Conversation Analysis on Social Networking Sites

Rami Belkaroui, Rim Faiz, Aymen Elkhlifi

Conversation Analysis on Social Networking Sites

Rami BELKAROUI
LARODEC, ISG Tunis
University of Tunis
Bardo,Tunisia
rami.belkaroui@gmail.com

Rim FAIZ
LARODEC, IHEC Carthage
University of Carthage
Carthage Presidency, Tunisia
rim.faiz@ihec.rnu.tn

Aymen ELKHLIFI
LALIC, Paris Sorbonne University
28 rue Serpente Paris 75006, France
aymen.elkhlifi@paris4.sorbonne.fr

***Abstract*—With the explosion of Web 2.0, people are becoming more communicative through expansion of services and multi-platform applications such as microblogs, forums and social networks which establishes social and collaborative backgrounds. These services can be seen as very large information repository containing millions of text messages usually organized into complex networks involving users interacting with each other at specific times. Several works focused only to retrieve separate tweets or those sharing same hashtags, but, it is not powerful enough if the goal of the search is to retrieve relevant tweets based on content. In addition, finding good results concerning the given subjects needs to consider the entire context. However, context can be derived from user interactions.**

In this work, we propose a new method to retrieval conversation on microblogging sites. It's based on content analysis and content enrichment. The goal of our method is to present a more informative result compared to conventional search engine. To valid our method, we developed the TCOND system (Twitter Conversation Detector) which offers an alternative, results to keyword search on twitter and google. We have evaluated our method on collected social network corpus related to specific subjects, and we obtained good results.

***Keywords*-Social Network, Twitter, Conversation retrieval, social media, user interactions.**

I. Introduction

Recent years have revealed the accession of interactive media, which gave birth to a huge volume of data produced by users called User Generated Content (UGC) in blogs and microblogs more precisely. These Microblogging services like Twitter, attract more and more users due to the ease and the speed of information sharing especially in real time. In addition, microblogging services [1] gives people the ability to communicate, interact and collaborate with each other, reply to messages from others and create conversations. Furthermore, microblogs tend to become a solid media for simplified collaborative communication.

Twitter, the microblogging service addressed in our work, is a communication mean and a collaboration system that allow users to share short text messages, which doesn't exceed 140 characters with a defined group of users called followers. Users can reply to each other simply by adding @sign in front name user they are replying to. This set of socio-technical features has made possible for Twitter to host a wide range of social interactions from the broadcasting of personal thoughts to more structured conversations among groups of friends [2]. While communicating people share different kind of information like common knowledge, opinions, emotions, information resources and their likes or dislikes. The analysis of those communications can be useful for commercial applications such as trends monitoring, reputation management and news broadcasting. In addition, one of main characteristic of Twitter is that users are not limited to produce contents, they can get involved indirectly in conversations with other users by liking and sharing user's posts.

This paper proposed a conversation retrieval method which can be used to extract conversation from twitter. Comparing with current methods, the new proposed not only extract directly reply tweets, but also relevant tweets which might be retweets or comments and other possible interactions. The method extract extensive posts beyond conventional conversation, which is much better called a discussion. In particular, the contributions of this paper are: first, the ability to provide an informative result for users' information needs based on user's content interactions analysis. Second, the definition of ranking function to order conversation results. Finally, the evaluation of the proposed method impact on keyword search results.

The rest of the document is organized as follows: we begin by presenting related work in related domains such as fourms discussion, Email threads. Then, we focus on more recent works addressing conversation retrieval on microblogging sites. In section 3, we propose our method allows to extract social user's content interactions. In section 4, we describe a set of ranking measures. The experimentation and evaluation results are detailed in section 5. Finally, we conclude and present same future works.

II. Related Work

Conversation retrieval topic is relevant for three main domains: forum search, email/thread detection and Twitter, which is the main domain used in our work. We present following these domains.

