Delineating Real-Time Events by Identifying Relevant Tweets with Popular Discussion Points

Muhammad Asif Hossain Khan
Graduate School of IST
The University of Tokyo, Tokyo, Japan
asif@mcl.iis.u-tokyo.ac.jp

Guangwen Liu
Center for Spatial Information Science
The University of Tokyo, Tokyo, Japan
liugw198209@mcl.iis.u-tokyo.ac.jp

Danushka Bollegala
Graduate School of IST
The University of Tokyo, Tokyo, Japan
danushka@iba.t.u-tokyo.ac.jp

Kaoru Sezaki
Center for Spatial Information Science
The University of Tokyo, Tokyo, Japan
sezaki@iis.u-tokyo.ac.jp

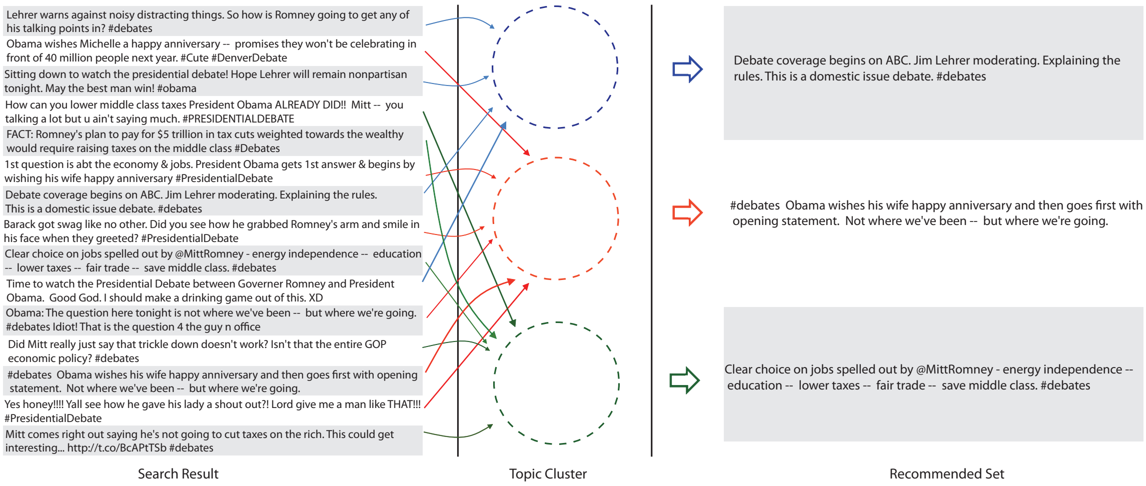
ABSTRACT

Twitter is increasingly becoming an ideal platform for getting access to firsthand real-time facts on the ground. Popular real-time events, which take place within a predefined period of time, causes upsurge of traffic in Twitter. During the event, a Twitter search user who puts event-related keywords/hashtags as a query string hopes to get a brief idea about the important occurrences of the event from the top search results. However, traditional approach of reverse chronological ordering of tweets that satisfy the Boolean query hardly meets that information need. In this paper, we propose an unsupervised method for recommending search users a set of tweets that best delineate an ongoing public event. The proposed graph-based retrieval algorithm is based on a hypothesis that the discussion points that are common among majority of event-relevant tweets are motivated by the important occurrences of the ongoing event. Hence, by identifying the popular discussion points in a collection of event-relevant tweets, and retrieving tweets comprising those discussion points, it is possible to outline real-time events. We further perform topical clustering on the relevant tweets before applying the retrieval algorithm on each topic cluster, so that, users interested in a particular aspect of the event can dig deeper into the search results returned for that particular cluster. Evaluation performed on about 270,000 relevant tweets generated during a real-world event reveals that, the tweets recommended by the proposed model could delineate the proceeding of the event with high precision and recall and could also outperform two intuitive and competitive baseline models.

I INTRODUCTION

Twitter is a micro–blogging site which allows its users to post 140–character messages, called *tweets*. It is increasingly becoming an ideal platform for getting access to firsthand real–time facts on the ground and listening to what people have to say about real-time events. Popular real-time events with different possible outcome, for example, real-time sports events such as mens' final in French Open, live public debates such as Presidential debates, live telecasted popular award ceremonies such as Academy Award etc. cause upsurge of traffic in Twitter while the events are taking place. These tweets range from small description of what is happening in real-time, highlights of important occurrences so far, personal comments and so on. A large user group has evolved who seeks these live updates [1]. Twitter's search users use event-specific keywords and/or hashtags for retrieving *event-relevant* tweets. Now a days it has become a norm for the public events to publish a set of hashtags well before the actual event commences. Twitter provides a search interface to its users to get access to the most recent tweets containing the terms in the search query. Recently, it has started to address the issue of relevance through incorporation of so called *resonance signal* and social graph of the search user into their relevance algorithm [1]. In essence, in response to a search query, relevant tweets (satisfying the Boolean query) are still presented in reverse chronological order with those with higher resonance signal (re-tweeted many times) and/or posted by someone from the quester's social graph are placed higher in the relevance order. Though, these modifications have undoubtedly improved the search experience of users, there is still room for improvement.

A search user putting the event related keywords or hashtags as the query string while the event is taking place has different information need than what a reverse chronological ordering of tweets satisfying that Boolean query can offer. It is reasonable to assume that he/she would like to see tweets discussing about what is happening in the event or has taken place thus far. Ongoing bantering among fervent supporters of the participants of the event, admiration or profanation regarding participants'



Lehrer warns against noisy distracting things. So how is Romney going to get any of his talking points in? #debates

Obama wishes Michelle a happy anniversary -- promises they won't be celebrating in front of 40 million people next year. #Cute #DenverDebate

Sitting down to watch the presidential debate! Hope Lehrer will remain nonpartisan tonight. May the best man win! #obama

How can you lower middle class taxes President Obama ALREADY DID!! Mitt -- you talking a lot but u ain't saying much. #PRESIDENTIALDEBATE

FACT: Romney's plan to pay for $5 trillion in tax cuts weighted towards the wealthy would require raising taxes on the middle class #Debates

1st question is abt the economy & jobs. President Obama gets 1st answer & begins by wishing his wife happy anniversary #PresidentialDebate

Debate coverage begins on ABC. Jim Lehrer moderating. Explaining the rules. This is a domestic issue debate. #debates

Barack got swag like no other. Did you see how he grabbed Romney's arm and smile in his face when they greeted? #PresidentialDebate

Clear choice on jobs spelled out by @MittRomney - energy independence -- education -- lower taxes -- fair trade -- save middle class. #debates

Time to watch the Presidential Debate between Governer Romney and President Obama. Good God. I should make a drinking game out of this. XD

Obama: The question here tonight is not where we've been -- but where we're going. #debates Idiot! That is the question 4 the guy n office

Did Mitt really just say that trickle down doesn't work? Isn't that the entire GOP economic policy? #debates

#debates Obama wishes his wife happy anniversary and then goes first with opening statement. Not where we've been -- but where we're going.

Yes honey!!!! Yall see how he gave his lady a shout out?! Lord give me a man like THAT!!! #PresidentialDebate

Mitt comes right out saying he's not going to cut taxes on the rich. This could get interesting... http://t.co/BcAPtTSb #debates

Debate coverage begins on ABC. Jim Lehrer moderating. Explaining the rules. This is a domestic issue debate. #debates

#debates Obama wishes his wife happy anniversary and then goes first with opening statement. Not where we've been -- but where we're going.

Clear choice on jobs spelled out by @MittRomney - energy independence -- education -- lower taxes -- fair trade -- save middle class. #debates

Figure 1: General framework of the proposed model. From left, search result returned by traditional social search engine is clustered into topical clusters and the proposed retrieval algorithm is applied on each cluster to recommend tweets incorporating popular discussion points among tweets in that cluster

past acts, ardent request for support etc. might not be what he/she would like the search results to be flooded with, though they might satisfy the criteria of recency. Again, as hundreds of thousands of tweets are generated during a live public event, relevant tweets should be arranged based on some criteria in addition to recency. There are often too many relevant tweets stating the same points in a slightly different way. Hence, for finding new content, a search user is forced to read scores of uninformative tweets, causing frustration [2]. Though twitter traffic is increasing at an exponential rate, users are still not prepared to go beyond the first few tens of recommended search results. The existence and increasing popularity of Twitter-based third-party services such as *Chirpstory*[1] or *Trendsmap*[2] substantiate that people aspire to filter uninformative tweets from their search feed and see the tweets of their interest in a more organized way. People are even happy to do this organization manually, as it is done in *Chirpstory*. Hence, a system that would automatically identify and recommend reasonably limited number relevant tweets delineating important occurrences of a real-time public event, would satisfy information need of a large user group.

Again, clustering the relevant tweets according to the topic of their discussion points and recommending tweets with popular discussion points from each cluster separately have manifold advantages. First, this diversification of recommended tweets enhances the coverage of the event's proceeding. Our evaluation substantiates this claim. Second, search users would be able to deeply explore the cluster containing only those tweets that discuss their topic of interest. When a twitter feed is overwhelmed by a large set of informative tweets, topical clustering can enhance the search experience of the user [3]. Bernstein et al. [4] also reported that active Twitter users found topic-based browsing more efficient and enjoyable than standard chronological interface. Finally, the retrieval algorithm that identifies tweets to be recommended from each cluster has to deal with less diversified collection of tweets.

