A Multi-level System for Sequential Update Summarization

Chunyun Zhang†
School of Computer Science and Technology
Shandong University of Finance and Economics
Jinan, Shandong
zhangchunyun1009@126.com

Zhanyu Ma*, Jiayue Zhang, Weiran Xu, Jun Guo
School of Information and communication engineering
Beijing University of Posts and Telecommunications
Beijing, China
mazhanyu@bupt.edu.cn,xuweiran@bupt.edu.cn
zhangjiayue@bupt.edu.cn, guojun@bupt.edu.cn

***Abstract*—When an emergency occurs, such as the outbreak of natural disaster, the news about the incident showed a trend of blowout. Mostly, these news are reproduced by people and spread out with duplicate, unimportant or wrong information. Hence, it is necessary and vital to provide individuals with timely and important information of these incidents during their development. In this paper, we present a new multi-level system which can broadcasts with useful, new, and timely sentence-length updates about a developing event. The new system proposed a novel method, which incorporates techniques from topic-level and sentence-level summarization. To evaluate the performance of the proposed system, we applied it to the task of sequential update summarization of Temporal Summarization (TS) track at Text Retrieval Conference (TREC). Experimental results showed that our proposed method have a good performance.**

I. Introduction

A time-critical news event refers to an unexpected news event, such as natural disaster (e.g. hurricane) and human accidents (e.g. air crasher), whose information about the topic is rapidly developing [1]. The news of the event is widely spread through multi-level news channels around the world. However, because of the diversity of journalistic sources, details reported about the event are redundant, dynamic, and sometimes mistaken. Especially during major events, which involve extensive damage to life or crippling of infrastructure, it is harder to collect authoritative news, causing rumors and unsubstantiated information to propagate [2]. Meanwhile the sudden events are also very important topics to individuals. People want get timely information, especially for these people who is relative to these sudden events, they even cannot afford waiting comprehensive reports to materialize [3].

Unfortunately, existing solutions cannot satisfy people's demands in getting sequential update summarizations about these events. Because this solution on sequential update summarization has its roots in text summarization, topic detection and tracking, and time-based summarization techniques. However, most current summarization systems can either use static summarization methods [4], [5], [6], [7], which only provide sentences extracted with particular properties based on traditional techniques of Natural Language Processing (NLP) [8], or use Information Retrieval (IR) [9] and topic detection and tracking (TDT) methods [10], [11], which only provide topic-level summaries. In most ways, the sequential update summarization is an event- and sentence-level analogue of "first topic detection" problem [9]. In all, there is no support for only presenting peoples with novel content (i.e. updates to the user) and updates can suffer from poor coverage, especially for smaller events, and unreliable information.

In this paper, we present a new multi-level summarization system, which focus on extracting sequential update summarization on sudden events. The system incorporates the technologies of topic-level and sentence-level summarization and tries to broadcast with useful, new, and timely sentence-length updates about a developing event. To evaluate the effectiveness of our new methodology, we applied it to the Sequential Update Summarization (SUS) task of Temporal Summarization (TS) track [12] at Text Retrieval Conference (TREC) [13]. By using the evaluation metric of SUS task, we computed the expected gain, expected latency gain, comprehensiveness and latency comprehensiveness on our extracted updates of 10 sudden events. Experimental results showed that our proposed method have a good performance.

In the remainder of this study, we first review some related work on summarization and topic threading in Section II. In Section III we formalize the problem. Section IV presents the novel multi-level summarization system on sequential update summarization. We conduct experimental results to verify the effectiveness of our proposed method in Section V and we conclude in Section VI.

II. Related Work

The problem of sequential update summarization has its roots in text summarization [14], [5], [15], topic detection and tracking [16], and time-based summarization techniques [10], [9].

Our work is much similar to extractive summarization where the summary consists of sentences extracted from the pool of relative documents about an event (i.e. multi-document summarization). The core technique of this research is to extract a sequential sentences which have high score of importance. G. Erkan et al. proposed a method which computed sentence importance based on the eigenvector of a graph representation of sentences [17]. Other methods [4], [14], [18] computed the importance of sentences by looking for cue words and phrases, and consider even more focused features such as sentence length and case of words.

