $x_{is} \in \{0,1, \dots ,10\}$. While the identity vector of tweets from the K influencers stays the same and known, except that $z_i \in \{1, \dots, K\}$, not only from two channels. A simple way to estimate θ_{ks} is to return the observed proportions of sentiment s for each influencer k, using the estimators $\hat{\theta}_{ks}$, this formula also solves the parameter θ_{ks} that maximizes $logP(x_s, z, \theta)$:

$$\hat{\theta}_{ks} = \frac{\#\ of\ tweets\ with\ sentiment\ s\ influenced\ by\ k}{total\ \#\ of\ tweets\ analyzed\ from\ influencer\ k} \qquad (2)$$

When transferring to the incomplete dataset case to obtain completions, the EM algorithm in Weka is initialized with the parameters $\hat{\theta}^{(t)} = [\begin{matrix} \hat{\theta}^{(t)}_{1s_1} & \cdots & \hat{\theta}^{(t)}_{1S} \\ \vdots & \ddots & \vdots \\ \hat{\theta}^{(t)}_{Ks_1} & \cdots & \hat{\theta}^{(t)}_{KS} \end{matrix}]$ by choosing the best result out of 10 executions of Simple K-Means of different seed values. Weka's implementation for EM assumes Gaussian mixtures with diagonal covariance matrices as the initial distribution function. The algorithm repeats the same steps mentioned earlier in the simple example, and each estimator $\hat{\theta}^{(t)}$is replaced by $\hat{\theta}^{(t)}_{ks}$ to fit the actual model's new equations.

5. Experiments

Under the administration of Professor Peter Molnar we harvested the tweets using a stream that was active since September 2012. The Twitter project was established on a fedora server at Clark Atlanta University. The website hosts all detailed information at the fedora website at[c], including the streaming keywords used to collect the tweets. The 140dev API framework was used to stream in the tweets, which is a free source code library written by Adam Green[d] and released under the General Public License (GPL).

We filter out the retweets (RTs; reposts) as part of cleaning the data from duplicates, unlike Myers Seth et al., since our scope is focused on finding the influence by comparing the sentiment of original tweets. Basically, it is worthless to analyze opinions which contain all zero sentiment vector, and that could result from either no adjective used or no trending topic mentioned. And thus, finding the frequent itemsets plays its role in reducing matrix sparsity, so when the sentiment is assigned to a topic we guarantee with high probability that the tweet would contain many topics. This is also the same reason we use the adjective filter to decrease the sparsity of the sentiment matrix used in the opinion clustering step. Nevertheless, we exclude the tweets which have more than one adjective, since we did not handle multi-sentiment tweets. We do not apply any technique to differentiate the reference of each adjective in a multi-sentiment tweet.

To find the trending topics tailored to our collected dataset we extracted all hashtags as the candidate topics then input them to the Apriori algorithm with the tweets. We found (275, 9735) frequent itemsets out of 7,812 hashtags, which were in 10 million tweets. We eliminated out the itemsets which contain only numbered hashtags, since they could associate other irrelevant alphanumeric hashtags. The resulted list of topics is as the following list of 30 words:

[obama, usa, tcot (Top Conservatives On Twitter), p2 (Progressive Propaganda), news, cnn, romney, teaparty, tiot (Top Independents On Twitter), usopen, dnc (Democratic National Committee), teamfollowback or tfb (you will follow back), economy, election, iran, israel, job, media, navy, nyc (New York City), ows (Occupy Wall Street), politics, twisters, usopen (Tennis Championship), vote, jakarta, london, politics, republican, fl(Fruity Loops studio)]

There are repeated entities being expressed by different hashtags like "mittromney" and "mitt". We combined those hashtags as the same using part of word searching, and the matches are recognized as the same entity. We used the tagdef website[e] to define the short hashtags.

[c] Peter Molnar. The Twitter Project. http://fedora.cis.cau.edu/~pmolnar/TWITTER/ (accessed February 15, 2014).
[d] Adam Green. 140Dev. http://140dev.com/ (accessed February 15, 2014).
[e] #tagdef. http://tagdef.com/ (accessed February 15, 2014).