A. Related Work in Forum/ Threads Search

An online forum is a Web application for holding discussions and posting User Generated Content in a particular domain, such as sports, recreation, techniques, travel, etc. In forums, conversations are represented as sequences of posts or threads, where the posts reply to one or more earlier posts. Several studies have looked at

identifying the structure of a thread, question-answer pairs or responses that relate to a previous question in the thread.

There are many works on searching forum threads that dealt with the reply-chains structure or reply-trees. [3] has concentrated on identifying the thread structure when explicit connections between messages are missing. Despite the fact that replies to posts in microblogging sites, are commonly explicit, it is proved that different autonomous conversations may be developed inside the same replies thread. Furthermore, distinct threads may belong related to macro-conversations. For example, being Twitter hashtags that connect separate threads by common topic. In [4] authors represent the principal differences between traditional IR tasks and searching in newsgroups. They use a mesures combination such as author metrics (posts number, number of replies, etc.) and features threads.

In question and answering, there are two streams of similar work: the first one is to find the best set of answers for a query, and the second is to identify question and answer pairs to build a querying system knowledge base.

In [5], authors aimed to discover the most relevant answer in question threads. They implemented a discussion-bot to automatically answer student queries in a threaded discussion by combining lexical similarity, speech acts and reputation of the author of posts into a similarity measure. To discover best potential answers, they used the HITS algorithm to find posts that are most likely to be answers to the initial question post. However, extract possible answers (the most informative message) using an algorithm with a rule-based traverse that is not optimal for selecting a best answer; consequently, the result may comprise redundant or incorrect information.

Similar to [5], [6] detected threads in which first posts are questions and its corresponding answers belong to the thread. To detect answers, they used features including the positions of the candidate answer posts, authorship and likelihood models based on content. [7] tested different combinations of these features with an SVM classifier and found post position and authorship result in the highest accuracies. For answers, [8] used language models to construct weighted similarity graphs between each question and the set of candidate answers. For each detected question, a page-rank like propagation algorithm was utilized to determine and rank the set of candidate answers.

B. Related Work in Email Threads Search

Previous research has been focused on using email structure especially emails threads [9], [10]. Thread detection is an important task which has attracted significant attention [11].

Email is one of the most important tools for treating conversations between people. Generally, a typical user mailbox encloses hundreds of conversations. Few works indirectly address to the problem of thread reply reconstruction. Accorded to [9], the detection of these conversations has been identified as an important task. Clustering the messages into coherent conversations useful to applications, among them, it gives users the opportunity to see a messages greater context they are reading and collating related messages automatically. [12] point out that by using text matching techniques to messages portions, it will be possible to detect threads effectively.

In [11] the authors suggested a method that allows to assemble messages having the same subject attributes and send them among the same group of people. However, conversations may span several threads with similar (but not exact) subject lines. Furthermore, a conversation not include all the participants in all the messages. In the same way, [13] developed an email client extension that makes it possible to clusters messages by topic. However, their clustering approach is focused on topic detection, hence messages belonging to different conversations on the same topic will be clustered together. In addition, [14] recreated reply emails chains, called email threads. The authors suggested two approaches, one based on using header meta-information, the other based on timing, subject and emails content. But, this method is specific for emails and the features cannot be easily extended for microblogs conversation construction.

[15] proposed an approach for conversation detection based on email attributes. They started by sharpening the distinction between email threads and conversations. The task was to assemble messages into consistent conversations using a function of similarity that takes all relevant email attributes, for example message subject, participants, date of submission, and message content. This method is similar to our method. We will use criteria for detecting tweets that will be in the same conversation.

C. Conversational Aspects on Microblogging Sites

Conversation retrieval is a new search paradigm for microblogging sites. It result from the intersection of Information Retrieval and Social Network Analysis (SNA). Most of Microblogs services provide a way to retrieve relevant information [16], but lack the ability to provide all discussion tweets. In addition, existing conversation retrieval approaches for microblogging sites [17] have so far focused on the particular case of a conversation formed by directly replying tweets.

