Summarizing a Document Stream

Hiroya Takamura, Hikaru Yokono, and Manabu Okumura

Precision and Intelligence Laboratory,
Tokyo Institute of Technology
{takamura,oku}@pi.titech.ac.jp,yokono@lr.pi.titech.ac.jp

Abstract. We introduce the task of summarizing a stream of short documents on microblogs such as Twitter. On microblogs, thousands of short documents on a certain topic such as sports matches or TV dramas are posted by users. Noticeable characteristics of microblog data are that documents are often very highly redundant and aligned on timeline. There can be thousands of documents on one event in the topic. Two very similar documents will refer to two distinct events when the documents are temporally distant. We examine the microblog data to gain more understanding of those characteristics, and propose a summarization model for a stream of short documents on timeline, along with an approximate fast algorithm for generating summary. We empirically show that our model generates a good summary on the datasets of microblog documents on sports matches.

1 Introduction

Microblogs such as Twitter[1] have recently gained popularity. They are different from other conventional blogs in that entries of microblogs are very short, called tweets and 140-characters long in Twitter, and therefore are mainly used to describe what users are doing or how they are feeling at this very moment, while conventional blogs usually deal with more coarse units of time such as days or weeks. As a new source of information, microblogs have drawn a great deal of attention of the public.

One distinctive aspect of microblogs as a source of information is that numerous short documents on a single topic are posted by many users. A typical example of topics of our interest is sports matches on TV. While people are watching a sports match on TV, they often post short documents about the match on the microblogs. We call such a set of short documents on timeline, *a document stream*. The purpose of this paper is to propose a summarization model for document streams of microblogs. With such a technique, we would be able to learn what is going on with regard to the topic, or what people think about the topic. Their contents can be description of facts, applause, criticism, or sometimes serious opinions. Sports matches are not the only example that interests us. Another example will be microblogs on TV dramas.

[1] http://twitter.com

P. Clough et al. (Eds.): ECIR 2011, LNCS 6611, pp. 177–188, 2011.

Video streaming service such as Ustream[2] would provide us with more potential application areas of the technique for document stream summarization.

In the document stream summarization, we need to take into consideration that short documents on microblogs are aligned on timeline. Let us take a soccer match as an example of topics. On microblogs, "good pass" at the 10th minute and "good pass" at the 85th minute should be two distinctive events in the match. However, there is often no apparent evidence of their distinctiveness, except posted time. We also need to be aware that documents on microblogs are not always posted at the instant that the event occurs; they are often posted with some delay. In this work, we take the extractive summarization approach [8], in which we choose several short documents from the data in order to generate a summary. Although the sentence extraction is often used in the standard text summarization task, the document extraction is sufficient in the current task since documents are very short in microblogs (not longer than 140 characters in Twitter) and each document usually conveys a simple piece of information. The remaining question is how to select documents. We will answer this question by proposing a new summarization method based on the p-median problem.

2 Numerous Short Documents on Timeline

In order to gain more understanding of characteristics of microblogs, we preliminarily collected and analyzed microblog data[3]. We choose a document stream on a soccer match, namely Japan vs. Hong Kong in East Asian Football Championship 2010, which Japan won with 3-0. We collected Japanese tweets (short documents) about this match from Twitter using Streaming API.

We first counted the frequency of the term "goal" at each time point in the match and obtained Figure 1. The curve in this figure was smoothed with the moving average method for the demonstration purpose.

The curve in Figure 1 has several peaks. We examined the tweets data and found out that they are caused by different events in the match. The peak at around the 25th minute was caused by tweets of users who were complaining about the lack of goals of Japan, which had shown poor performance in the previous scoreless match against China. The peaks at around the 40th, the 80th, the 100th minutes were caused respectively by the first goal of a Japanese striker Keiji Tamada, the goal of Marcus Tulio Tanaka, and the second goal of Keiji Tamada. Tweets on these goal events are often similar to each other. This observation suggests that when we measure the similarity of two documents on timeline during the summarization process, we have to take the difference of the posted times into account so that we can distinguish documents on one event from those on another similar event at another time point.