5.1.1. Clustering (EM)

When the adjective matches one of the words in the AFINN list, the corresponding score is assigned. We filtered out the RTs, and tweets that did not mention topics, adjectives and/or news channels.

By using the “training set” (the default evaluation choice); Weka classifies the training instances into clusters according to the cluster representation and computes the percentage of instances falling in each cluster after generating them. This setting has resulted into 5 clusters, where it took Weka 10.43 seconds. Table 2 shows the distribution of the tweets among the clusters. We present the distribution of the sentiment towards each topic among the clusters using the minimum and maximum. The minimum and maximum were calculated using the mean and standard deviation of the results. We red mark the nonintersecting clusters, which express segregation from other clusters. While Table 3 shows the number and percentage of tweets which mentioned the news channels among the isolated clusters and all clusters. In both tables we show topics which have isolated clusters only.

Table 2. Minimum and maximum of clusters’ sentiment towards topics which showed segregated opinions

Topic	Cluster 0	Cluster 1	Cluster 2	Cluster 3	Cluster 4
OWS (Occupy Wall Street)					
Min	-0.109	+0.82365	-0.2798	+0.0966	-0.38045
Max	+0.221	+1.80035	+0.2798	+0.5676	+0.15225
Romney					
Min	+0.3636	+0.3211	-0.07445	+0.68925	-1.8223
Max	+1.6556	+1.3571	+1.57705	+1.08635	-0.5339
Obama					
Min	+1.37935	+1.65555	-0.0416	+0.8786	-2.75655
Max	+2.58725	+2.14185	+1.7602	+1.1206	-2.06565

Table 3. Number and percentage of news channels mentioned among the isolated clusters and all clusters

Topic & Sentiment score range	ABC	NY times	Fox	CNN	Reuters	NBC	Total
OWS (Occupy Wall Street) > +1							
All Clusters	0	0	1	21	8	2	32 (10.3%)
Cluster 1 only	0	0	1	6	8	1	16 (5.1%)
Romney < -1							
All clusters	6	1	10	55	12	7	91 (29%)
Cluster 4 only	2	0	1	13	9	2	27 (8.7%)
Obama < -2							
All clusters	12	3	19	105	17	19	175 (56.6%)
Cluster 4 only	5	0	2	29	11	10	57 (18.4%)
Obama > +1							
All Clusters	18	6	27	187	31	32	301 (97%)
Cluster 0 & 1 only	5	0	2	29	11	10	57 (18.4%)

From Table 2, we observe that cluster 1 expresses segregation in opinion towards the “Occupy Wall Street” (OWS), where the range of sentiment used by this cluster is between 0.82365 and 1.80035. From Table 3, we conclude that the media influenced the segregated opinions (cluster 1) to be positive about Occupy Wall Street with words of scores more than +1 by a probability of 5.1%, and influenced all clusters by a probability of 10.3%. From Table 2, cluster 4 expressed segregation in opinion towards the topic “Romney,” where the range of sentiment used by this cluster is between -1.8223 and -0.5339. From Table 3, we conclude that the media influenced the segregated opinions (cluster 4) to be negative about Romney with sentiment less than -1 by a probability of 8.7% and influenced all clusters by a probability of 29%. Lastly, clusters 0, 1 and 4 express interesting isolation in the opinion towards the topic “Obama”. From Table 2, cluster 0 and 1 are isolated together in the positive region between scores 1.36935 and 2.58725 and cluster 4 is isolated in the negative region between -2.75655 and -2.06565. From Table 3, we conclude that the media influenced segregated opinions (clusters 0 & 1) of score more than +1 by a probability of 18.4% and all opinions by a probability of 97%. Also, the media influenced the other segregated opinions (cluster

4) to be negative about “Obama” with scores less than -2 by a probability of 18.4%, and influenced all clusters by a probability of 56.6%.