In [18] the authors concentrated on different microblogging conversations aspects. They proposed a simple model that produces basic conversation structures taking into account the identities of each conversation member. Other related works focusing on different aspects of microblogging conversations are [19], [20] that deal respectively with conversations tagging and topics identification. These works presents limitations, most relevant messages don't even contain any hashtag.

Recently, various researches focused on the task of conversation retrieval in microblogs [21], [22], [23], [24]. [21] proposed a user-based tree model for retrieving conversations from microblogs. They considered only tweets that directly respond to other tweets by the use of @sign as a marker of addressivity. The advantage of this approach is to have a coherent conversation based on the direct links between users. The downside is that this method does not

consider tweets that do not contain the @sign. Similarly [17] proposed a method to build conversation graphs, formed by users replying to tweets. In this case, a tweet can only directly reply to other tweet. However, users can get involved indirectly in conversations communities by commenting, liking, sharing user's posts.

III. A New Method for Twitter Conversation Detection: TCOND

A. Twitter Conversation Definition

[21] defined conversation as a tree where nodes represent short text messages posted by users at specific timestamps in reply to a parent nodes. Similarly [17] defined a conversation as a reply tree which is a graph where vertices are tweets and a directed edge represents one tweet that is a reply to another.

Contrary to [17], [21], we define a conversation as a set of short text messages posted by a user at specific timestamps on the same topic. This messages can be directly replied to other users by using "@username" or indirectly by liking, retweeting, commenting and other possible interactions.

B. Our System Architecture

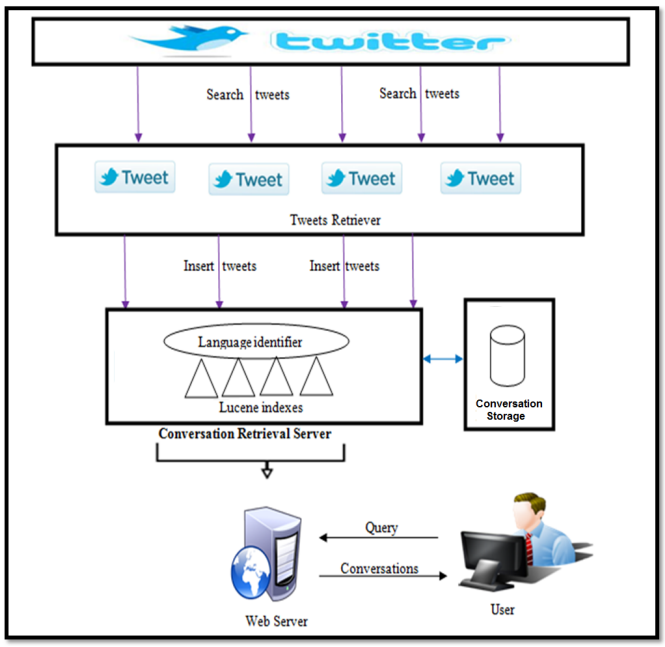

Figure 1. Our System "TCOND" Architecture

In Figure1, we have illustrated our system architecture that is made of four applications:

- One Web/Application Server (interface between users and system).
- One Conversation Server (storage, indexing and search of conversations).
- One Trend Server (gets trending topics and distribute them among tweet retrieval clients).
- One Tweet Retrieval Clients (to retrieve tweets in parallel).

The Trend Server repeatedly gets from Twitter API the current trending topics, i.e., the most discussed topics. These topics are distributed to the Tweet Retrieval Clients that use Twitter Search API[1] to get the corresponding tweets and also to retrieve the conversation chains. These are sent to the Conversation Server, that stores tweets and other information like the number of followers of every user participating to the conversations. The IR engine Lucene[2] is used to index the conversations text and to associate it to their identifiers, from which they can be later efficiently retrieved. Users may then query the system through a Web application.