†The author finished this work in Beijing University of Posts and Telecommunications.
*Corresponding author: Zhanyu Ma.

QSHINE 2015, August 19-20, Taipei, Taiwan
Copyright © 2015 ICST
DOI 10.4108/eai.19-8-2015.2260848

```
<event>
  <id>1</id>
  <start>1329910380</start>
  <end>1330774380</end>
  <query>buenos aires train crash</query>
  <type>accident</type>
  <locations/>
  <deaths/>
  <injuries/>
</event>
```

Fig. 1. A sudden event definition for "2012 Buenos Aires Rail Disaster" in the SUS task.

Topic detection and tracking (TDT) refers to the document-level tasks associated with detecting and tracking news events [16]. Authors of [9] suggested to retrospectively select novel and relevant sentences from a stream of news articles. However, the TDT is more topic-based than sentence-based. In most ways, the sequential update summarization is an event and sentence-level analogue of TDT's "first topic detection" problem [9].

Referring the time-based summarization as the task of temporal summarization, most these systems focuses on temporal expression extraction from text normalizing references to dates, times, and elapsed times [10]. The system in [19] generated the meaningful temporal summarization of event-related updates and automatically annotates the identified events in a timeline. Methods proposed in [20] retrieved sequential versions of a single web page during predefined time intervals. Paper [21] presented a framework that extracts events relevant to a query from a collection of documents, and places these events along a timeline.

III. Problem Definition

A sudden event, *e*, is a temporally acute topic with a clear onset time, $[t_s, t_e]$. An event query, Q_e, a representation of the event description expressed by a user during the event. The set of keywords associated with the event, $\mathcal{K}(e)$, represents the important information that should be included in the updates to deliver to the users (e.g. The location where the event happened, the death number caused by this event.). The system observes a temporally-ordered stream of documents, $[d_1, d_2, \cdots]$. On the observation of d_t, the system makes a decision to emit zero or more updates. The pool of candidate updates consists of sentences in documents which comprised of the most recent *k* documents in the event timeframe. A sudden event in the SUS task of temporal summarization track at TREC is represented as Figure 1. The sequential update summarization should be simulated as Algorithm 1.

IV. Multi-level Sequential Update Summarization System

Base on the SUS task of the TS track, we constructed a multi-level sequential update summarization system in this paper. The framework of the system is illustrated in Figure 2. The framework contains three main modules: preprocess and information retrieval module, keywords mining module and sentence scoring module.

Algorithm 1 Sequential Update Summarization System.

Require:
SequentialUpdateSummarization $\{S, C, q, t_s, t_s\}$:
S=the SUS system;
C=time-ordered corpus;
Q_e=keyword query of a sudden event;
t_s=start time of a sudden event;
t_e=end time of a sudden event;

Ensure: updates set U

```
U ← {};
S : Retrieval(q)
for d ∈ C do
    do
    S : Process(d);
    t ← d.Time();
    if t ∈ {t_s, t_e} then
        then
        U_t ← S.Decide;
        for u ∈ U_t do
            do
            U.Append(u; t);
        end for
    end if
end for
```

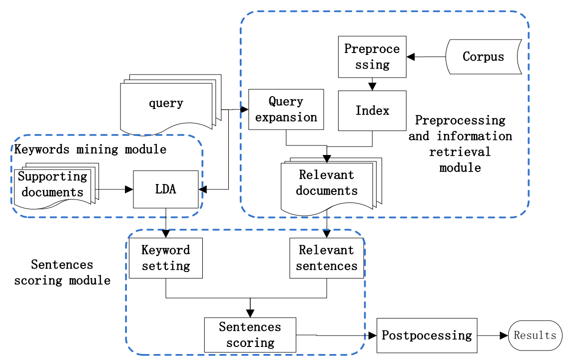

Fig. 2. The framework of the multi-level sequential update summarization system

A. Preprocessing and Information Retrieval Module

Because the original corpus of the SUS task is needed to be preprocessed, the overall general process of this module is described as follows:

- Decrypt File. The first step is to decrypt the files using the authorized key from authority. This step converts the GPG file format to SC file format.
- Deserialization. We use stream corpus toolbox to parse these SC files to TXT files. The organization of TREC provided the stream corpus toolbox to parse these SC files. The stream corpus toolbox gives a common data interchange format for document processing pipelines which apply language processing tools to large streams of text.
- Build Index. This step is to build index by Indri [22] for information retrieval.

