Multi-document Summarization Based on Atomic Semantic Events and Their Temporal Relationships

Yllias Chali(✉) and Mohsin Uddin

University of Lethbridge, Lethbridge, AB, Canada
chali@cs.uleth.ca, mdmohsin.uddin@uleth.ca

Abstract. Automatic multi-document summarization (MDS) is the process of extracting the most important information, such as events and entities, from multiple natural language texts focused on the same topic. In this paper, we experiment with the effects of different groups of information such as events and named entities in the domain of generic and update MDS. Our generic MDS system has outperformed the best recent generic MDS systems in DUC 2004 in terms of ROUGE-1 recall and f_1-measure. Update summarization is a new form of MDS, where novel yet salient sentences are chosen as summary sentences based on the assumption that the user has already read a given set of documents. We present an event based update summarization where the novelty is detected based on the temporal ordering of events, and the saliency is ensured by the event and entity distribution. To our knowledge, no other study has deeply experimented with the effects of the novelty information acquired from the temporal ordering of events (assuming that a sentence contains one or more events) in the domain of update multi-document summarization. Our update MDS system has outperformed the state-of-the-art update MDS system in terms of ROUGE-2 and ROUGE-SU4 recall measures. All our MDS systems also generate quality summaries which are manually evaluated based on popular evaluation criteria.

1 Introduction

Automatic multi-document summarization (MDS) extracts core information from the source text and presents the most important content to the user in a concise form [24]. The important information is contained in textual units or groups of textual units which should be taken into consideration in generating a coherent and salient summary. In this paper, we propose an event based model of the generic MDS where we represent the generic summarization problem as an atomic event extraction as well as a topic distribution problem. Another new type of summarization called update MDS, whose goal is to get a salient summary of the updated documents supposing that the user has read the earlier documents about the same topic. The best of the recent efforts to generate update summary use graph based algorithms with some additional features to explore the novelty of the document [9,20,36]. Maximal Marginal Relevance (MMR) based

N. Ferro et al. (Eds.): ECIR 2016, LNCS 9626, pp. 366–377, 2016.
DOI: 10.1007/978-3-319-30671-1_27

approach [3] is used to blindly filter out the new information. These approaches discard the sentences containing novel information if they contain some old information from the previous document sets [7].

Steinberger et al. [33] use the sentence time information in the Latent Semantic Analysis (LSA) framework to get the novel sentences. They only consider the first time expression as the anchored time of the sentence, but sentences may contain multiple time expressions from various chronologies. For instance, consider the sentence *"Two members of Basque separatist group ETA arrested while transporting half a tonne of explosives to Madrid just prior to the* ***March 2004*** *bombings received jail sentences of 22 years each on* ***Monday****"*. Here we get two[1] time expressions: *March 2004* and *Monday.* The first expression represents the very old information, and the second one represents the accurate anchoring of the sentence. If we consider the first time expression as the sentence's time, like Steinberger et al. [33] would, then it would give us false novel/update information. This is why we take into account all of the events of a sentence to calculate its anchored time. In this paper, we also design a novel approach by taking into account all of the events in a sentence and their temporal relations to ensure the novelty, as well as the saliency, in update summarization. We represent the novelty detection problem as a chronological ordering problem of the temporal events and time expressions. Our event based sentence ranking system uses a topic model to identify all of the salient sentences. The rest of the paper is organized as follows. Section 2 reviews previous related works in text summarization. Section 3 describes our proposed summarization models. Section 4 gives the evaluation of our systems. Section 5 presents some conclusions and future works.

2 Related Works

Every document covers a central theme or event. There are other sub-events which support the central event. There are also many words or terms across the whole document, which can act as an individual event, they contribute to the main theme. Named entities such as time, date, person, money, organizations, locations, etc., are also significant because they build up the document structure. Although events and named entities are terms or group of terms, they have a higher significance than the normal words or terms. Those events and named entities can help to generate high performing summaries. Filatova and Hatzivassiloglou [11] used atomic events in extractive summarization. They considered events as a triplet of two named entities and verb (or action noun), where the verb (or action noun) connects the two named entities. Several greedy algorithms based on the co-occurrence statistics of events are used to generate a summary. They showed that event-based summaries get a much better score than the summaries generated by *tf*idf* weighing of words. Li et al. [19] also defined the same complex structure as an event and the PageRank algorithm [29] is applied to

[1] Here **'22 years'** is a time period. Time periods do not carry important information for detecting novelty.

estimate the event relevance in a summary generation. Another recent summarization work based on the event semantics is done by Zhang et al. [37]. Their events may contain an unlimited number of entities. Due to the complex nature of all of the previous authors' defined events, it is hard to use their defined event concept in a topic model to get the semantic event distribution in text.