In this paper, we propose a method for recommending the search users a set of tweets that best delineate the proceeding of a public event while the event is taking place. This method would be effective for those events which satisfy certain criteria such as a) it can elicit large response from Twitter users and, b) it is a real-time event which takes place within certain defined period of time. As different users might be interested in different aspects or sub-topics of an event, we first group the relevant tweets into topical clusters. Then within each cluster, we retrieve tweets that comprise of the concepts discussed in majority of tweets in the cluster and recommend the users a set of tweets that best match this criterion. Hence, the collection of tweets recommended from all the clusters cover different aspects of the event and together they give users clear ideas about the important moments in the event. Figure 1 shows the general framework of the proposed model. Our experimental evaluation shows that this strategy could effectively delineate important occurrences in a real-time event using a limited set of recommended tweets.

[1] http://chirpstory.com/
[2] http://trendsmap.com/

Our contributions are as follows:

- We present an algorithm that uses the information in the tweet collection to retrieve a set of tweets that best represent the main discussion points in the collection.
- Moreover, as the retrieval algorithm is totally unsupervised and needs no prior knowledge about the event, it can be used for any event that satisfies the aforementioned criteria.

It is to be noted that the work we present here is an extended version of our previous work [5]. We have added the following extensions to our previous work in this paper:

1) We have done an analysis on the stability of the proposed model by varying the number of topical clusters.
2) We have studies the impact of increasing the number of tweets in the recommended set on the achieved precision and recall.
3) We have presented the proposed algorithms in thorough detail.
4) We have developed and presented a user interface for the proposed model, which conveys the way we envision to leverage the proposed method in a functional system for summarizing important moments of ongoing events in real-time.

II RELATED WORK

Numerous efforts for characterizing an event using relevant tweets have been made by researchers in recent years. Chakrabarti et al. [6] proposed a method for summarizing highly structured and recurring events such as football matches. They assumed that new events had already been detected by some other methods. Their proposed method tried to extract a few tweets that best describe the interesting occurrences in the event. They trained an HMM based model to identify occurrences of sub-events based on tweet "activity threshold" within a time segment. For retrieving tweets that are close to other tweets in the corpus they used a "*tf-idf* with *cosine similarity*" based model that we have used as a baseline model in this paper. Availability of highly structured recurring events is scant in reality and hence their approach would not be able to handle vast majority of real-world events. Our proposed model does not rely on the structure or recurrence of events. Moreover, our method is completely unsupervised. On top of that, we have shown in the evaluation section that our proposed tweet retrieval algorithm outperforms the *tf-idf-cosine* model.

Sharifi et al. [7] also proposed a method for microblog summarization. Their model outputs a single sentence that serves as a journalistic summary of the event. They proposed two models for measuring relevance of co-occurrences in the tweets — one similar to the *tf-idf-cosine* model used in [6] and the other is a graph based model. The later model makes a graph of words around the "key phrase" based on the top N tweets returned by Twitter given the same *key phrase* as query. Their method returns a single sentence as a summary of the corpus. They used a frequency based approach to rank collocations around the key phrase and picked up tweets containing longest phrase obtained in this way. Our proposed model also identifies word co-occurrences that are popular among the relevant tweets and in the "Proposed Method" section we have analytically shown that the proposed method can distinguish co-occurrences with higher association strength better than the frequency based approach. Nichols et al. [8] used a slight variation of the phrase graph model proposed in [7] to generate a three-sentence summary for important moments in an event. Sudden upsurge of tweet traffic is used for detecting important moments. Finally, they added up the scores of each phrase encountered in the longest sentence of a tweet for obtaining a tweet score and output the top three tweets with the highest score.

Hu et al. [9] used a topic model to extract sense from Twitter feed relevant to public and televised events. Their model enables auto-segmentation of the events and characterization of tweets into two categories: *episodic* and *steady* tweets. However, they need a transcript of the event to acquire topical knowledge about the event, which can only be obtained after the event. Hence, their model serves as a post-event analysis tool that can measure how much attention each segment of a public event received in Twitter feed. In contrast, our method needs no transcript of any sort of external knowledge about the event and we identify the relevant tweets while the event is taking place.

Some efforts have been made for generating visual summaries of tweets on a topic [10], [11]. However, they do not offer sentence-level summaries and their recommended word clouds or word labels must be interpreted by users themselves.

A common requirement in [6] and several other works on summarization is the need to detect important moments in the tweet collection by some third party system. Our proposed solution has no such prerequisites. Hence, unlike our approach none of the aforementioned endeavors propose an unsupervised model which requires no external knowledge about the event to group event-relevant tweets in topical clusters and retrieve tweets that comprise popular discussion points within each cluster so that the recommended tweets serve as a journalistic summary of the event while the event is taking place.

III PROBLEM SETTING

In this paper we are addressing the problem of identifying a set of tweets relevant to some specific event that can best delineate the ongoing event. Our proposed solution is based on the assumption that the discussion points that are common among majority of the relevant tweets are motivated by the proceedings of the event. If this hypothesis is true, then the problem transforms into finding the tweets that comprise maximum number of common points discussed in the majority of tweets. Evaluation performed on a real-world event and its relevant tweets substantiate that the hypothesis holds. The proposed model could achieve up to 80% recall with 81.6% precision for some debate segments while trying to delineate the proceeding of a US presidential debate using relevant tweets.

Formally, given a collection of n event-relevant tweets $\mathcal{T} = \{t_1, t_2, \dots t_n\}$, we extract a set $\mathcal{C} = \{\mathcal{C}_1, \mathcal{C}_2, \dots \mathcal{C}_k\}$ of k topic clusters; i.e. tweets in $\mathcal{T}$ are partitioned into the k clusters in a way such that for all $i \neq j$, $\mathcal{C}_i \cap \mathcal{C}_j = \emptyset$. We then learn a scoring function s, such that for tweets $t_i, t_j \in \mathcal{C}_l$, if t_i comprises more popular points discussed in the rest of the tweets assigned to $\mathcal{C}_l$ than tweet t_j, then $s(t_i) > s(t_j)$. For, each cluster $\mathcal{C}_l \in \mathcal{C}$, we recommend a set of tweets $\mathcal{R}_l$ by maximizing the objective function in eq. (1), where, $\mathcal{C}_l \backslash \mathcal{R}_l$ is the set difference, $sim(t_i, t_j)$ is the similarity between tweets t_i and t_j and τ is the acceptable similarity threshold.

$$\mathcal{R}_l = \{\underset{t_i \in \mathcal{C}_l \backslash \mathcal{R}_l}{\arg\max}\, s(t_i) : \forall t_j \in \mathcal{R}_l \; sim(t_i, t_j) < \tau \text{ and } |\mathcal{R}_l| = K\} \quad (1)$$

For keeping the model simple, we have decoupled the problem. For a set of all relevant tweets, we first divide them into topical clusters and then apply the proposed tweet retrieval algorithm on each cluster independently.

IV PROPOSED METHOD

1 TOPICAL CLUSTERING OF TWEETS

Topic models have been developed to identify hidden semantic structure in a text corpus. Originally developed for standard documents, many earlier researches have leveraged them in the domain of short-text. Hong et al. [12] made a comprehensive study on the use of topic models in Twitter. While many of the earlier works [13], [14] leveraged the popular topic model LDA [15] or its slight variation on individual tweets directly, others [16], [17] tried to aggregate tweets into lengthy pseudo-documents based on some criteria and then applied LDA. More recently, Yan et al. [18] proposed a variation of LDA keeping the length restriction of short-text in mind. While it is possible to incorporate any topic model in the proposed model, for simplicity, we have opted for the vanilla LDA for topical clustering of tweets. We plan to do a comprehensive study in future on how the performance of our system varies due to changes in topic model.

LDA assumes that each tweet in a given collection $\mathcal{T}$ is generated using a multinomial distribution, θ, over k topics. Each topic on the other hand is associated with a multinomial distribution, φ, over the vocabulary. Topic assignment for each word in $t \in \mathcal{T}$ is performed by sampling a particular topic z from multinomial distribution θ_t associated with the tweet. A particular word $w \in t$ is generated by sampling from the multinomial distribution φ_z associated with the topic z. This generative process is repeated n_t times (n_t is the total number of words in tweet t) to produce t. α and β are hyper–parameters for the dirichlet priors of θ and φ respectively. Following [13] we have used symmetric dirichlet priors $\alpha = \beta = 0.01$.

LDA represents each tweet in the collection as a distribution over the k topics. We use this topical distribution to assign the tweets into k topic clusters. Let, $\mathbf{DP}$, a $|\mathcal{T}| \times k$ matrix, hold the topic distribution returned by LDA. The i-th row of $\mathbf{DP}$ holds the topic distribution for tweet $t_i \in \mathcal{T}$. We define a $k \times k$ matrix $\mathbf{A}$ to hold the mean topic distribution (mean probability distribution over topics) of tweets in a cluster. Initially, $\mathbf{A}_{i.} =$ {unit vector in the direction of i-th topic dimension}. Then, as shown in Eq. (2), for each tweet $t_j \in \mathcal{T}$ we use Jensen-Shannon divergence to identify the topic cluster z for which the topical distance between t_j and the average topic distribution of the cluster is minimum. After each tweet assignment, we update the mean topic distribution matrix $\mathbf{A}$.

$$z = \underset{i \in [1,k]}{\arg\min}\, (JSDiv(\mathbf{A}_{i.}, \mathbf{DP}_{j.})) \quad (2)$$

$$JSDiv(\mathbf{A}_{i.}, \mathbf{DP}_{j.}) = \frac{1}{2}(D_{KL}(\mathbf{A}_{i.} \| \mathbf{M}) + D_{KL}(\mathbf{DP}_{j.} \| \mathbf{M})) \quad (3)$$

$$\text{and, } \mathbf{M} = \frac{1}{2}(\mathbf{A}_{i.} + \mathbf{DP}_{j.})$$

D_{KL} in Eq. (3) is the *Kullback-Leibler Divergence* which defines the divergence from distribution q to distribution p as: $D_{KL}(p \| q) = \sum_i p(i) \log \frac{p(i)}{q(i)}$.