C. Conversation Detection on Twitter Microblogs

We propose a method which combines a set of conversational features and the directly exchanged text messages in order to extract extensive posts beyond conventional conversation. In the following, we will present more details about our two approach steps.

1) Direct Conversation Detection: In this step, we aim to collect all tweets in reply directly to other tweets. Obviously, a reply to a user will always begin with "@username". Our goal in this step is to create reply tree. The reply tree construction process consists of two algorithms run in parallel **Recursive Root Finder Algorithm** and **Iterative Search Algorithm**.

Algorithm 1 Recursive Root Finder (A:twitter)

Let T be a tweet collected from Twitter (ID tweet)
while (type (T_i) !=root) **do**
  Extract T_i- 1 by matching field "in reply to status id"
end while
A : twitter = A : twitter – 1

Let T0 is the root (first tweet published) of the conversation C and T is a single tweet of the conversation retrieved. Let consider T_i the type of tweet T. A tweet can have three types: root, reply or retweet. The goal of the Recursive Root Finder Algorithm is to identify the conversation root T0 given T. Note that when the algorithm starts,$|T|$ is not known.

Algorithm 2 Iterative Search

N = the set of all tweets in C
for j = 1 to j $\leq |N|$ **do**
  Search for tweets addressed to author.
  Extract replies to T_i by matching field in reply to status id.
end for

Once, the conversation root T0 has been established, the Iterative Search Algorithm is used to seek the remainder of conversation C by searching all tweets (named N) which compose C addressed to T_i using matching field "in reply to status id". It is run repeatedly until some conditions, indicating that the conversation has ended, are met.

[1] http://dev.twitter.com/doc/get/search
[2] www.lucene.apache.org

2) Conversational Features: To the best of our knowledge, there has not been previous work on the structure of reply-based on indirectly conversation. Therefore, we define a new features that may help to detect tweets related indirectly to a same conversation. The goal of this step is to extract tweets that may be relevant to conversation without the use of "@username". We use the following notations in the sequel:

- t_i is a tweet present in direct conversation (tweets in reply to other tweets directly).
- t_j is a tweet that can be linked indirectly to conversation.

The features we used are:

- ***Using the same URL***:

Twitter allows users to include URL as a supplement information to their tweets. By sharing an URL, an author would enrichment the information published in his tweet. This feature is applied to collect tweets that share the same URL. P1 is a binary function.

$$P1(t_i, t_j) = \begin{cases} 1 \text{ if t contains the same URL.} \\ 0 \text{ otherwise.} \end{cases} \quad (1)$$

- ***Hashtags Similarity***:

The # symbol, called hashtag, is used to mark a topic in a tweet or to follow conversation. Any user can categorize or follow topics with hashtags. We used this feature to collect tweets that share the same hashtags. P2 is a binary function.

$$P2(t_i, t_j) = \begin{cases} 1 \text{ if t contains the same hashtag.} \\ 0 \text{ otherwise.} \end{cases} \quad (2)$$

- ***Tweets Time Difference***:

The time difference is highly important feature for detecting tweets linked indirectly to conversation. We use the time attribute to efficiently remove tweets having a large distance in terms of time compared to conversation root. The difference in time, measured in seconds, between two tweets t_i, t_j.

- ***Tweets Publication dates***:

Date attribute are highly important for detecting conversations. Users tend to post tweets about conversational topic within a short time period. The Euclidean distance has been used to calculate how similar two posts publication dates are.

- ***Content***:

The criterion Content refers to the thematic relevance traditionally calculated by IR systems standards. We compute the textual similarity between each element in t_i, t_j taking the maximum value as the similarity measure between two messages. The similarity between two elements is calculated using the well-known tf-idf cosine similarity, $sim(t_i, t_j)$.