- Information Retrieval. The last step is to use Indri as a tool for information retrieval. This step enable users to submit the queries and obtain most relevant documents.

B. Keywords Mining Module

In this module, we utilize hierarchical Latent Dirichlet Allocation to find potential topics and returns the most representative words of each topic as keywords.

Latent Dirichlet Allocation (LDA) [23] is a statistical model, specically a topic model, which can be used to identify hidden topic from a large document collection corpus. The basic idea of LDA is that a document can be considered as a mixture of a limited number of topics and each meaningful word in the document can be associated with one of these topics. Given a corpus of documents, LDA attempts to identify a set of topics, associate a set of words with a topic, and dene a specic mixture of these topics for each document in the corpus.

In this paper, for each event in each hour, we firstly retrieve the most 500 relevant documents, and then extract keywords by LDA in current hour. In this module, we use the GibbsLDA++ tool [24] to extract keywords. We firstly use the LDA toolkit to discover two topics and choose the most representative words for each topic; secondly, discover 5 new topics by the same method under the topic discovered in the last step and choose the most representative words of each topic; lastly, integrate the two level representative words of each topic to form keywords set $\mathcal{K}(e)$.

C. Sentences Scoring Module

We utilize four sentence scoring method in this module: KLP method, SKD method, and KS method.

The first method assumes an update is a long sentence which shoots much more keywords and should be placed on the first place in a paragraph. Hence, it consider three important factors: the keywords diversity, the length of a sentence, and the position of the sentence, which we named as KLP method. The scoring metric is as following:

$$\begin{aligned} Score(s_i) = \alpha \frac{\sum\limits_{w \in s_i} tf(w) \cdot idf(w)}{\max\limits_{s_j \in d}\{\sum\limits_{w \in s_j} tf(w) \cdot idf(w)\}} + \\ \beta \frac{Length_{s_i}}{\max\limits_{s_j}\{length_{s_j}\}} + \gamma position_{s_i} \end{aligned}, \quad (1)$$

where $w \in \mathcal{K}(e)$ is one of keywords of event e extracted in Section IV-B, α, β, γ are weight of the keywords diversity, length and the position of sentences respectively. When compute the $idf(w)$, the documents are referred to relevant documents in the current hour. If a sentence is placed on the first place of a paragraph, $position_{s_i} = 1$, or $position_{s_i} = 0$.

The second method assumes that an update should be a short length sentence with larger keywords diversity. Because a too long sentence is normally a retrospective summary of an event, not an update. We named this metric as SKD, whose scoring metric is as following:

$$Score(s_i) = \frac{1}{N(N+1) \cdot Length} \cdot \sum_{j=1}^{k-1} \frac{Score(w_j) \cdot Score(w_{j+1})}{distance(w_j, w_{j+1})}, \quad (2)$$

where N is the number of keywords included in s_i, the $Score(w)$ is the confidence of keyword w obtained from Section 4.2, the $distance(w_j, w_{j+1})$ is the distance between w_j and w_{j+1}.

The third method is a keyword shooting method, which only considers the diversity of keywords included in the sentence. We named it as KS method. Its scoring metric is as following:

$$Score(s_i) = \frac{|V_{keywords} \cap S_i|}{|S_i|}, \quad (3)$$

where $V_{keywords}$ is the keyword vector of the event e. $s_i$is the ith related sentences of event e.

After getting high confidence sentences, the post-processing module will do the duplicate removal to sentences, which first finds same sentences with different sentences id, then compares the stream id of all sentences and choose the one with the earliest time information as the submission sentence.