For clearer image about the isolated clusters Figure 3 shows simple error bar graph plots of the range ofscores spanned by the clusters towards the three topics. The probabilities stated in our conclusions is based on considering tweets which have negative sentiment less than the maximum score, and positive sentiment more than the minimum score, of the isolated clusters with respect to each particular topic, since they exhibit the segregation among other tweets.

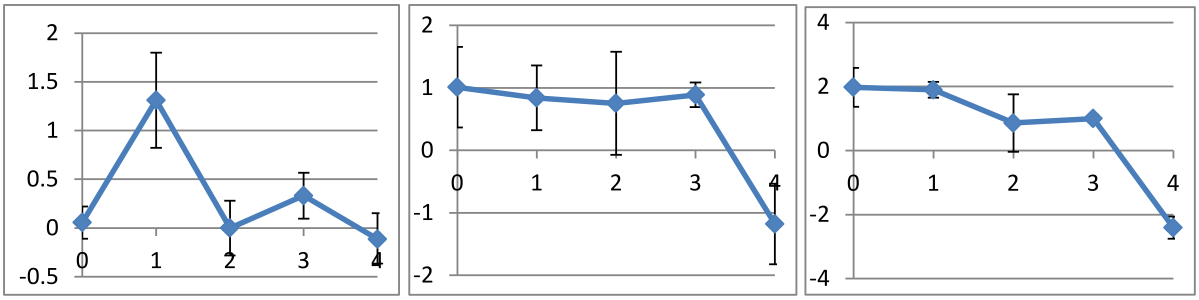

Figure 3. The range of clusters’ sentiment towards topics “OWS”, “Romney” and “Obama” from left to right respectively. In each graph the x-axis represents the cluster number and the y-axis represents the sentiment score. The error bar represents the range of sentiment score.

6. Summary

In conclusion, we proposed the challenge of measuring the influence of mainstream media on Twitter users. Our main interest was to focus on segregated groups according to each topic. We counted the number of news channel mentions among the sentiment spanned by the segregated groups, whether the tweets were in or out of those segregated groups. We constructed an aspect-based opinion mining framework to detect the segregated groups by first finding the trending topics using Apriori algorithm, then constructing the sentiment matrix using the AFINN scoring list, and lastly clustering the tweets represented in the sentiment matrix.

In our future work we aim to collect the sentiment of the same user account towards multiple topics instead of only each tweet. This will allow us to take RT into consideration, since each account will be a separate entity to be forming a unique profile of opinion. Also, to solve the problem of multi-sentiment tweets using the sentiment structure to differentiate between the sentiments of each topic in the same tweet. This will enable the framework to come up with more accurate results than the current instead of ignoring those tweets.

References

1. Blair-Goldensohn S, Hannan K, McDonald R, Neylon T, Reis G, Reynar J. Building a sentiment summarizer for local service reviews. In WWW Workshop on NLP in the Information Explosion Era; 2008. p. 14.

2. D'Alessio D, Allen M. Selective Exposure and Dissonance after Decisions. Psychological Reports 91; 2002. p 527–32.

3. DeMarzo P, Vayanos D, Zwiebel J. Persuasion bias, social influence, and unidimensional opinions. The Quarterly Journal of Economics 2003; 118.3: 909-968.

4. Myers S, Zhu C, Leskovec J. Information diffusion and external influence in networks. Proceedings of the 18th ACM SIGKDD international conference on Knowledge discovery and data mining. ACM; 2012.

5. Hansen L, Arvidsson A, Nielsen F, Colleoni E, Etter M. Good friends, bad news: affect and virality in twitter. In Future information technology. Springer Berlin Heidelberg; 2011. p 34-43.

6. Berger J, Milkman K. What makes online content viral?. Journal of Marketing Research 49.2; 2012. p 192-205.

7. Xindong W, Zhang C, Zhang S. Efficient mining of both positive and negative association rules. ACM Transactions on Information Systems (TOIS) 22.3; 2004. p 381-405.

8. Do C, Batzoglou S. What is the expectation maximization algorithm? Nature biotechnology 26.8; 2008. p 897-900.