We also point out that users do not always post tweets right after the event (i.e., goal). For example, the tweets containing "goal" start to increase at the 37th minute, and are frequently posted until around the 55th minute. We need to be able to recognize that those tweets mention a single event.

[2] `http://www.ustream.tv`

[3] This dataset is different from the one used in the empirical evaluation in Section 6.

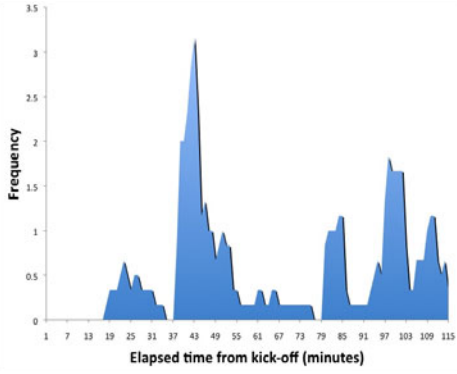


Fig. 1. Frequency of the messages containing the term "goal" in tweets on Japan vs. Hong Kong on Feb. 11, 2010. The curve was smoothed with the moving average method.

These two observations show two opposite characteristics of microblogs; (i) very similar documents can mention different events if they are temporally distant, (ii) documents on a single topic can be posted with some temporal delay. We conjecture that these characteristics are not unique to microblogs on soccer, but are shared by microblogs with other topics as well. We would like to construct a summarization model that can handle these two opposite characteristics.

3 Summarization Model Based on Facility Location Problem

We first introduce a text summarization model proposed by Takamura and Okumura [15], which we will use as the basis of our method. Their model is based on the facility location problems [3]. These problems are applicable to practical issues of determining, for example, hospital location or school location, where the facility is desired to be accessible to the customers in the area. The p-median problem is a facility location problem which has the cardinality constraints that the number of selected facility sites is upperbounded by constant p. We consider here that customer locations are also potential facility sites. In their model, a summary is generated such that every sentence (short document, in our case) in the given data is assigned to and represented by one selected sentence (short document) as much as possible. In the following, we will explain their method as a method for selecting documents, though their method is originally based on sentence extraction.

Let us denote by e_{ij} the extent to which document d_j is inferred by document d_i. We call this score e_{ij} *the inter-documental coefficient*. In the previous work using this model [15], e_{ij} was defined as $e_{ij} = |d_i \cap d_j|/|d_j|$, where d_i is regarded as the set of content words contained in the document, and therefore $d_i \cap d_j$ represents the intersection of d_i and d_j.

If we denote by z_{ij} the variable which is 1 if d_j is assigned to d_i, otherwise 0, then the score of this whole summary is going to be $\sum_{i,j} e_{ij} z_{ij}$, which will be maximized. We next have to impose the cardinality constraint on this maximization so that we can obtain a summary of length p or shorter, measured by the number of documents. Let x_i denote a variable which is 1 if d_i is selected, 0 otherwise. The cardinality constraint is then represented as $\sum_i x_i \leq p$. The p-median summarization model is formalized as follows:

$$
\begin{aligned}
max.\ & \textstyle\sum_{i,j} e_{ij} z_{ij} \\
s.t.\ \quad & z_{ij} \leq x_i; \quad \forall i, j, && (1)\\
& \textstyle\sum_i x_i \leq p, && (2)\\
& \textstyle\sum_i z_{ij} = 1; \ \forall j, && (3)\\
& z_{ii} = x_i; \quad \forall i, && (4)\\
& x_i \in \{0, 1\}; \ \forall i, && (5)\\
& z_{ij} \in \{0, 1\}; \ \forall i, j. && (6)
\end{aligned}
$$

(1) guarantees that any document to which another document is assigned is in a summary. (2) is the cardinality constraint. (3) guarantees that every document is assigned to a document, and (4) means that any selected document is assigned to itself. The integrality constraint on z_{ij} (6) is automatically satisfied in the problem above. Although this optimization problem is NP-hard and intractable in general, if the problem size is not so large, we can still find the optimal solution by means of the branch-and-bound method [4].