Determining Number of Topics in Tweet Collection

LDA takes the number of topics k as an input parameter. However, it is not possible to know in advance the number of discussion topics in a collection of tweets. Following the approach of [19], [20], [21], we used a *cluster validation* method for determining the most appropriate value of k for the tweet collection $\mathcal{T}$.

The objective of the cluster validation method is to identify the most appropriate number of clusters (topics) given a range of possible values $\mathcal{I}$ for a specific tweet collection $\mathcal{T}$. It assumes that for the most appropriate value $k^* \in \mathcal{I}$, the cluster structure estimated from $\mathcal{T}$ would be most stable against re-sampling. Grouping of tweets as a result of a particular clustering is stored in a $|\mathcal{T}| \times |\mathcal{T}|$ connectivity matrix $\mathbf{C}$, which is defined as: $\mathbf{C}_{i,j} = 1$, if $t_i, t_j \in \mathcal{T}$ has been assigned to the same cluster by some clustering algorithm, otherwise, $\mathbf{C}_{i,j} = 0$.

The assumption behind the validation method is that for the ideal value k^*, a clustering algorithm applied to a tweet collection $\mathcal{T}$ and that applied to another tweet collection $\hat{\mathcal{T}} \subset \mathcal{T}$, would result in identical clustering of the tweets in $\hat{\mathcal{T}}$; i.e. if $t_i, t_j \in \hat{\mathcal{T}}$ are put in the same cluster in the former case, then they will be placed together in the later case too. $|\hat{\mathcal{T}}| = \mu |\mathcal{T}|$. Following [19], [20] we have used $\mu = 0.9$ in our experiment. For a particular clustering algorithm applied on $\mathcal{T}$, Eq. (4) measures the proportion of tweet pairs in each cluster [21], which are not put in different clusters when the same clustering algorithm is applied on $\hat{\mathcal{T}}$. We shall refer to this proportion as the *proportion of stability.*

$$F_k(\hat{\mathbf{C}}, \mathbf{C}) = \frac{\sum_{i,j} 1\{\hat{\mathbf{C}}_{ij} = \mathbf{C}_{ij} = 1, t_i, t_j \in \hat{\mathcal{T}}\}}{\sum_{ij} 1\{\mathbf{C}_{ij} = 1, t_i, t_j \in \hat{\mathcal{T}}\}} \tag{4}$$

However, Eq. (4) is biased towards smaller values of k. If $k = 1$, i.e. there is only one cluster, then $F(\hat{\mathbf{C}}, \mathbf{C})$ will be equal to 1, the highest possible value for F_k. Intuitively, the fraction will keep getting smaller as k keeps increasing. Hence, we use Eq. (5), proposed by [20], which normalizes the value of $F(\hat{\mathbf{C}}, \mathbf{C})$, diminishing the effect of k.

$$\acute{F}_k = F_k(\hat{\mathbf{C}}, \mathbf{C}) - F_k(\hat{\mathbf{G}}, \mathbf{G}) \tag{5}$$

Table 1 explains the connectivity matrices used in Eq. (5). “RAND” is a clustering algorithm based on uniformly drawn random assignments of tweets among the clusters. The second part of the right hand side of Eq. (5) measures the *proportion of stability* achieved by random assignment of tweets to different clusters. This function is also biased towards smaller values of k. Hence, by subtracting this biased factor from the *proportion of stability* achieved by LDA, the effect of the number of clusters k is diminished.

Connectivity Matrix	**Clustering Alg.**	**Tweet Corpus**
$\mathbf{C}$	LDA	$\mathcal{T}$
$\hat{\mathbf{C}}$	LDA	$\hat{\mathcal{T}} \subset \mathcal{T}$
$\mathbf{G}$	RAND	$\mathcal{T}$
$\hat{\mathbf{G}}$	RAND	$\hat{\mathcal{T}} \subset \mathcal{T}$

Table 1: Definition of Connectivity Matrices used in Eq. (5)

To reduce the effects of chance, the process in Eq. (5) is repeated ρ times, each time taking a different subset of tweets $\hat{\mathcal{T}}^{(n)}$. Following [19] we have used $\rho = 6$ in our experiment. The average of the values, as shown in Eq. (6), represents the measure of stability achieved by a particular choice of k.

$$\tilde{F}_k = \frac{1}{\rho} \sum_{n=1}^{\rho} \Big(F_k(\hat{\mathbf{C}}^{(n)}, \mathbf{C}) - F_k(\hat{\mathbf{G}}^{(n)}, \mathbf{G}) \Big) \tag{6}$$

Hence, the most appropriate number of topics k^* for the tweet collection $\mathcal{T}$ can be obtained by solving Eq. (7)

$$k^* = \arg\max_{k \in \mathcal{I}} \tilde{F}_k \tag{7}$$

2 IDENTIFYING RELEVANT TWEETS FROM TOPIC CLUSTERS

After dividing tweets into topic clusters, we focus on identifying the tweets that comprise key discussion points within the clusters. For tweets in each topic cluster, we construct a lexical graph and then apply variant a of the PageRank algorithm [22] to determine the score for individual lexical units in the graph. Tweets comprising higher proportion of high-scored lexical units are recommended to the users. The following subsections describe the procedure.

2.1 CONSTRUCTING THE LEXICAL GRAPH

As we are trying to identify the key discussion points in the tweet collection, using unigram as lexical unit seems to be a reasonable choice. In our lexical graph, an edge between two nodes represents the strength of association between the unigrams in the tweet collection. Following we describe our nodes, edges and edge-weight selection strategies in detail.

As we are trying to identify key discussion points by using the strength of association between words in the tweets, anything other than words in the tweets are not useful information for us. So, we remove all URLs, user references (@user), numerals, time expression and non printable characters from the tweet collection. Duplicate tweets and tweets with less than 10 terms are excluded from the corpus. Retweets are the main sources of duplicate tweets. Tweets such as *"Don't wanna hear it . . . no! no! no! no! why why why why . . . "* are common in any twitter feed, which contain enough terms, but not any useful information. To identify such repetition of words in a tweet t, we used Shannon's entropy. If $\mathbf{w}= \{w_1, w_2 \ldots w_n\}$ are the unigrams in t with frequency $\mathbf{f}= \{f_1, f_2 \ldots f_n\}$, the entropy of the tweet is obtained as: $H(t) = -\sum_{i=1}^{n} \frac{f_i}{n} \log_2(\frac{f_i}{n})$ All tweets with $H(t) \leq \xi$ are excluded from the corpus.

Let $\mathcal{U}$ be the set of all unigrams encountered in the collection of tweets. Earlier research [23] that used graph–based ranking algorithms reported that better results can be obtained by restricting incorporation of vertices to the graph using syntactic filters, which select only lexical units of a certain part–of–speech. [24] also reported that integration of part–of–speech information into their learning process could build a better classifier for extracting keywords from document abstracts. We adopt a similar approach and only unigrams in the set $\mathcal{U}^* = \{w : w \in \mathcal{U} \text{ and } \mathcal{POS}(w) \in \{\textit{verb, noun, adjective}\}\}$ participate in the subsequent co-occurrence identification process, where $\mathcal{POS}(w)$ returns the part–of–speech of a unigram, which we determine using the "Stanford Log–linear Part–Of–Speech Tagger" [25].

As Twitter users often do not follow any standard grammar for their posts, we look for co-occurrences where the two unigrams stand in more flexible relationship to one another. Hence, instead of looking for pair of unigrams immediately following each other, we use a collocation window of 3 – thus considering each unigram pair in the window as a potential co-occurrence. For example, the phrase "President Barack Obama" would produce three co-occurrences; "President Barack", "President Obama" and "Barack Obama" when the collocation window is set to 2 or higher. As reported in [26], this method is quite successful at terminology extraction and determining appropriate phrases for natural language generation. Let, $\hat{\mathcal{B}} = \{(w_1, w_2) : w_1, w_2 \in \mathcal{U}^* \text{ and } dist_t(w_1, w_2) \leq 3 \text{ for some } t \in \mathcal{T}\}$. The function $dist_t(u, v)$ returns the distance between unigrams u and v in tweet t.

	$f(w_1)$	$f(w_2)$	$f(w_1, w_2)$	$(-2\log\lambda)$
blue tie	30	5	3	29.5
blue flag	30	20	3	19.4

Table 2: Co-occurrence frequency is not the best way to measure strength of association

To determine whether an identified co-occurrence is statistically significant, we have adopted the "Likelihood Ratio" measure for hypothesis testing of independence proposed in [27], which takes into account the volume of data that has been considered for calculating the frequency of the identified co-occurrences as well as the frequency of the individual words comprising the co-occurrences. For sparse data (as in case of Twitter) this approach is more appropriate than the χ^2 test [28].

For each identified co-occurrence we calculate its likelihood ratio. Likelihood ratio a ratio of two hypotheses that tells how much more likely one hypothesis is over another. The hypothesis of independence $\mathcal{H}_1$ states that there is no association between the words in the co-occurrence beyond chance occurrences. The second hypothesis $\mathcal{H}_2$ states that the association between the words in the co-occurrence are statistically significant. The likelihood ratio of the two hypotheses is $\lambda = \frac{L(\mathcal{H}_1)}{L(\mathcal{H}_2)}$. $(-2\log\lambda)$ is asymptotically a χ^2 distribution. Hence, we reject the hypothesis of independence, $\mathcal{H}_1$, for an identified co-occurrence with 95% confidence if $-2\log\lambda \geq 7.88$, which is the critical value for χ^2 distribution with 1-degree of freedom at confidence level $\alpha = 0.005$. Let, $\mathbf{L}$ be a $1 \times |\hat{\mathcal{B}}|$ vector holding the values of $(-2\log\lambda)$ for the co-occurrences in $\hat{\mathcal{B}}$. Therefore, our identified co-occurrences with statistical significance from the tweet collocation are $\mathcal{B}^* = \{b : b \in \hat{\mathcal{B}} \text{ and } \mathbf{L}_b \geq 7.88\}$. Let, $\mathcal{U}^b = \{w : ((w, w') \in \mathcal{B}^* \text{ or } (w', w) \in \mathcal{B}^*) \text{ and } w, w' \in \mathcal{U}^*\}$.