- ***Similarity Function***:

Finally, the similarity between tweets indirectly linked to conversation and tweets which are present in the reply-tree is calculated by a linear combination between their attributes.

IV. Conversation Ranking

In the last section, we defined a method to detect conversation. Now, we introduce a ranking function that can be used to rank results of conversations search task. This is an aggregation of other functions representing the relative importance of different conversation aspects. It's worth noting that most of the measures indicated in the following have been defined in other contexts, and their practical usefulness has been proved several times. Here we propose their joint application to ranking conversations microblogs search task.

The first aspect regards the exchanged text message. To rank text messages we can compute their **relevance** with regard to some information requirements. However, text relevance of single tweets can be evaluated using any IR model, and to evaluate the relevance of an entire conversation we can calculate the average relevance of its interactions. Many standard models such as the boolean, vector-space or more complex models can be used, but this is a traditional topic in IR for which we do not present details here. In our implementation, we use the Appache Lucene library with its built-in ranking functions. In addition, the **messages popularity** can be defined in several different ways to evaluate the ranking conversations. This can be usually computed easily in Social Network, e.g., counting the number of likes, sharings or retweets received by the message. In the same way, we can use **conversation frequency** (number of interactions) that may tell us something more than a single message can. Finally, the same people may exchange messages, but at different times this may be more or less important and the rate at which messages are exchanged can be indicative of the level of interest/emotion attached to conversation. Therefore, we will also use **time-related measures**. In our case, computed the difference between an input timestamp and an internal timestamp of conversation (starting, medium or ending).

V. Experiments and Results

The following experiment has been designed to gather some knowledge on the impact of our results on end-users. For this experiment we have selected two events and queried our dataset using Google[3], Twitter search engine[4] and our method. Then we have asked a set of assessors to rate the top-10 results of every search task, to compare these approaches. In order to measure the quality of the results, we use the Normalized Discounted Cumulative Gain (NDCG) at 10 for all the judged event. In addition, we used a second metric which is the Precision at top 10. In the following, we first describe the experimental

[3]www.google.com
[4]Search.twitter.com.

setting, then we present the results and finally we provide an interpretation of the data.

A. Experimental Settings

The analysis presented in this section is based on a social database collected over a period of the first two weeks of July 2013 by monitoring microbloging system Twitter posts (tweets). In particular, we used a sample of about 63 000 posts containing trending topic keywords. Trending topics have been determined directly by Twitter and we have selected the most frequent ones during the monitoring period.

To evaluate the results of our search tasks we have used a set of 60 assessors with three relevance levels, namely highly relevant (value equal to 2), relevant (value equal to 1) or irrelevant (value equal to 0). The assessors selected among students and colleagues of the authors (with backgrounds in computing and social sciences), on a voluntary base, and no user was aware of the underlying systems details. Every user was informed of two events happened during the sampling period: the first event is "the 100st edition of the Tour France" and the second is "the death of computer mouse inventor Douglas Engelbart". For each event we performed three searches:

1. One using Google.
2. One using Twitter Search.
3. One using our method (TCOND).

The evaluators were not aware of which systems had been used. Every user for each search task was presented with two conversations selections, one for each of the previous options with the corresponding top-10 results.

B. Experimental Outcomes and Interpretation Results

	P@10 (Average%)	NDCG (Average%)
Task1		
Google	59.62	56.86
Twitter	65.73	59.71
TCOND	**73.28**	**64.52**
Task2		
Google	57.31	56.02
Twitter	62.78	58.45
TCOND	**67.27**	**62.73**

Table I
TABLE OF VALUES FOR COMPUTING OUR WORKED EXAMPLE

We compare our conversation retrieval method with the results returned by Google and by Twitter search engine using two metrics namely the P@10 and the NDCG@10. From this comparison, we obtained the values summarized in Table 1, where we notice that our method overcomes the results given by both of Google and Twitter. The reason of these promising values is the fact that we combine a set of conversational features and direct replies method to retrieve conversation may have a significant impact on the users' evaluation.