V. Experiments

A. Data and Topics

The data used in the SUS task of TS track is provided by Organizer of KBA track [25] at TREC, which is hosted by Amazon Public Dataset service. This corpus [26] consists of a set of time-stamped documents from a variety of news and social media sources covering the time period October 2011 through January 2013, whose time span is 17 months with 11,248 hours. There are more than 1 billion documents, each with absolute time stamp that places it in the stream, which is mainly composed by news, social (blog, forum,), web (e.g., arxiv, linking events) content. All documents contain a set of sentences, each with a unique identifier.

There are 10 events (topics) [27] in the SUS task, each has a single type title, description (URL to Wikipedia entry), begin and end times, query keywords. Types are taken from {accident, shooting, storm, earthquake, bombing} and they have a set of attributes, such as location, death, financial impact and so on. The ten test topics is list in Table I.

B. Results

To evaluate the updates provided by our multi-level SUS system, we used the evaluation metric [28] defined by the authority of temporal summarization track. The evaluation process is mainly Update-Nugget Matching. Nuggets are manually extracted atomic pieces of information relevant, which are perceived as relevant and novel for the editions on Wikipedia articles. The Gold Nugget Extraction defined the space of relevant information for the queries, and the update-nugget matching matched this information to our provided updates to evaluate their accuracy and coverage of the information space.

TABLE I. QUERIES AND TITLES OF 10 TEST TOPICS OF TEMPORAL SUMMARIZATION TRACK [27].

Query of topics	Title of topics
1. Buenos aires train crash	2012 Buenos Aires Rail Disaster
2. Pakistan factory fire	2012 Pakistan garment factory fires
3. Colorado shooting	2012 Aurora shooting
4. Sikh temple shooting	Wisconsin Sikh temple shooting
5. Hurricane isaac	Hurricane Isaac (2012)
6. Hurricane sandy	Hurricane Sandy
7. Midwest derecho	June 2012 North American derecho
8. Typhoon bopha	Typhoon Bopha
9. Guatemala earthquake	2012 Guatemala earthquake
10. Tel aviv bus bombing	2012 Tel Aviv bus bombing

TABLE II. THE μ AND σ OF EXPECTED GAIN AND EXPECTED LATENCY GAIN OVER ALL EVENTS OF THE MULTI-LEVEL SUS SYSTEM. (THE E[GAIN] IS THE EXPECTED GAIN WHICH IS SIMILAR TO TRADITIONAL NOTIONS OF PRECISION IN INFORMATION RETRIEVAL; E [LATENCY GAIN] IS THE TIME-SENSITIVE EXPECTED GAIN.)

Methods	E[Gain]	E[latency gain]
The best reported	*0.149(0.101)*	*0.136 (0.090)*
KLP (0.5,0.3,0.2)	0.065 (0.034)	0.067 (0.026)
KLP (0.5,0.2,0.3)	0.065 (0.034)	0.067 (0.026)
KLP(0.6,0.2,0.2)	0.071 (0.039)	0.074 (0.031)
SKD	0.103 (0.084)	0.103 (0.050)
KS	0.149 (0.101)	0.136 (0.090)

In all, the evaluation metrics will measure the degree to which a system can generate these nuggets in a timely manner.

Based on our three sentence scoring method, we implement the multi-level SUS system and extracted top 60 updates per event sorted by the provided confidence scores (highest first). Table II and Table III illustrate the five results of these three methods. The four parameters are evaluated by comparing generated updates with gold nuggets by using expected gain, expected latency gain, comprehensiveness, and Latency Comprehensiveness metrics. The expected gain is similar to traditional notions of precision in IR. Expected latency gain is a time-sensitive expected gain. Comprehensiveness is similar to recall in IR, which evaluates coverage of gold nuggets. The latency Comp. is the time-sensitive comprehensiveness [28]. The results in italic are the best reported results in the SUS task in 2013 [28].