Frequency of a co-occurrence in the tweet collection is a natural candidate for measuring its strength of association. Several earlier works [7], [8] used this metric. However, this metric does not take into account frequencies of constituent unigrams. Let us consider that we have identified two co-occurrences "*blue tie*" and "*blue flag*" both occurring three times in our tweet collection. Let, the frequencies of unigrams are $blue = 30$, $tie = 5$ and $flag = 20$. From these counts, it is easily understandable that the unigram "*blue*" has a stronger association with the unigram "*tie*" than the unigram "*flag*" in our collection of tweets. This is because 60% of all appearances of the unigram "*tie*" is collocated with the unigram "*blue*", whereas for the unigram "*flag*" the percentage is only 15%. If strength of association is measured using only the frequency of co-occurrence, this information will be lost. Whereas, as we have mentioned earlier, the frequencies of individual unigrams are taken into consideration in calculating λ. Table 2 shows the values of $(-2\log\lambda)$ calculated for the bigrams "*blue tie*" and "*blue flag*" (considering a total of $12,000$ terms in the corpus). From the table, it is evident that $(-2\log\lambda)$ can capture the strength of association better than the frequency-based approach, because it assigns higher weight to co-occurrence "*blue tie*" over "*blue flag*", whereas, the frequency-based approach assigns the same weight to both co-occurrences. Hence, we use $(-2\log\lambda)$ as the edge weight between two unigrams in the lexical graph. Thus, our lexical graph for the tweets of one topic cluster is an undirected weighted graph $G = (\mathcal{U}^b, \mathcal{B}^*, \mathbf{W})$, where each unigram in $\mathcal{U}^b$ is a node in the graph and each identified co-occurrence in $\mathcal{B}^*$ defines an edge connecting two nodes. The weight matrix $\mathbf{W}$ is defined as follow:

$$\mathbf{W}_{ij} = \begin{cases} \mathbf{L}_b & \text{if } b = (w_i, w_j) \in \mathcal{B}^*, \\ 0 & \text{otherwise.} \end{cases}$$

2.2 IDENTIFYING TWEETS RELEVANT TO THE TOPIC

We have used a graph-based ranking algorithm on our constructed lexical graph to identify key points discussed in the tweet collection. Graph-based ranking algorithms have been successfully used for analyzing link structure over Internet [22], [29], ranking key-phrases in document abstracts [23], analyzing social network structure [16] and in numerous other domains. These algorithms essentially exploit the link structure of the entire network to determine the importance of individual node in the graph. They iterate until a certain threshold condition is met and at each iteration propagate the structural information further deep into the graph. Upon convergence, a score is affixed to each node, representing its importance.

Probably the most famous graph-based ranking algorithm, PageRank [22], is based on intuitive notion of endorsement. In PageRank, a page can have high ranking if many pages point to it or some other high–ranking pages point to it. Hence, if the citation relationship is represented by a graph $G = (V, E)$, where each node in V is a webpage and each edge in $E = (v_i, v_j)$ is a citation of page v_j in page v_i, then PageRank is defined as follows:

$$\mathbf{PR}(v_j) = (1-d) + d \sum_{(v_i,v_j)\in E} \frac{\mathbf{PR}(v_i)}{|\{v_k : (v_i, v_k) \in E\}|} \tag{8}$$

Eq. (8) tries to model a *random surfer's* browsing behavior. Parameter $d \in (0,1)$, called the *damping factor*, is the probability at each page that the random surfer will get bored and request another random page. The denominator of the fraction in eq. (8) is a normalizing factor ensuring that links from all pages are not counted equally. Eq. (8) is a recursive equation which iterates until convergence and can start with any set of initial ranks [22].

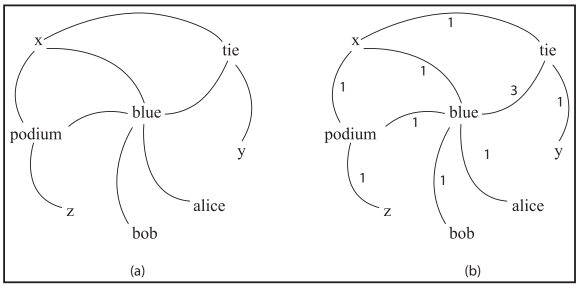


Figure 2: Illustrative graph showing word association in a tweet collection

There are two major differences in the web domain and our application domain. In the domain of web surfing, citation directions are essentially directional. Hence, PageRank algorithm is traditionally applied on directed graphs. However, for co-occurrences, the order of words is not that important. For example, in the phrases "*knocked the door*" and "*door was knocked*", it is important to observe that the word "*door*" has high association with the word "*knocked*" – which word is coming first has little importance. Hence, we opted for an undirected lexical graph.

The second difference is, when a web page refers to another page, assignment of weight to such citation is not well defined. Hence, original PageRank uses un-weighted graph. However, the strength of association between co-occurrences in a tweet collection can be quantified. Incorporation of the knowledge of association strength between two unigrams allows one unigram to endorse the other in proportion to their level of association. This helps to alleviate the rank of the co-occurrence as a group. We illustrate this more clearly using figure 2. It is evident from the graphs that the word "*blue*" is highly popular in the tweet collection represented by the graphs. Let us consider the words "*podium*" and "*tie*". Figure 2(a) reveals that both the collocations "*blue tie*" and "*blue podium*" have appeared in the tweet collection. However, suppose the collocation "*blue tie*" appears several times more than "*blue podium*", indicating that "*blue tie*" has caught more public attention than "*blue podium*". Hence, if everything else remains same, the tweets mentioning "*blue tie*" should be ranked higher than those talking about "*blue podium*", because they would better represent popular discussion points. This would be possible, if the word "*tie*" is ranked higher than the word "*podium*". However, traditional PageRank algorithm that uses eq. (8) would assign same rank to words "*podium*" and "*tie*". Hence, in the proposed method, we use a weighted version of eq. (8) as expressed in eq. (9).

$$\mathbf{PR}'(v_j) = (1-d) + d \sum_{(v_i,v_j)\in E} \frac{\mathbf{W}_{ij} * \mathbf{PR}'(v_i)}{\sum_{(v_i,v_k)\in E} \mathbf{W}_{ik}} \quad (9)$$

Figure 2(b) is a weighted version of figure 2(a) capturing the fact that the collocation strength of "*blue tie*" is three times higher than that of "*blue podium*". The results of applying eq. (8) on the graph in figure 2(a) and that of applying eq. (9) on the graph of figure 2(b) is shown in table 3. We have used $d = 0.85$ following [22] and used a convergence threshold of $\zeta = 0.0001$ following [23]; i.e. the algorithm converges if in two successive iterations, the rank of any of the nodes does not change by more than ζ. As shown in table 3, the word "*tie*" is assigned higher rank than the word "*podium*" by the proposed solution, which helps us reach our target objective.

We have applied the weighted, undirected version of PageRank algorithm on the constructed lexical graph for a topic cluster. Initially, $\mathbf{PR}'(w)$ is set to 1 for all $w \in \mathcal{U}^b$. Initial scores of the nodes can be set to any unique value [22]. Upon completion, each unigram $w \in \mathcal{U}^b$ receives its score in $\mathbf{PR}'(w)$. Let ρ is a vector in $\mathbb{R}^l$ such that $l = |\mathcal{U}^b|$ and $\rho = \{\mathbf{PR}(w) : w \in \mathcal{U}^b\}$. Let, $y : t \to \{0,1\}^l$ be a function representing the set of words $w \in \mathcal{U}^b$ present in tweet t. Then the score associated with any tweet $t \in \mathcal{T}$ is

$$s(t) = y(t) \cdot \rho \quad (10)$$

Prior works [8] also adopted this *sum-up-the-token-weights* approach and reported better results while summarizing a tweet corpus. The score of a tweet is a relative measure indicating how many of the popular co-occurrences or terms the tweet contains in comparison to the other tweets in the collection. After removing the near duplicate tweets from the collection, the tweets are sorted in descending order of their scores. Top-K tweets from each topic cluster are recommended to the users. Of course, a user interested in any particular topic can further dig deeper into the sorted list of tweets in that cluster. The collection of recommended tweets from all clusters forms the "**recommended set**" for the event.