Concentration on the first messages selection (related to the Tour de France), conversations obtained with our method receive higher scores with compared to Google and Twitter's selection. By switching to the second event selection (related to the death of computer mouse inventor), we can see a similar scenario that our method's selection is the one with the higher scores. According to the free comments of some users and following the qualitative analysis of the posts in the two selections we can see that Google and twitter received lower scores not because they contained posts judged as less interesting, but because some posts were considered not relevant with regard to the searched topic.

Focusing on the two messages selection, we observe that both conversations selections obtained with twitter search has higher scores with respect to Google's selection. These results lead us toward a more general interpretation of the collected data. It appears that the usage of social metrics have a significant impact on the users' degree interest in the retrieved posts. In addition, the process of retrieving conversations from Social Network differs from traditional Web information retrieval, it involves human communication aspects, like the degree interest in the conversation explicitly or implicitly expressed by the interacting people.

C. Properties of Conversations

In this part, we state the main observations about the top-10 conversations results detected using our conversation retrieval method. We study the conversations distribution duration (number of hours since the original tweet until the last tweet) and conversations frequency (the number of messages that compose conversation).

- **Conversations Frequency**

Figure 2. Conversation Levels Deep

We examined the conversations'frequency which is the length of the maximum path to a leaf from the root (Figure2). Most conversations that occur in Twitter appear to be dyadic exchanges of three to five messages sent over a period of 15 to 30 minutes. Of all tweets that generated a reply, 84.81% have only one reply. Another 10.7% attracted a reply to the original reply the conversation was two levels deep. Only 1.53% of Twitter conversations are three levels deep after the original tweet, there is a reply, reply to the reply, and reply to the reply of reply.

- **Conversations Duration**

The analysis we made has demonstrated that the majority of conversations are not continued if the oldest tweet in conversation is more than 5 hours old.

Figure 3. Conversations Duration

We found that 97.87% of @replies take place within the first hour of the original tweet being published, while

an additional 0.98% of replies happen in the second hour. Subsequently, reply activity dramatically declines as it is shown in Figure3. There is no way to know for certain that a conversation will not be replied to at some indeterminate time in the future.

VI. Conclusion

In this paper, we explored a new method for detecting conversations on microblogging sites: an information retrieval activity exploiting a set of conversational features in addition to the directly exchanged text messages to retrieve conversation. In particular, we have defined a set of metrics as relevance, popularity, timeliness and frequency to be used in the computation of the ranking of a conversation.

The previous observations indicate that conversations are typically short and do not provide all the context of users' interactions. In addition, Our experimental results show the importance of using conversational features and considering all the possibilities of interactions between the participants in order to provide the entire conversation that has been published as well as its context.

We believe that the type of conversations described in this work can benefit applications that rely on microblogging posts from the end user's perspective. Future work will further research the conversational aspects by including human communication aspects, like the degree of interest in the conversation explicitly or implicitly expressed by the interacting people and their influence/popularity by gathering data from multiple sources from Social Networks in real time.

References

[1] L. B. Jabeur, L. Tamine, and M. Boughanem, "Uprising microblogs: A bayesian network retrieval model for tweet search," in *Proceedings of the 27th Annual ACM Symposium on Applied Computing*. New York, NY, USA: ACM, 2012, pp. 943–948. [Online]. Available: http://doi.acm.org/10.1145/2245276.2245459

[2] D. Boyd, S. Golder, and G. Lotan, "Tweet, tweet, retweet: Conversational aspects of retweeting on twitter," in *Proceedings of the 2010 43rd Hawaii International Conference on System Sciences*, ser. HICSS '10. Washington, DC, USA: IEEE Computer Society, 2010, pp. 1–10. [Online]. Available: http://dx.doi.org/10.1109/HICSS.2010.412