From Table II, we can see that the KS method has the best expected gain and expected latency gain, which are equal to the best reported results. We can conclude that compared with the KLP and SKD method, the keywords diversity in KS method is the most effective metric to decide the importance of updates.

From Table III, we can see that the KLP method has the best comprehensiveness and latency comp., KS method has the worst comp. and latency comp., while the performance of SKD method is between the KLP method and KS method. That is to say, the KLP method utilized a general metric on scoring updates which can cover much more nuggets.

By comparing the different weights of KLP method from Table II and Table III, we can conclude that the keyword diversity is more important than the effect of sentence length and sentence position in the KLP method.

TABLE III. THE μ AND σ OF COMPREHENSIVENESS AND LATENCY COMPREHENSIVENESS OVER ALL EVENTS OF THE MULTI-LEVEL SUS SYSTEM (COMPREHENSIVENESS IS SIMILAR TO RECALL IN IE, WHICH EVALUATES COVERAGE OF NUGGETS; LATENCY COMP. IS THE TIME-SENSITIVE COMPREHENSIVENESS.).

Methods	Comprehensive	Latency Comp.
The best reported	*0.445 (0.191)*	*0.571 (0.358)*
KLP (0.5,0.3,0.2)	0.224 (0.178)	0.292 (0.270)
KLP (0.5,0.2,0.3)	0.224 (0.178)	0.288 (0.262)
KLP(0.6,0.2,0.2)	0.204 (0.146)	0.260 (0.217)
SKD	0.131 (0.138)	0.176 (0.203)
KS	0.099 (0.099)	0.126 (0.164)

In addition, by combining the results of Table II and Table III, we can conclude that the expected gain has reciprocal relationship with comprehensiveness, like the precision and recall in information retrieval. The KLP method utilizes a more comprehensive metric which consider more factors in scoring sentences, which is threaten to choose long sentences. But it has the worst gain and latency gain. The KS method proposed only the keyword diversity to evaluate sentences,and it has good performance on expected gain and expected latency gain. All these indicate that a good update should be not too long a sentence which includes many keywords.

VI. CONCLUSIONS

This paper presents a multi-level sequential update summarization system, which incorporates techniques from topic-level and sentence-level summarization. The multi-level system focuses attention on the SUS task of Temporal Summarization track at TREC and broadcasts with useful, new, and timely sentence-length updates about a developing sudden event. We applied the multi-level SUS system to extract updates of ten sudden events of the SUS task, and experimental results showed that our proposed system has a good performance.

ACKNOWLEDGMENT

This work is partly supported by National Natural Science Foundation of China (NSFC) grant No. 61402047, 61511130081, and 61273217, Scientific Research Foundation for Returned Scholars, Ministry of Education of China, Sweden STINT initiation grant Dnr. IB2015-5959, the Research Fund for Excellent Young and Middle-aged Scientists of Shandong Province (No. BS2013DX035), Chinese 111 program of Advanced Intelligence and Network Service under grant No. B08004, and EU FP7 IRSESMobileCloud Project (Grant No. 612212).

REFERENCES

[1] Q. Guo, F. Diaz, and E. Yom-Tov, "Updating users about time critical events," in *Advances in Information Retrieval*. Springer, 2013, pp. 483–494.

[2] M. Mendoza, B. Poblete, and C. Castillo, "Twitter under crisis: Can we trust what we rt?" in *Proceedings of the first workshop on social media analytics*. ACM, 2010, pp. 71–79.

[3] E. Yom-Tov and F. Diaz, "Out of sight, not out of mind: on the effect of social and physical detachment on information need," in *Proceedings of the 34th international ACM SIGIR conference on Research and development in Information Retrieval*. ACM, 2011, pp. 385–394.

[4] H. P. Luhn, "The automatic creation of literature abstracts," *IBM Journal of research and development*, vol. 2, no. 2, pp. 159–165, 1958.

[5] J. Goldstein, M. Kantrowitz, V. Mittal, and J. Carbonell, "Summarizing text documents: sentence selection and evaluation metrics," in *Proceedings of the 22nd annual international ACM SIGIR conference on Research and development in information retrieval*. ACM, 1999, pp. 121–128.