Many of the tweets convey the same information in slightly different forms which often frustrates search users looking for new content [2]. We removed duplicate tweets during pre-processing step. Tao et al. [2] did a comprehensive study on the effectiveness of various similarity measurement methods for identifying tweets demonstrating different levels of similarity. For simplicity, we use a variation of Jaccard distance, sometimes referred to as Simpson or Overlap distance [30], to remove near-duplicate tweets from the set of tweets recommended to the users. Let, the function $S(t)$ returns the set of words in tweet t. Then, Simpson distance between two tweets is defined as $simp(t_1, t_2) = 1 - \frac{|S(t_1) \cap S(t_2)|}{min(|S(t_1)|, |S(t_2)|)}$. Let us consider two tweets t_1: "*President Obama on Romney's tax plan: I think math, common sense, and our history shows us that's not a recipe for job*" and t_2: "*I think math, common sense, and our history shows us that's not a recipe for job*". For these two tweet, Jaccard distance $Jaccard(t_1, t_2) = 0.27$ whereas, $simp(t_1, t_2) = 0$. The set of tweets, $\mathcal{R}_l$, recommended for each cluster $\mathcal{C}_l \in \mathcal{C}$ is selected by maximizing the objective function in eq. (11).

	blue	**tie**	**podium**	**x**
unweighted	2.1	1.3	1.3	1.24
weighted	2.36	1.67	1.13	1.04

Table 3: Difference in word ranking between weighted and un-weighted versions of the PageRank algorithm

$$\mathcal{R}_l = \{\underset{t_i \in \mathcal{C}_l \setminus \mathcal{R}_l}{\arg\max}\, s(t_i) : \forall t_j \in \mathcal{R}_l \ sim(t_i, t_j) < \tau \\ \text{and } |\mathcal{R}_l| = K\} \quad (11)$$

V PERFORMANCE EVALUATION

The objective of the proposed model is to recommend a set of tweets to the search users, who put event related keywords and/or hashtags as a search query, so that, the recommended set can delineate the real-time event as accurately as possible. The evaluation has been performed on a real-world event using actual tweets generated while the event was taking place. We used the standard measures of precision and recall to evaluate the performance of the proposed model against two competing models. For the sake of evaluation, we divided the real-world event into several segments and evaluated the performance of the competing models for each segment. The logic behind the segmentation was that user may place the query at any time during the event and might want to know highlights of the discussion points either for the entire event so far or for any specific time interval between the event's commencement and the present time. However, performing evaluation for a continuous time domain is impractical. So, we segment the event at equal time intervals and apply the retrieval algorithm on each

#debate, #debates, #Denverdebate, #election2012, "#obama", "#romney", #barakobama, #mittromney, #obama2012, #romneyryan2012, #presidentialdebate

Table 4: Track keywords used for Twitter Search API

interval independently. It is to be noted that, though a tweet might be posted within the time interval of one of our defined segments, it is not necessary that it would refer to some occurrences that took place only within the boundary of that segment. The tweet might refer to any previous occurrences or points of the event since it commenced. We keep this in mind while calculating precision and recall for a segment, so that the evaluation is not biased by the specific choice of segment boundaries or length of individual segments. The retrieval process of the recommended set by the proposed model has been laid out in the previous section. In this section we present the baseline models, the experimental setup and the evaluation results in detail.

1 BASELINE MODELS

1.1 TF-IDF-COSINE MODEL

This model has been used in earlier research [6] for determining the relevance of tweets to be recommended to the users. The objective of this model is to select those tweets which are closest to all other tweets in the tweet collection. Hence, the objective function is quite similar to the objective function we laid out in the introduction part of this paper. Hence, this model is a good candidate as a baseline model for evaluating the performance of the proposed model. In this model, each tweet is represented as a length $|\mathcal{V}|$ vector of *tf-idf* of its constituent words, where $\mathcal{V}$ is the set of vocabulary. Let, $tf_{w,t}$ be the normalized term frequency of term w in tweet t. The inverse document frequency of a term in the tweet collection $\mathcal{T}$ is represented as $idf_{w,\mathcal{T}} = \log \frac{\mathcal{T}}{|t \in \mathcal{T} : w \in t|}$. tf-idf$_{w,t}$ $= tf_{w,t} * idf_{w,\mathcal{T}}$. Cosine similarity between two vectors u and v is defined as $cosine(u, v) = \frac{u.v}{\|u\|\|v\|}$. The ranking score of a tweet t is determined as: $score(t) = \sum_{t' \in \mathcal{T}} cosine(t, t')$. For the sake of fairness among competing models, we removed duplicate and near duplicate tweets from the model's *recommended set* using *Simpson distance* method described earlier.

1.2 RESONANCE MODEL

Twitter uses a specialized ranking function which among other indicators also considers the resonance signal to compute a relevance score for each tweet [1]. Resonance signal includes the users' interactions with a tweet, e.g. number of times the tweet has been replied or retweeted. Hence, a relevant tweet, which is retweeted many times, enjoys higher score. The "Resonance Model" uses the *retweet–count* (number of times a tweet has been retweeted) of a tweet to emulate the resonance signal. In this model, the retweet–count of a tweet is considered as its relevance score. To avoid duplicate tweets from appearing in the tweets recommended from each cluster, only the tweet with the highest retweet count among a set of *peer retweets* (set of retweets whose source tweet is the same) is considered. Simpson distance is used to get rid of near duplicate tweets.

2 EXPERIMENTAL SETUP

2.1 PARAMETER SETTING

Our model has several parameters as described in the "Proposed Method" section. Some of these parameter values were chosen based on the results reported by earlier research works and we mentioned them where the parameters have been introduced. For the parameters ξ used in identifying tweets with repeating terms and τ in eq. (11), we determined their values using a development dataset while checking over a range of reasonable values ($[0.5, 5]$ for ξ and $[0, 1]$ for τ). The development dataset was different from that used for the performance evaluation of the proposed method. Based on the experimental results we finally set the values $\xi = 2.5$ and $\tau = 0.6$.

2.2 THE REAL-TIME EVENT

We chose the first US presidential debate held on October 3, 2012 as our target real-time event. This event satisfies the criteria that we have laid out in the introduction part. It elicited a huge response from Twitter users and it was held within a period of 90 minutes. Date and time of the event was announced months before the event. We used Twitter's Streaming API with "track" request parameter and used the track keywords listed in table 4 to collect a sample of tweets generated during 21:00 to 22:30 Eastern Time, while the debate was taking place. For this experiment we considered only English tweets. A total of 212,308 different users posted 270,337 tweets in English. We denote this set of tweet collection as $\mathcal{T}$. We realize that our collected tweets are a subset of all relevant tweets and a better designed crawler with more sophisticated hardware platform might have collected more relevant tweets. However, designing such crawler is out of scope of this research work and we believe our tweet collection is a representative sample of all relevant tweet.

2.3 DEBATE SEGMENTS

As mentioned in the beginning of this section, we divided the event into fifteen 6–minute segments. This is a design choice which can be set to any value and as all competing models worked on the same intervals, it does not impact the performance comparison. The segments are represented as tuples $\mathcal{S} = \{(\mathcal{S}_i, \mathcal{B}_i, \mathcal{E}_i, \mathcal{N}_i) : i \in [1, n_s]\}$, where $\mathcal{S}_i$ is the set of important discussion points ("focal points") identified during *ground truth construction* that can delineate important happenings during segment i. $\mathcal{B}_i$ and $\mathcal{E}_i$ are respectively the beginning and ending time of segment i and $\mathcal{N}_i = |\mathcal{S}_i|$ is the number of *focal points* in segment i. In this experiment we used $n_s = 15$ (number of debate segments). Our tweet collection was also segmented accordingly. Hence, $\mathcal{T} = \bigcup_{i=1}^{n_s}\{\mathcal{T}_i : \forall t \in \mathcal{T}_i \text{ and } t \text{ is generated between } \mathcal{B}_i \text{ and } \mathcal{E}_i\}$.

2.4 GROUND TRUTH CONSTRUCTION

For each debate segment, we tried to identify the key points discussed within the segment and the key rhetoric made by the candidates. The selections were made keeping in mind that these points should be enough to give someone, who could not follow the live telecast, a brief overview about what happened during that segment. No efforts were made to select uniform number of points per discussion topic or per unit of time. As a result, in some cases, consecutive sentences from one speaker made two focal points, while in some other cases, there was no focal point from consecutive paragraphs, as the annotators could not find any point that they thought might help to characterize the segment. Four annotators worked independently by going through the video of the debate, reading the transcript and watching the highlights presented in mainstream news media such as BBC, CNN and Fox News. We call each identified point a "***focal point***" of the segment. Annotators were not allowed to go through the relevant tweets while identifying focal points. The final set of selected focal points contained only those points on which majority of the annotators could reach in accord. This set contained an average of 18 focal points per segment. Let, $p_{i,j}$ denote the j-th focal point picked from segment i. Hence, $\mathcal{S}_i = \{p_{i,j} : j \in [1, \mathcal{N}_i]\}$. For paucity of space it is not possible to list all identified focal points. Hence, we present only a few samples from some selected segments in table 5.

Segment	**Focal Point**
1	*I'm sure this was the most romantic place you could imagine, here with me*
5	*The president said he'd cut the deficit in half. Unfortunately, he doubled it.*
10	*Obamacare will make middle class families secure.*
15	*We ended war in Iraq and going to wind down war in Afganistan.*

Table 5: Sample *Focal Points* for different debate segments

2.5 ANALYZING TWEETS IN RECOMMENDED SET

As all the tweets in our experiment satisfied the Boolean query comprising the track keywords, they are in some way relevant to the event. However, the objective of the proposed model is to increase the fraction of tweets in the recommended set that can outline the proceeding of the event. Hence, four annotators tried to manually identify tweets in the recommended sets that referred to some identified *focal points* in the debate. We call these tweets "citation tweets". The same annotators, who participated in ground truth construction, also participated in analyzing tweets in the recommended sets. They also observed another category of tweets, which did not refer to any focal point, but made general comments on the proceedings of the event, such as the facial expressions of the candidates, performance of the moderator etc. These tweets could also convey valuable information regarding the event's proceeding to the search users and we refer to them as "narration tweets". The third category of tweets made no direct reference to the debate, but discussed on the upcoming election, personal opinion about candidates such as their past and prospect etc., which convey no event-relevant information to the user. We call them the "distant tweets". It should be mentioned here that the purpose of the proposed model is not to classify the tweets into these categories, but to increase the proportion of citation and narration tweets in the recommended set and reduce distant tweets. These classifications are presented here only to clarify the evaluation procedure. Some example tweets from each category is presented below:

- *Citation Tweets:* Tweets directly quoting or making comments on a focal point. For example, "*Obama: The question here tonight is not where we've been– but where we're going. #debates*

Idiot! That is the question 4 the guy n office". Let,

$$fp(t) = \{\text{set of focal points cited in tweet } t\}$$

$$ref(t,i) = \begin{cases} 1 & \text{if } (fp(t) \cap \bigcup_{j=1}^{i} \mathcal{S}_j) \neq \emptyset \\ 0 & \text{otherwise.} \end{cases}$$

- *Narration Tweets:* Tweets commenting on the proceedings of the debate. For example, "*Anyone notice how Obama's blue tie and Romney's red and white tie makeup the American flag? #PresidentialDebate* " or "*Donno who is winning, but Jim Lehrer is certainly loosing ... got no control over anyone ... #debates*". Let,

$$narr(t) = \begin{cases} 1 & \text{if } t \text{ is a } \textit{narration} \text{ tweet} \\ 0 & \text{otherwise.} \end{cases}$$

- *Distant:* Personal opinion about the debate, candidates, upcoming election, opposition supporters etc. For example, "*Only 34 days until Election Day. RT if you're excited to cheer on President Obama in tonights #debate.*"

Some tweets also referred to more than one focal points even from different segments. There were also tweets that were a mixture of *Citation* and *Narration*. For example, "*Romney - president imposed a middle class economy tax. obama's only dodging so far. romney already used biden's middle class being buried. #Denverdebate*".

2.6 EVALUATION MEASURES

We have used the standard measures of precision and recall for comparing the performance of the proposed model against the baseline models. Recall for a segment i gives the fraction of *focal points* in S_i that are referred by some tweets in the recommended sets for any segment from segment i to the last segment, n_s.

$$Recall(\mathcal{S}_i) = \frac{\sum_{j=1}^{\mathcal{N}_i} covered(p_{i,j})}{\mathcal{N}_i}$$

$$covered(p_{i,j}) = \begin{cases} 1 & \text{if } p_{i,j} \in fp(t) \wedge t \in \bigcup_{l=i}^{n_s} \mathcal{RS}_l \\ 0 & \text{otherwise.} \end{cases}$$

If a focal point $p_{i,j}$ is referred by some tweet in the recommended sets for any segment from i until n_s, it is considered to be *covered*.

For a segment i, precision P gives the fraction of tweets in $\mathcal{RS}_i$ that refer to some focal points of any segment from the first segment to segment i or that can be categorized as *narration* tweets.

$$P(\mathcal{S}_i) = \frac{|\{t : t \in \mathcal{RS}_i \text{ and } (ref(t,i) \vee narr(t)) = 1\}|}{|\mathcal{RS}_i|}$$

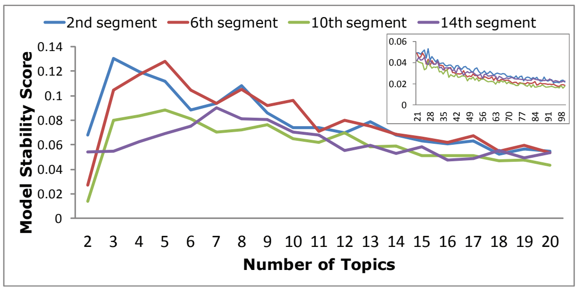


Figure 3: Model stability score for different number of topics

3 RESULTS

We applied cluster validation method on each tweet collection $\mathcal{T}_i$ corresponding to each segment i for obtaining the optimum number of topic cluster for the segment. The topic range $\mathcal{I}$ was set to 2–100. The model stability scores for some selected clusters are shown figure 3. It can be observed that as the number of topics increases after 10, the model gets more and more unstable, causing a gradual descent in stability score. All the segments showed similar trend. Hence, for the sake of clarity, we show here only some selected segments. Scores for topics 2–20 has been shown in the main graph and that for topics 21–100 has been shown in the mini-graph. Highest number of topic clusters identified for any of the debate segments was 7.

All competing models were applied on each identified topic cluster within each segment to retrieve a set of tweets from that cluster. For evaluation we considered top-K tweets returned by each model. So, if k clusters were identified in a particular debate segment $\mathcal{T}_i$, then we agglomerated the $k * K$ tweets returned by each model for performance comparison. We call this set the *Recommended Set* for segment $\mathcal{T}_i$ and denote it by $\mathcal{RS}_i$. We used the value $K = 5$ and $K = 10$ in this evaluation. Figure 4 and figure 7 shows the achieved precision by the three models for $K = 5$ and $K = 10$ respectively. The recall achieved by the three models are shown in figure 5 and figure 8 . The proposed model performs better than both the baseline models in terms of both precision and recall for majority of debate segments. We shall discuss the results in detail in the discussion section.

To evaluate the impact of topical clustering, we performed an experiment where the proposed tweet retrieval algorithm was applied on each tweet segment without performing topical clustering. Figure 6 shows the results of the experiment. To evaluate the impact of duplicate and near duplicate

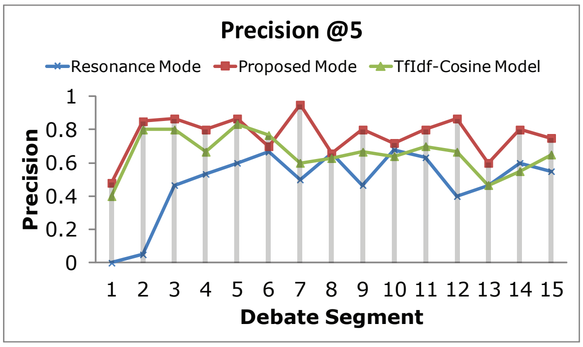


Figure 4: Performance comparison between the proposed model and the baseline models at $K = 5$ in terms of precision

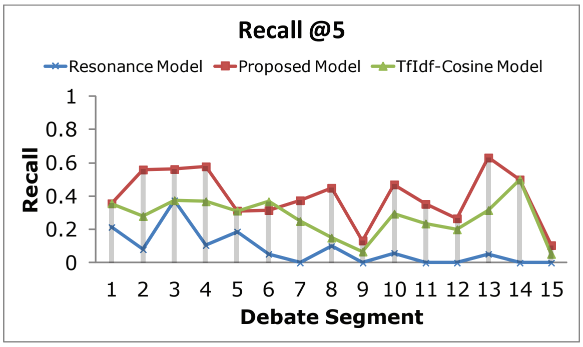


Figure 5: Performance comparison between the proposed model and the baseline models at $K = 5$ in terms of recall

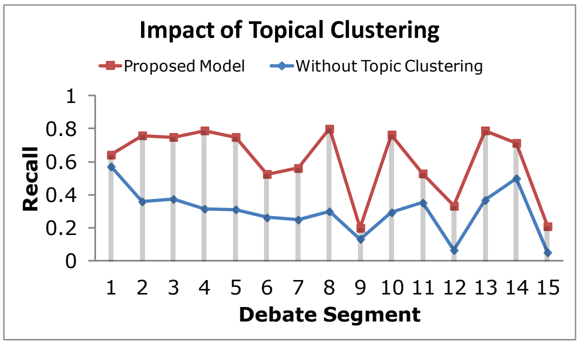


Figure 6: Impact of topical clustering on recall at $K = 10$

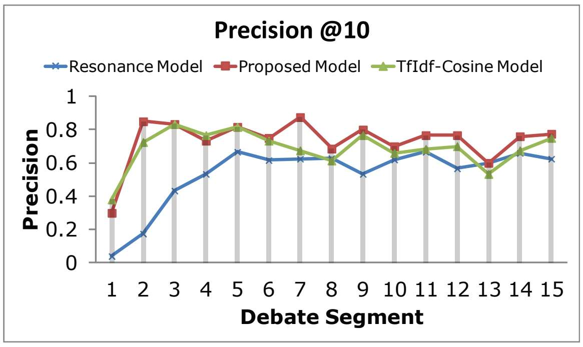


Figure 7: Performance comparison between the proposed model and the baseline models at $K = 10$ in terms of precision

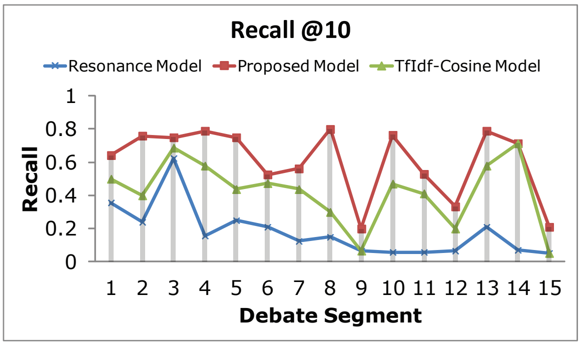


Figure 8: Performance comparison between the proposed model and the baseline models at $K = 10$ in terms of recall

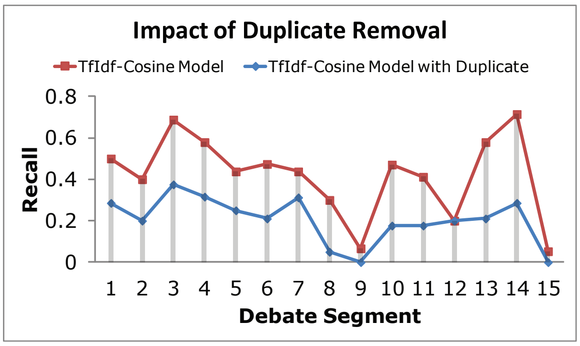


Figure 9: Impact of duplicate and near–duplicate removal from recommended set on recall at $K = 10$

removal we applied the *tf-idf-cosine* model on each tweet segment $\mathcal{T}_i$ and generated the recommended set without removing duplicate or near duplicate tweets. The results are shown in figure 9.