[3] Y.-C. Wang, M. Joshi, W. W. Cohen, and C. P. Ros, "Recovering implicit thread structure in newsgroup style conversations." in *ICWSM*, E. Adar, M. Hurst, T. Finin, N. S. Glance, N. Nicolov, and B. L. Tseng, Eds. The AAAI Press, 2008. [Online]. Available: http://dblp.uni-trier.de/db/conf/icwsm/icwsm2008.html

[4] W. Xi, J. Lind, and E. Brill, "Learning effective ranking functions for newsgroup search," in *Proceedings of the 27th annual international ACM SIGIR conference on Research and development in information retrieval*, ser. SIGIR '04. New York, NY, USA: ACM, 2004, pp. 394–401. [Online]. Available: http://doi.acm.org/10.1145/1008992.1009060

[5] D. Feng, E. Shaw, J. Kim, and E. Hovy, "Learning to detect conversation focus of threaded discussions," in *Proceedings of the main conference on Human Language Technology Conference of the North American Chapter of the Association of Computational Linguistics*, ser. HLT-NAACL '06. Stroudsburg, PA, USA: Association for Computational Linguistics, 2006, pp. 208–215. [Online]. Available: http://dx.doi.org/10.3115/1220835.1220862

[6] L. Hong and B. D. Davison, "A classification-based approach to question answering in discussion boards," in *Proceedings of the 32nd international ACM SIGIR conference on Research and development in information retrieval*, ser. SIGIR '09. New York, NY, USA: ACM, 2009, pp. 171–178. [Online]. Available: http://doi.acm.org/10.1145/1571941.1571973

[7] J. Seo, W. B. Croft, and D. A. Smith, "Online community search using thread structure," in *Proceedings of the 18th ACM conference on Information and knowledge management*, ser. CIKM '09. New York, NY, USA: ACM, 2009, pp. 1907–1910. [Online]. Available: http://doi.acm.org/10.1145/1645953.1646262

[8] S. Ding, G. Cong, C. yew Lin, and X. Zhu, "Using conditional random fields to extract contexts and answers of questions from online forums," in *In Proceedings of Association for Computational Linguistics ACL-08: HLT*, 2008, pp. 710–718.

[9] B. Kerr, "Thread arcs: an email thread visualization," in *Proceedings of the Ninth annual IEEE conference on Information visualization*, ser. INFOVIS'03. Washington, DC, USA: IEEE Computer Society, 2003, pp. 211–218. [Online]. Available: http://dl.acm.org/citation.cfm?id=1947368.1947407

[10] D. Lam, S. L. Rohall, C. Schmandt, and M. K. Stern, "Exploiting E-mail Structure to Improve Summarization," 2002. [Online]. Available: http://citeseerx.ist.psu.edu/viewdoc/summary?doi=10.1.1.12.7056

[11] B. Klimt and Y. Yang, "Introducing the enron corpus." in *The Third Conference on Email and Anti-Spam (CEAS), July 27-28, 2006, Mountain View, California, USA. 2006*, 2004. [Online]. Available: http://dblp.uni-trier.de/db/conf/ceas/ceas2004.html#KlimtY04

[12] D. E. Lewis, K. A. Knowles, B. Smith, and Writes, "Threading electronic mail - a preliminary study." *Information Processing and Management: an International Journal Special issue: methods and tools for the automatic construction of hypertext archive*, vol. 33, no. 2, pp. 209–217, 1997. [Online]. Available: http://dblp.uni-trier.de/db/journals/ipm/ipm33.html#LewisK97

[13] G. Cselle, K. Albrecht, and R. Wattenhofer, "Buzztrack: topic detection and tracking in email," in *Proceedings of the 12th international conference on Intelligent user interfaces*, ser. IUI '07. New York, NY, USA: ACM, 2007, pp. 190–197. [Online]. Available: http://doi.acm.org/10.1145/1216295.1216331