[6] D. Wang, S. Zhu, T. Li, and Y. Gong, "Comparative document summarization via discriminative sentence selection," *ACM Transactions on Knowledge Discovery from Data (TKDD)*, vol. 6, no. 3, p. 12, 2012.

[7] J. J. Pollock and A. Zamora, "Automatic abstracting research at chemical abstracts service," *Journal of Chemical Information and Computer Sciences*, vol. 15, no. 4, pp. 226–232, 1975.

[8] E. Hovy and C.-Y. Lin, "Automated text summarization and the summarist system," in *Proceedings of a workshop on held at Baltimore, Maryland: October 13-15, 1998*. Association for Computational Linguistics, 1998, pp. 197–214.

[9] J. Allan, R. Gupta, and V. Khandelwal, "Temporal summaries of new topics," in *Proceedings of the 24th annual international ACM SIGIR conference on Research and development in information retrieval*. ACM, 2001, pp. 10–18.

[10] I. Mani and G. Wilson, "Robust temporal processing of news," in *Proceedings of the 38th Annual Meeting on Association for Computational Linguistics*. Association for Computational Linguistics, 2000, pp. 69–76.

[11] R. Swan and J. Allan, "Automatic generation of overview timelines," in *Proceedings of the 23rd annual international ACM SIGIR conference on Research and development in information retrieval*. ACM, 2000, pp. 49–56.

[12] "Temporal summarization," http://www.trec-ts.org/documents/, 2013.

[13] "Trec," http://trec.nist.gov/, 2013.

[14] H. P. Edmundson, "New methods in automatic extracting," *Journal of the ACM (JACM)*, vol. 16, no. 2, pp. 264–285, 1969.

[15] J. Goldstein, V. Mittal, J. Carbonell, and M. Kantrowitz, "Multi-document summarization by sentence extraction," in *Proceedings of the 2000 NAACL-ANLPWorkshop on Automatic summarization-Volume 4*. Association for Computational Linguistics, 2000, pp. 40–48.

[16] J. Allan, "Introduction to topic detection and tracking," in *Topic detection and tracking*. Springer, 2002, pp. 1–16.

[17] G. Erkan and D. R. Radev, "Lexrank: graph-based lexical centrality as salience in text summarization," *Journal of Artificial Intelligence Research*, pp. 457–479, 2004.

[18] J. Kupiec, J. Pedersen, and F. Chen, "A trainable document summarizer," in *Proceedings of the 18th annual international ACM SIGIR conference on Research and development in information retrieval*. ACM, 1995, pp. 68–73.

[19] M. Georgescu, D. D. Pham, N. Kanhabua, S. Zerr, S. Siersdorfer, and W. Nejdl, "Temporal summarization of event-related updates in wikipedia," in *Proceedings of the 22nd international conference on World Wide Web companion*. International World Wide Web Conferences Steering Committee, 2013, pp. 281–284.

[20] A. Jatowt and M. Ishizuka, "Temporal web page summarization," in *Web Information Systems–WISE 2004*. Springer, 2004, pp. 303–312.

[21] H. L. Chieu and Y. K. Lee, "Query based event extraction along a timeline," in *Proceedings of the 27th annual international ACM SIGIR conference on Research and development in information retrieval*. ACM, 2004, pp. 425–432.

[22] "Indri," http://www.lemurproject.org/indri.php.

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

[24] "Gibbs lda ++," http://sourceforge.net/projects/gibbslda/.

[25] "Knowledge based acceleration," http://trec-kba.org/, 2013.

[26] "Kba data," http://s3.amazonaws.com/aws-publicdatasets/trec/kba/index.html, 2013.

[27] "Test topics," http://trec.nist.gov/data/tempsumm/2013/testTopics.xml, 2013.

[28] J. Aslam, M. Ekstrand-Abueg, V. Pavlu, F. Diaz, and T. Sakai, "Trec 2013 temporal summarization," in *Proceedings of the 22nd Text Retrieval Conference (TREC), November*, 2013.