Though the *tf-idf-cosine* model closely contends with the proposed model in terms of precision, the proposed model clearly outperforms it in terms of recall (fig. 5, fig. 8). This substantiates that our method for delineating an ongoing event is better than that of the *tf-idf-cosine* model.

VI DISCUSSION

The precision of the proposed model outperforms that of both the baseline models for most of the

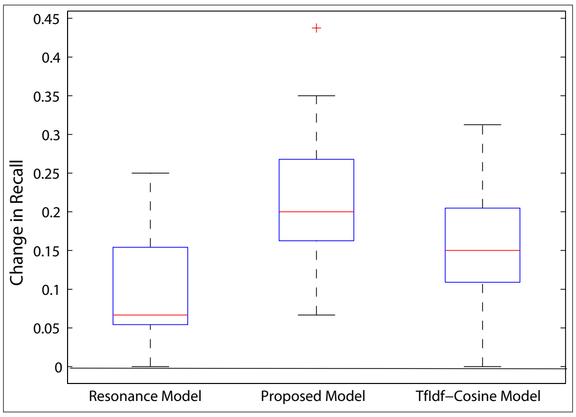


Figure 10: Impact on Recall in different debate segments for an increase in K from 5 to 10

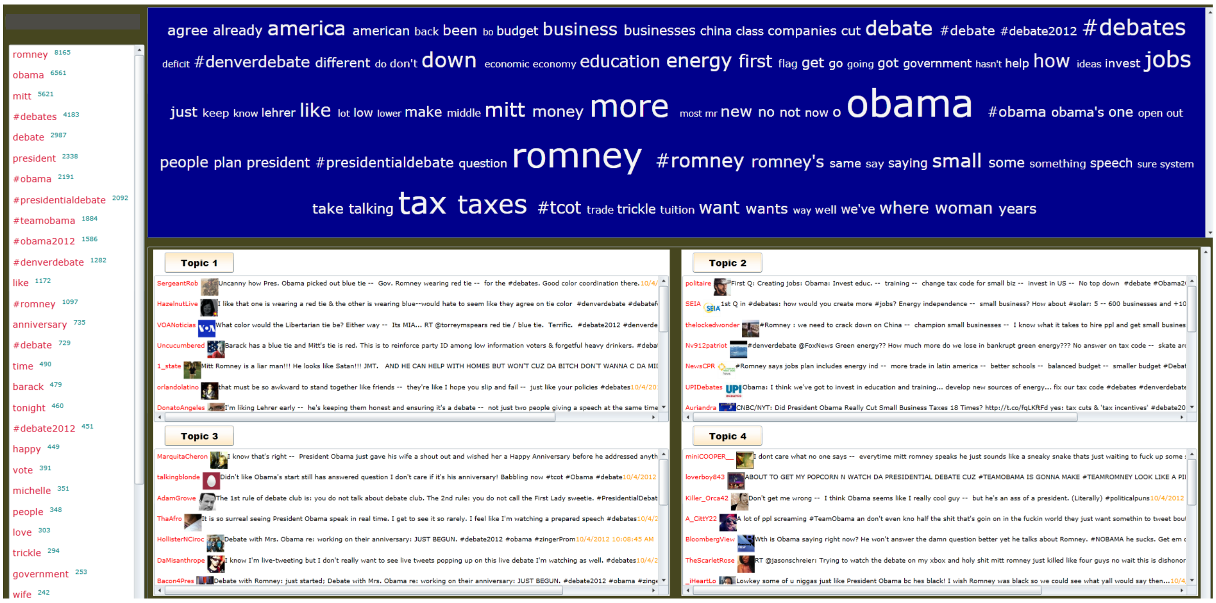
debate segments (fig. 4, fig. 7). For the first six segments, precision of the Resonance model is significantly lower than that of the other two models. This is because, *"citation"* and *"narration"* tweets take some time to compete with the *"distant"* tweets in terms of retweet count. A *"distant"* tweet might be generated long before the actual event commences, however *"citation"* and *"narration"* tweets can only be generated while the event is taking place. Both figures 7 and 8 affirm that the most re-tweeted tweets not necessarily report the most important discussion points in an ongoing events.

Objective of search engines is to maximize the precision of retrieval in the top K returned pages even at the cost of low recall [22]. However, our objective function is significantly different from theirs. One of our core objectives is to delineate as many important moments of an ongoing event as possible. Hence, increasing recall is more important for us even at the cost of reasonable loss in precision. Figure 4 and figure 7 reveal that increase in K from 5 to 10 has little effect on precision, at least, for the proposed model and the *tf-idf-cosine* model. However, as we can see from figure 5 and figure 8, recall improves significantly for all three models when K is increased. However, increasing K would put a burden on the end user as he/she has to read more tweets to get an overview about the event. For example, if there are seven topical clusters in the search result and we use $K = 10$, the user has to read 70 tweets. Hence, it is a tradeoff between the effort a user devotes vs. the detail of the event he perceives. We further investigated the amount by which recall for the competing models improved for different debate segments due to increase in K. Figure 10 shows the amount of improvement in recall due to changes in K from 5 to 10 for different debate segments using box plot. As the figure reveals, the improvement in recall in different debate segments is more prominent (higher median) for the proposed model. The proposed model shows positive improvement for all debate segments (minimum is 0.067), which is not the case for the other two models (minimum is 0).

The recall for debate segment 9 is substantially lower than the other segments for all three models. We observed that the discussion points in this segment were related to "Government Regulations". Whereas, that of segments 8 and 10 were about "Medicare and Social Security" and "*Affordable Care Act*" respectively. It is apparent from the analysis that people are more concerned about healthcare and social security related issues compared to government regulation related issue. So, discussion topics in most relevant tweets generated within these three segments were dominated by healthcare and social security issues. This result adverts that the proposed method can also give insights about the influence that different segments in a public event exert on general population.

The impact of topical clustering in recall is evident from figure 6. As expected, it contributes to the amelioration of recall. Vanilla LDA, which was not designed for short-text, does not offer best topical clustering for tweets [18]. However, the experiment clearly reveals that topical clustering improves the quality of recommended set. Hence, by replacing vanilla LDA with a better topic model it would be possible to further improve the performance. Figure 9 corroborates to the intuitive expectation that removal of duplicates and near–duplicates from recommended sets improves the recall substantially.

We have developed a user interface for our proposed method. Figure 11 is a screenshot of the developed UI. It has three parts: a) the left panel b) the top panel (in blue) and c) the bottom panel. The bottom panel shows the tweets in different topic clusters. Number of topics varies dynamically and is determined using the method described in subsection 1 of section IV. Users can scroll down to see the tweets from all the topics. Tweets from different topical clusters are presented in different sub-panels. Tweets in a particular sub-panel are presented in descending order of their relevance score determined by eq. 10. If the user finds tweets in a particular topic interesting, he/she may go through all the tweets in the sub-panel corresponding to that topic. Each sub-panel allows users to scroll down the list of all tweets in that particular topic. Once a user clicks on the button labeled with the topic number (e.g. Topic 1), the word cloud in the top panel (in blue) shows the most frequent words in the



romney	8165
obama	6561
mitt	5621
#debates	4183
debate	2987
president	2338
#obama	2191
#presidentialdebate	2092
#teamobama	1884
#obama2012	1586
#denverdebate	1282
like	1172
#romney	1097
anniversary	735
#debate	729
time	490
barack	479
tonight	460
#debate2012	451
happy	449
vote	391
michelle	351
people	348
love	303
trickle	294
government	253
wife	242

agree already america american back been bo budget business businesses china class companies cut debate #debate #debate2012 #debates
deficit #denverdebate different do don't down economic economy education energy first flag get go going got government hasn't help how ideas invest jobs
just keep know lehrer like lot low lower make middle mitt money more most mr new no not now o obama #obama obama's one open out
people plan president #presidentialdebate question romney #romney romney's same say saying small some something speech sure system
take talking tax taxes #tcot trade trickle tuition want wants way well we've where woman years

Topic 1

SergeantRob Uncanny how Pres. Obama picked out blue tie -- Gov. Romney wearing red tie -- for the #debates. Good color coordination there.10/4/
HazelnutLive I like that one is wearing a red tie & the other is wearing blue--would hate to seem like they agree on tie color #denverdebate #debatef
VOANoticias What color would the Libertarian tie be? Either way -- Its MIA... RT @torreymspears red tie / blue tie. Terrific. #debate2012 #denverde
Uncucumbered Barack has a blue tie and Mitt's tie is red. This is to reinforce party ID among low information voters & forgetful heavy drinkers. #deba
1_state Mitt Romney is a liar man!!! He looks like Satan!!! JMT. AND HE CAN HELP WITH HOMES BUT WON'T CUZ DA BITCH DON'T WANNA C DA MII
orlandolatino that must be so awkward to stand together like friends -- they're like I hope you slip and fail -- just like your policies #debates10/4/20
DonatoAngeles I'm liking Lehrer early -- he's keeping them honest and ensuring it's a debate -- not just two people giving a speech at the same time

Topic 2

politaire First Q: Creating jobs: Obama: Invest educ. -- training -- change tax code for small biz -- invest in US -- No top down #debate #Obama2
SEIA 1st Q in #debates: how would you create more #jobs? Energy independence -- small business? How about #solar: 5 -- 600 businesses and +10
thelockedwonder #Romney : we need to crack down on China -- champion small businesses -- I know what it takes to hire ppl and get small busines
Nv912patriot #denverdebate @FoxNews Green energy?? How much more do we lose in bankrupt green energy??? No answer on tax code -- skate ar
NewsCPR #Romney says jobs plan includes energy ind -- more trade in latin america -- better schools -- balanced budget -- smaller budget #Debat
UPIDebates Obama: I think we've got to invest in education and training... develop new sources of energy... fix our tax code #debates #denverdebate
Auriandra CNBC/NYT: Did President Obama Really Cut Small Business Taxes 18 Times? http://t.co/fqLKftFd yes: tax cuts & 'tax incentives' #debate20