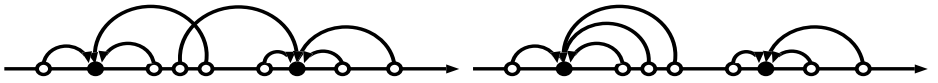
4 Summarizing a Document Stream

On the basis of the text summarization model in Section 3, we will propose a summarization model for a document stream consisting of short documents on timeline.

We focus on the two characteristics of microblogs observed in Section 2; (i) very similar documents can mention different events if they are temporally distant, (ii) documents can be posted with some temporal delay. We explain two distinct ideas for incorporating these two characteristics. The first idea is the modification of coefficients e_{ij}. The second one is the linear-partition constraint on document assignment.

4.1 Modification of Coefficients e_{ij}

If two documents are temporally distant, we would like them to have a small coefficient e_{ij} even if their contents are similar. Thus we multiply e_{ij} with a decreasing function of difference in time as a decay factor. Although there are many possible decreasing functions, we use the function : $0.5^{|t(d_i)-t(d_j)|/\beta}$, where $t(d)$ is the time d was created at (measured by seconds), and β is a positive constant. Exploration for good decreasing functions would be left as future work.



(a) Unconstrained (b) Linear-partition

Fig. 2. Document assignments. The horizontal straight arrow represents the timeline. Black circles represent selected documents. White circles represent unselected documents. The curved arrows represent the assignment relations.

The coefficient of two temporally-distant documents is going to be very small due to the decay factor. The similar documents posted with a slight delay will still have a high coefficient. Note that too large β results in giving a large coefficient to temporally-distant similar documents, and that too small β results in giving a small coefficient to temporally-close similar documents. The final definition of the coefficient is as follows :

$$e_{ij}^{\text{time}} = e_{ij} 0.5^{|t(d_i)-t(d_j)|/\beta}. \tag{7}$$

The proposed model will tend to select longer documents, which cover more words. We also test a variant of the proposed model where we penalize longer documents by multiplying e_{ij}^{time} with length-penalty factor $1/c_i$.

4.2 Linear-Partition Constraint on Document Assignment

Another idea for handling the characteristics of microblogs is to impose, what we call, *the linear-partition constraint* on document assignment in the p-median summarization model in Section 3.

Let us begin with defining the unconstrained assignment and the linear-partition assignment. In our task, the documents are aligned on timeline. The p-median summarization model allows a document to be assigned to any document on timeline. Therefore the arrows indicating document assignment can cross each other as in Figure 2 (a). In the figure, the horizontal straight arrow represents the timeline. Black circles represent selected documents. White circles represent unselected documents, which are assigned to one of the black circles. We call it *the unconstrained assignment*. In contrast, when the arrows do not cross each other, we call it *the assignment with the linear-partition constraint*, or simply *the linear-partition assignment*, illustrated by Figure 2 (b).

We propose to impose the linear-partition constraint on the p-median text summarization model; we use the linear-partition assignment. In this new model, two temporally-distant documents are unlikely to be linked unless all the documents between the two documents are similar. Delay in posted time is disregarded as long as similar documents are present in-between. Temporally-close documents are likely to form a cluster. The linear-partition constraint is incorporated into the p-median text summarization model by adding the following two constraints to the previous optimization model:

$$z_{ij} \leq z_{ik}; \forall i, j, k (j \leq k \leq i),$$
$$z_{ij} \leq z_{ik}; \forall i, j, k (i \leq k \leq j).$$

Although the search space has been reduced by adding these constraints, this problem is not easy to deal with, since the number of constraints is on the order of $O(N^3)$, where N is the number of documents. To our knowledge, there is no algorithm that finds the exact solution within a reasonable computational time.