[14] J.-Y. Yeh, "Email thread reassembly using similarity matching." in *The Third Conference on Email and Anti-Spam (CEAS), July 27-28, 2006, Mountain View, California, USA. 2006*, 2006. [Online]. Available: http://dblp.uni-trier.de/db/conf/ceas/ceas2006.html#Yeh06

[15] S. Erera and D. Carmel, "Conversation detection in email systems," in *Proceedings of the IR research, 30th European conference on Advances in information retrieval*, ser. ECIR'08. Berlin, Heidelberg: Springer-Verlag, 2008, pp. 498–505. [Online]. Available: http://dl.acm.org/citation.cfm?id=1793274.1793335

[16] S. Cherichi and R. Faiz, "Relevant information discovery in microblogs - combining post's features and author's features to improve search results," in *KDIR 2013 - Proceedings of the International Conference on Knowledge Discovery and Information Retrieval, Vilamoura, Algarve, Portugal, 19 - 22 September, 2013*, 2013, pp. 128–135.

[17] P. Cogan, M. Andrews, M. Bradonjic, W. S. Kennedy, A. Sala, and G. Tucci, "Reconstruction and analysis of twitter conversation graphs," in *Proceedings of the First ACM International Workshop on Hot Topics on Interdisciplinary Social Networks Research*, ser. HotSocial '12. New York, NY, USA: ACM, 2012, pp. 25–31. [Online]. Available: http://doi.acm.org/10.1145/2392622.2392626

[18] R. Kumar, M. Mahdian, and M. McGlohon, "Dynamics of conversations," in *Proceedings of the 16th ACM SIGKDD international conference on Knowledge discovery and data mining*, ser. KDD '10. New York, NY, USA: ACM, 2010, pp. 553–562. [Online]. Available: http://doi.acm.org/10.1145/1835804.1835875

[19] J. Huang, K. M. Thornton, and E. N. Efthimiadis, "Conversational tagging in twitter," in *Proceedings of the 21st ACM conference on Hypertext and hypermedia*, ser. HT '10. New York, NY, USA: ACM, 2010, pp. 173–178. [Online]. Available: http://doi.acm.org/10.1145/1810617.1810647

[20] S. Song, Q. Li, and N. Zheng, "A spatio-temporal framework for related topic search in micro-blogging," in *Proceedings of the 6th international conference on Active media technology*, ser. AMT'10. Berlin, Heidelberg: Springer-Verlag, 2010, pp. 63–73. [Online]. Available: http://dl.acm.org/citation.cfm?id=1886192.1886204

[21] M. Magnani, D. Montesi, G. Nunziante, and L. Rossi, "Conversation retrieval from twitter," in *Proceedings of the 33rd European conference on Advances in information retrieval*, ser. ECIR'11. Berlin, Heidelberg: Springer-Verlag, 2011, pp. 780–783. [Online]. Available: http://dl.acm.org/citation.cfm?id=1996889.1997002

[22] M. Magnani, D. Montesi, and L. Rossi, "Information propagation analysis in a social network site." in *ASONAM*, N. Memon and R. Alhajj, Eds. IEEE Computer Society, 2010, pp. 296–300. [Online]. Available: http://dblp.uni-trier.de/db/conf/asunam/asonam2010.html

[23] A. Bruns and J. E. Burgess, "#Ausvotes: how twitter covered the 2010 australian federal election," *Communication, Politics and Culture*, vol. 44, no. 2, pp. 37–56, 2011. [Online]. Available: http://search.informit.com.au/documentSummary;dn=627330171744964;res=IELHSS

[24] M. Magnani, D. Montesi, and L. R. 0003, "Conversation retrieval for microblogging sites." *Information. Retrieval Journal*, vol. 15, no. 3-4, pp. 354–372, 2012. [Online]. Available: http://dblp.uni-trier.de/db/journals/ir/ir15.html#MagnaniMR12