Topic 3

MarquitaCheron I know that's right -- President Obama just gave his wife a shout out and wished her a Happy Anniversary before he addressed anyth
talkingblonde Didn't like Obama's start still has answered question I don't care if it's his anniversary! Babbling now #tcot #Obama #debate10/4/2012
AdamGrowe The 1st rule of debate club is: you do not talk about debate club. The 2nd rule: you do not call the First Lady sweetie. #PresidentialDebat
ThaAfro It is so surreal seeing President Obama speak in real time. I get to see it so rarely. I feel like I'm watching a prepared speech #debates10/4/2
HollisterNCiroc Debate with Mrs. Obama re: working on their anniversary: JUST BEGUN. #debate2012 #obama #zingerProm10/4/2012 10:08:45 AM
DaMisanthrope I know I'm live-tweeting but I don't really want to see live tweets popping up on this live debate I'm watching as well. #debates10/4/2
Bacon4Pres Debate with Romney: just started; Debate with Mrs. Obama re: working on their anniversary: JUST BEGUN. #debate2012 #obama #zinge

Topic 4

miniCOOPER__ I dont care what no one says -- everytime mitt romney speaks he just sounds like a sneaky snake thats just waiting to fuck up some
loverboy843 ABOUT TO GET MY POPCORN N WATCH DA PRESIDENTIAL DEBATE CUZ #TEAMOBAMA IS GONNA MAKE #TEAMROMNEY LOOK LIKE A PI
Killer_Orca42 Don't get me wrong -- I think Obama seems like I really cool guy -- but he's an ass of a president. (Literally) #politicalpuns10/4/2012
A_CittY22 A lot of ppl screaming #TeamObama an don't even kno half the shit that's goin on in the fuckin world they just want somethin to tweet bou
BloombergView Wth is Obama saying right now? He won't answer the damn question better yet he talks about Romney. #NOBAMA he sucks. Get em
TheScarletRose RT @jasonschreier: Trying to watch the debate on my xbox and holy shit mitt romney just killed like four guys no wait this is dishonor
_iHeartLo Lowkey some of u niggas just like President Obama bc hes black! I wish Romney was black so we could see what yall would say then...10/4

Figure 11. Developed UI for the proposed method

tweets from that topic cluster. The size of each word is proportional to its frequency in the tweets of that cluster. This gives users a brief idea about the topic discussed in the tweets in that particular cluster. The left panel lists the most frequent words observed in the tweets in the whole search result and also shows the number of tweets containing those keywords. Users may click on any particular keyword to see all the tweets containing that keyword.

VII CONCLUSION

In this paper, we present a method for recommending the search users a set of tweets that can best delineate an ongoing real-time event. The proposed method is based on a hypothesis that the discussion points that are common in the majority of relevant tweets generated during a major public event, are motivated by the important occurrences in the event. Hence, by identifying popular discussion points in the collection of relevant tweets, and recommending set of tweets comprising many of those discussion points, the proposed method can delineate the proceeding of a real-time event. However, the method would only be useful for those events that can elicit a large public response in Twitter, and that takes place within certain pre-defined period of time. As the model is unsupervised, which requires no prior knowledge about the event, it can be leveraged to generate journalistic summary of any real-time public event satisfying the aforementioned criteria. Evaluation performed on a large set of relevant tweets posted during the first US presidential debate between President Obama and Governor Romney revealed that the proposed model could delineate the proceeding of the event with 81.6% precision and up to 80% recall for some debate segments.

References

[1] M. Busch, K. Gade, B. Larson, P. Lok, S. Luckenbill, and J. Lin, "Earlybird: Real-time search at twitter," in *Proc. IEEE Data Engineering (ICDE)*. IEEE, 2012, pp. 1360–1369.

[2] K. Tao, F. Abel, C. Hauff, G.-J. Houben, and U. Gadiraju, "Groundhog day: near-duplicate detection on twitter," in *Proc. of WWW*, 2013, pp. 1273–1284.

[3] M. Naaman, J. Boase, and C.-H. Lai, "Is it really about me?: message content in social awareness streams," in *Proc. of CSCW*. ACM, 2010, pp. 189–192.

[4] M. S. Bernstein, B. Suh, L. Hong, J. Chen, S. Kairam, and E. H. Chi, "Eddi: interactive topic-based browsing of social status streams," in *Proc. of UIST*. ACM, 2010, pp. 303–312.

[5] M. A. H. Khan, D. Bollegala, G. Liu, and K. Sezaki, "Multi-tweet summarization of real-time events," in *Proc. of ASE/IEEE In-*

ternational Conference on Social Computing. IEEE, 2013.

[6] D. Chakrabarti and K. Punera, "Event summarization using tweets," in *Proc. of ICWSM.* AAAI, 2011, pp. 66–73.

[7] B. Sharifi, M.-A. Hutton, and J. K. Kalita, "Experiments in microblog summarization," in *Proc. of IEEE Second International Conference on Social Computing*, 2010.

[8] J. Nichols, J. Mahmud, and C. Drews, "Summarizing sporting events using twitter," in *Proc. of IUI.* ACM, 2012, pp. 189–198.

[9] Y. Hu, A. John, D. D. Seligmann, and F. Wang, "What were the tweets about? topical associations between public events and twitter feeds," in *Proc. ICWSM.* AAAI, 2012.

[10] B. OConnor, M. Krieger, and D. Ahn, "Tweetmotif: Exploratory search and topic summarization for twitter," in *Proc. of ICWSM.* AAAI, 2010, pp. 2–3.

[11] A. Marcus, M. S. Bernstein, O. Badar, D. R. Karger, S. Madden, and R. C. Miller, "Twitinfo: aggregating and visualizing microblogs for event exploration," in *Proc. of CHI.* ACM, 2011, pp. 227–236.

[12] L. Hong and B. D. Davison, "Empirical study of topic modeling in twitter," in *Proc. of the First Workshop on Social Media Analytics.* ACM, 2010, pp. 80–88.

[13] D. Ramage, S. Dumais, and D. Liebling, "Characterizing microblogs with topic models," in *Proc. of ICWSM*, vol. 5, no. 4. AAAI, 2010, pp. 130–137.

[14] Y. Wang, E. Agichtein, and M. Benzi, "Tmlda: efficient online modeling of latent topic transitions in social media," in *Proc. of the 18th ACM SIGKDD.* ACM, 2012, pp. 123–131.

[15] D. Blei, A. Ng, and M. Jordan, "Latent dirichlet allocation," *the Journal of machine Learning research*, vol. 3, pp. 993–1022, 2003.

[16] J. Weng, E. Lim, J. Jiang, and Q. He, "Twitterrank: finding topic-sensitive influential twitterers," in *Proc. of WSDM.* ACM, 2010, pp. 261–270.

[17] W. X. Zhao, J. Jiang, J. Weng, J. He, E.-P. Lim, H. Yan, and X. Li, "Comparing twitter and traditional media using topic models," in *Advances in Information Retrieval.* Springer, 2011, pp. 338–349.

[18] X. Yan, J. Guo, Y. Lan, and X. Cheng, "A biterm topic model for short texts," in *Proc. of WWW*, 2013, pp. 1445–1456.

[19] S. Brody and N. Elhadad, "An unsupervised aspect-sentiment model for online reviews," in *NAACL-HTL.* ACL, 2010, pp. 804–812.

[20] Z. Niu, D. Ji, and C. Tan, "I2r: Three systems for word sense discrimination, chinese word sense disambiguation, and english word sense disambiguation," in *Proc. of the 4th International Workshop on Semantic Evaluations.* ACL, 2007, pp. 177–182.

[21] E. Levine and E. Domany, "Resampling method for unsupervised estimation of cluster validity," *Neural computation*, vol. 13, no. 11, pp. 2573–2593, 2001.

[22] S. Brin and L. Page, "The anatomy of a large-scale hypertextual web search engine," *Computer networks and ISDN systems*, vol. 30, no. 1, pp. 107–117, 1998.

[23] R. Mihalcea and P. Tarau, "Textrank: Bringing order into texts," in *Proc. of EMNLP*, vol. 4. Barcelona, Spain, 2004, pp. 404–411.

[24] A. Hulth, "Improved automatic keyword extraction given more linguistic knowledge," in *Proc. of EMNLP.* ACL, 2003, pp. 216–223.

[25] K. Toutanova, D. Klein, C. Manning, and Y. Singer, "Feature-rich part-of-speech tagging with a cyclic dependency network," in *Proc. of NAACL-HLT-Volume 1.* ACL, 2003, pp. 173–180.

[26] F. Smadja, "Retrieving collocations from text: Xtract," *Computational linguistics*, vol. 19, no. 1, pp. 143–177, 1993.

[27] T. Dunning, "Accurate methods for the statistics of surprise and coincidence," *Computational linguistics*, vol. 19, no. 1, pp. 61–74, 1993.

[28] C. Manning and H. Schütze, *Foundations of statistical natural language processing.* MIT press, 1999.

[29] J. Kleinberg, "Authoritative sources in a hyperlinked environment," *Journal of the ACM (JACM)*, vol. 46, no. 5, pp. 604–632, 1999.

[30] D. Bollegala, Y. Matsuo, and M. Ishizuka, "Measuring semantic similarity between words using web search engines," in *Proc. of WWW*, vol. 7, 2007, pp. 757–786.