We should be aware that this model with the linear-partition constraint can capture the characteristics of microblogs only when we can assume that there are no parallel multiple topics. The model should work in the current task of interest. In the situation where there are parallel multiple topics, we would need to use the unconstrained assignment of documents. We do not insist that the actual relations between documents are linear-partitioning. We are merely attempting to generate a good summary by assuming the linear-partition. Our purpose is not to predict the true (linear-partition or unconstrained) assignments of sentences.

Adding the linear-partition constraint has the advantage that we can think of an approximate fast algorithm described in Section 4.3.

4.3 Approximate Algorithm for Summarization with the Linear-Partition Constraint

We introduce an approximate algorithm for solving the p-median problem with the linear-partition constraint. The algorithm iterates the document assignment to medians and the median update, as well-known k-means clustering does. The pseudo-code of the algorithm can be described as follows. In the algorithm, the

Approximate algorithm for p-median on a line
randomly pick initial p medians $d_{m_1}, \cdots, d_{m_p}$
such that $\forall i \leq j,\ t(d_{m_i}) \leq t(d_{m_j})$
while there is any change in medians
  for $l = 1$ **to** p
    reassign $d_{m_l+1}, \cdots, d_{m_{l+1}-1}$ to either d_{m_l} or $d_{m_{l+1}}$
  end for
  for $l = 1$ **to** p
    find the best median out of documents assigned to d_{m_l}
  end for
end while

local reassignment of documents can be performed by first finding

$$h_{max} = \mathrm{argmax}_{h:m_l \leq h < m_{l+1}} \sum_{j=m_l}^{h} e_{m_l j} + \sum_{j=h+1}^{m_{l+1}} e_{m_{l+1} j}$$

and then assigning $d_{h_{max}}$ and the documents on the left side of $d_{h_{max}}$ to d_{m_l}, and those on the right side to $d_{m_{l+1}}$. This local maximization can be performed fast by means of a simple dynamic programming, since the value of the objective function at $h+1$ can be obtained by adding $e_{m_l h+1}$ to and subtracting $e_{m_{l+1} h+1}$ from the value of the objective function at h.

Finding the best median out of documents assigned to d_{m_l} can be performed simply by calculating the objective function for each median candidate.

It is easy to show that each iteration increases the value of the objective function. Therefore, this algorithm at least finds the local maximum.

5 Related Work

There are a number of pieces of work in which integer linear programming is used for text summarization [9,6] or sentence compression [2].

Sharifi et al. [13] attempted to summarize microblogs. Their attempt was, given a query term, to find frequent linguistic patterns that contain the query term. The resulting summary is usually shorter than a single document of microblogs. Hence their objective is completely different from the objective in this paper, which is to generate a summary consisting of multiple documents.

Microblogs are currently being studied extensively[12,11]. O'Connor et al. [10] presented a search application for Twitter. Along with a list of tweets containing the query, this application returns frequent significant terms and tweets containing each term. Swan and Jensen [14] used temporal information to find significant terms, and applied their method on Topic Detection and Tracking (TDT) corpus [1]. Both of their objectives are different from ours.

The Topic Detection and Tracking (TDT) task [1] is related to the present task, in that documents are aligned on timeline in both tasks. Our technique in this paper will be applied to the documents that are found through TDT task. The current task can also be regarded as an instance of clustering data stream [7]. Methods developed in the field of clustering data stream would be promising in the current task and should be explored in the future work.

6 Experiments

6.1 Evaluation of Document Stream Summarization

We present an automatic evaluation measure for this task of short documents summarization on timeline. We assume that reference summaries consisting of time-stamped short documents are available. For the conventional text summarization, ROUGE [5] is very well-known and has been used by many researchers. It calculates the recall indicating how many word types in the reference summary are covered by the generated summary. However, ROUGE will not work as an evaluation measure for our task, because word types that appear at distant time points should be regarded as different word types. In our modified ROUGE, word types of a short document in the reference summary are regarded as covered only if the word type is contained in a selected short document and the difference of the time stamps of the two documents (i.e., one in the reference summary, the other in the generated summary) is within a constant. We set this constant to 10 minutes in our experiments.

We used only content words (i.e., nouns, verbs, adjectives) in the calculation of ROUGE scores. We also removed some Japanese stopwords such as *suru*, *iru*, *naru*, which roughly mean *do*, *be*, *become* respectively.

Table 1. Statistics on tweet data and gold standard data

opponent	fixture	tweet data # of documents (tweets) before filtering	tweet data # of documents (tweets) after filtering	gold standard # of documents	gold standard # of words
Cameroon	June 14, 2010	61666	2814	26	439
Netherlands	June 19, 2010	56976	3219	29	580
Denmark	June 24, 2010	93336	5196	41	690

6.2 Data Preparation

We prepared datasets of document streams consisting of tweets on soccer matches. This dataset consists of Japanese tweets (short documents) mentioning the three matches of Japan in group stage of 2010 FIFA World Cup : Japan vs. Cameroon, Japan vs. Netherlands, Japan vs. Denmark. We used Streaming API provided by Twitter to collect tweets with the relevant hashtags : #soccer, #jfa, #wc2010, #jfa2010, #daihyo, #2010wc. Most users supposedly posted the tweets while they were watching TV broadcast of the matches.

After examining the data, we realized that many of the tweets are not appropriate for summary parts, since they are often simply shouts, just names of Japanese players, or some text fragments that do not make sense by themselves. We thus extracted the tweets that contain both names of players and football terms[4]. The numbers of tweets before and after this filtering for each match are shown in Table 1. Although the filtering greatly reduces the number of documents in Table 1, selecting a few documents from those is still a hard problem. For the matches against Cameroon, Netherlands and Denmark, there are several thousand documents left after filtering, in which case the exact solution of the optimization problem for our model is unobtainable. In order to make the exact solution obtainable for the purpose of comparative experiments, we also created smaller datasets by random sampling (details in Section 6.3). We also created the gold standard data consisting of reference summaries for evaluation purpose. Each document in the gold standard data is also very short, usually consisting of 1 or 2 sentences. The statistics of the gold standard data is also shown in Table 1. We used the text reports on the internet [5] as reference summaries.

Word segmentation and part-of-speech tagging were performed on all the documents including tweet data and gold standard data. We used MeCab[6].

6.3 Experimental Setting

In the calculation of e_{ij} and e_{ij}^{time}, each sentence is represented as a set of content words. We use base forms of content words (nouns, verbs, and adjectives) that

[4] The lists of player names and football terms were manually created with the help of the webpage of Japan national football team (`http://samuraiblue.jp/`) operated by Japan Football Association.

[5] `http://mainichi.jp/enta/sports/soccer/10fwc/graph/2010061402,2010061901,2010062403`

[6] `http://mecab.sourceforge.net/`

are not stopwords. The maximum summary lengths were set to be the same as the lengths of the reference summaries : 26, 29 and 41 documents respectively for the matches against Cameroon, Netherlands and Denmark.

The comparative experiment on the smaller datasets. The first experiment is conducted on the smaller datasets each containing randomly sampled 500 documents[7]. This random sampling was performed 10 times for each setting. The result is the average of these 10 trials. Since this dataset is small, we were able to the branch-and-bound method implemented in ILOG CPLEX version 11.1 to obtain the exact solution to the p-median problem. We compared the exact solutions with the approximate solutions. The length-penalty factor was not employed in this experiment.

The experiment on the larger datasets. The second experiment is conducted on the larger datasets of 2010 FIFA World Cup, each of which consists of 2814, 3219 and 5196 documents, respectively in Table 1. Since it is practically impossible to obtain the exact solution of the p-median problem of this size, we use only the approximate algorithm (Section 4.3). The algorithm is applicable only to the summarization model with the linear-partition constraint. We test both the model with the length-penalty factor and the model without it. For this experiment, we compare the proposed method with the following baselines.

- *random*: this baseline method simply selects p documents randomly. Since the result depends on the generated random values, we executed this method 100 times to obtain 100 summaries and computed the average value of the ROUGE scores of those summaries. Although this method is governed by randomness, if there are similar documents in the data, this method is likely to select one out of those documents.
- *equal*: this baseline method sorts the documents in the order of their created times, and selects p documents so that the intervals of the selected documents are going to be equal. The intervals are measured by the number of documents, not by time.
- *interval*: this baseline method first splits the timeline into intervals of equal length (10 seconds, in our experiments) and selects p intervals that have the largest number of documents. For each of these intervals, the method selects the document with the largest cosine similarity (calculated by bag-of-words vectors) to the union of the documents in the interval.

6.4 Results

We first report the result of the experiment on the smaller datasets each consisting of 800 documents. We test both the branch-and-bound method that yields the exact solution with or without the linear-partition constraint, and the approximate algorithm. The result is summarized in Table 2. We can see that the

[7] For 600 or more documents, the branch-and-bound method did not converge after 4 hours for some settings.

Table 2. ROUGE scores of p-median summarization model on the smaller datasets

		p-median					
		unconstrained				linear-partition	
		exact				*exact*	*approx.*
		e_{ij}	e_{ij}^{time}			e_{ij}	e_{ij}
vs.	summary length	–	$\beta = 300$	$\beta = 600$	$\beta = 900$	–	–
Cameroon	26	0.222	0.247	0.253	0.254	0.218	0.232
Netherlands	29	0.232	0.282	0.279	0.279	0.241	0.251
Denmark	41	0.265	0.300	0.305	0.299	0.296	0.274
average	32.0	0.240	0.276	0.279	0.277	0.252	0.252

Table 3. ROUGE scores of p-median summarization model with the linear-partition document assignment and the baseline methods on the larger datasets of Japan's group stage of 2010 FIFA World Cup. The approximate algorithm was used for solving p-median problem. 'w/o pnlty' means the proposed model without the length-penalty factor, while 'w/ pnlty' means the one with the length-penalty factor.

		baselines			p-median	
					w/o pnlty	w/ pnlty
vs.	summary length	*random*	*equal*	*interval*	*approx.*	
Cameroon	26	0.145	0.173	0.178	0.236	0.200
Netherlands	29	0.156	0.176	0.156	0.309	0.225
Denmark	41	0.188	0.214	0.245	0.314	0.270
average	32.0	0.163	0.188	0.193	0.286	0.232

modification of coefficient improves summarization performance for each match. The ROUGE score of e_{ij}^{time} is stable in the range of $\beta = 300$ to 900. The linear-partition constraint also improves summarization performance without regard to the choice of the algorithms (exact or approximate). This result shows that both of the proposed models (i.e., the modification of coefficients, the linear-partition constraint) work well in the summarization of document stream on timeline.

The average computational times were 44.07 seconds for the exact algorithm with the unconstrained assignment, and 1341.79 seconds for the exact algorithm with the linear-partition constraint, while that of the approximate algorithm was less than a second.

We next report the result of the experiment on the larger datasets of 2010 FIFA World Cup, each of which consists of 2814, 3219 and 5196 documents, respectively in Table 1. The approximate algorithm (Section 4.3), imposing the linear-partition constraint, was used for solving p-median problem. The result is summarized in Table 3. As the table shows, the p-median summarization model with the linear-partition constraint outperformed the three baselines in terms of the ROUGE score for each of the three matches. Also when we impose the length-penalty, the proposed model is still superior to the baselines. We note that the summaries generated by the proposed model with the length-penalty was shorter than those by the baselines on average.

Table 4. An example of summary (Japan vs. Denmark). Tweets are originally in Japanese, and translated into English by the authors. We set the length limit to 10 tweets due to space limitation, and the length-penalty was imposed. Elapsed time from kick-off includes half time.

elapsed time	selected tweets
12m34s	Endo got the yellow card!!!
18m26s	Honda's beautiful freekick!
30m41s	Another freekick goal! This time, by Endo.
54m48s	The 1st half was over. We are 2 points ahead.
91m48s	Okazaki came on as sub of Matsui.
99m28s	Goalie Kawashima made a great save on penalty, but conceded a goal right after that.
116m43s	Okazaki, GOAL! Honda, good pass! Japan is ahead, 3-1.

As an example, we show the summary generated by the proposed method with the length-penalty. Due to the space limitation, we pick the match between Japan and Denmark and set the length limit to 10 tweets.

7 Conclusion

We introduced the task of summarizing document stream of microblogs such as Twitter. Through data analysis, we confirmed that temporally distant documents may refer to distinct events even if they are similar in terms of the words used in those documents, and also that users sometimes post documents with some delay in time. We proposed a summarization model that takes these characteristics of microblogs into account. We also proposed a fast approximate algorithm for generating a summary out of the proposed model. Through experiments on microblog data on soccer matches, we showed that the proposed model improves the quality of summaries.

As future work, we will have to conduct more detailed evaluation on the proposed method including experiments on diverse datasets and manual evaluation of the generated summaries. The method would become more sophisticated with the exploration of other inter-documental coefficients. We would also like to apply our method for other types of topics such as TV dramas or Ustream broadcasting. We filtered tweets using players' names and football terms. Those keywords have to be automatically acquired when this method is applied to many other domains. We are working on keyword extraction from web contents such as Wikipedia, from which Japanese players' names are also avilable[8].

Another interesting direction is the evaluative summarization of document streams of microblogs. Currently, we focus on factual summarization. However, microblog users often express opinions or emotions on events. The modification to the inter-documental coefficients that gives larger weights on evaluative documents would make an evaluative summary of document streams.

[8] `http://en.wikipedia.org/wiki/Japan_national_football_team`

References

1. Allan, J., Carbonell, J., Doddington, G., Yamron, J., Yang, Y., Amherst, U., Umass, J.A.: Topic detection and tracking pilot study. In: DARPA Broadcast News Transcription and Understanding Workshop, pp. 194–218 (1998)
2. Clarke, J., Lapata, M.: Global inference for sentence compression: An integer linear programming approach. J. of Artificial Intelligence Research 31, 399–429 (2008)
3. Drezner, Z., Hamacher, H.W. (eds.): Facility Location: Applications and Theory. Springer, Heidelberg (2004)
4. Hromkovič, J.: Algorithmics for Hard Problems. Springer, Heidelberg (2003)
5. Lin, C.: ROUGE: a package for automatic evaluation of summaries. In: Proceedings of the Workshop on Text Summarization Branches Out, pp. 74–81 (2004)
6. Lin, H., Bilmes, J.: Multi-document summarization via budgeted maximization of submodular functions. In: HLT-NAACL, pp. 912–920 (2010)
7. Mahdiraji, A.R.: Clustering data stream: A survey of algorithms. International Journal of Knowledge-based and Intelligent Engineering Systems 13, 39–44 (2009)
8. Mani, I.: Automatic Summarization. John Benjamins Publisher, Amsterdam (2001)
9. McDonald, R.: A study of global inference algorithms in multi-document summarization. In: Amati, G., Carpineto, C., Romano, G. (eds.) ECIR 2007. LNCS, vol. 4425, pp. 557–564. Springer, Heidelberg (2007)
10. O'Connory, B., Krieger, M., Ahn, D.: Tweetmotif: Exploratory search and topic summarization for twitter. In: AAAI Conf. on Weblogs and Social Media, pp. 384–385 (2010)
11. Petrovic, S., Osborne, M., Lavrenko, V.: Streaming first story detection with application to twitter. In: NAACL, pp. 181–189 (2010)
12. Ritter, A., Cherry, C., Dolan, B.: Unsupervised modeling of twitter conversations. In: NAACL, pp. 172–180 (2010)
13. Sharifi, B., Hutton, M.A., Kalita, J.: Summarizing microblogs automatically. In: NAACL, short paper, pp. 685–688 (2010)
14. Swan, R., Jensen, D.: Timemines: Constructing timelines with statistical models of word usage. In: SIGKDD Workshop on Text Mining, pp. 73–80 (2000)
15. Takamura, H., Okumura, M.: Text summarization model based on the budgeted median problem. In: CIKM, short paper, pp. 1589–1592 (2009)