Our defined semantic event is an atomic term, which is similar to the TimeML [31]. Pustejovsky et al. [31] consider events as a cover term for situations that happen, occur, hold, or take place. Event spans can be a period of time. Aspect, intentional state, intentional action, perception, occurrence, and modal can be events. We consider some classes of event expressions such as verbs (e.g., launched, cultivated, resigned, won) and event nominals (e.g., Vietnam War, Military operation).

Events like deverbal nouns are used in G-FLOW [6] to identify discourse relations to ensure coherency in a summary. Our generic MDS system uses the event and entity distribution, obtained from a topic model in sentence ranking, to generate a quality summary.

Update summarization, the newest type of challenge for summarization communities, is introduced first in DUC'2007[2]. Several popular generic summarization approaches, such as LexRank [10], TextRank [26] were used in update summarization without paying attention on the novelty detection. Fisher and Roark [12] used a domain-independent supervised classification to rank sentences and then they extract all of the sentences containing old information by using some filtering rules. QCQPSum [21] involved the previous documents in an objective function formulation and a reinforcement propagation in the new documents. It did not try to extract the novel information at the semantic level. We can see a few semantic analysis based novelty detection approaches: the Iterative Residual Rescaling (IRR) based LSA framework [32] and the Bayesian multinomial probability distribution based approach [7]. The state-of-the-art update summarization system, h-uHDP model [16] used Hierarchical Dirichlet Process (HDP) [35] to get the history epoch and the update epoch distribution. They used Kullback-Leibler (KL) [15] divergence based greedy approach to select novel sentences. All of the above approaches neglected the semantic temporal information which is crucial in novelty detection.

3 Our Methodologies

3.1 Pre-processing of the Data Set

In this paper, we use Stanford CoreNLP[3] for tokenization, named entity recognition, and cross-document coreference resolution. We remove all of the candidate sentences containing quotations. We also remove the candidate sentences whose length are less than 11 words. Sentences containing quotations are not appropriate for summary and shorter sentences carry a small amount of relative

[2] http://duc.nist.gov/duc2007/tasks.html.
[3] http://nlp.stanford.edu/software/corenlp.shtml.

information [17]. After tokenization, we remove stop words. We use Porter Stemmer [30] for stemming. Stemmed words or terms are then fed to Latent Dirichlet Allocation (LDA) engine for further processing. We use ClearTK[4] system [2] for event and temporal relation extraction.

3.2 Generic Summarization

The generic summarization problem is formulated as follows. Any cluster c contains n documents and all of the documents are equiprobable. All of the documents in each cluster are sorted in the descending order of their creation time[5]. The topic probability of each topic T_j can be calculated by Eq. (1) where $j \in \{1, \ldots, K\}$ and K is the number of topics of Latent Dirichlet Allocation (LDA) Model.

$$P(T_j) = \sum_{d=1}^{n} P(T_j|D_d)P(D_d) \tag{1}$$

To increase the coherence of the summary, we calculate sentence position score, S_p. If SC_d is the number of sentences in Document D, S_p can be calculated by Eq. (2) where sentence position index, $i \in \{0, \ldots, SC_d - 1\}$.

$$S_p = 1 - \frac{i}{SC_d} \tag{2}$$

The score of a sentence can be computed by Eq. (3).

$$Score(S) = S_p \times \sum_{t \in S} (P(t) \times W_g) \tag{3}$$

In Eq. (3), W_g is the specific weight factor for each group of terms. TC_g is the number of terms in one group g where $g \in \{event(e), named\text{-}entity(n), other(o)\}$. We consider empirically W_g is 1 for the group called *other* (which is a set of normal terms other than events and named entities) and W_g for groups event and named entity can be calculated by Eq. (5).

$$M = max_g TC_g, g \in \{e, n, o\} \tag{4}$$

Here, M is the number of terms in the highest group.

$$W_g = \frac{M}{TC_g} \tag{5}$$

Our weight calculating scheme ensures larger weights for event and named entity groups and also prevents the high occurring group from scoring high. The steps of our generic MDS system are as follows:

---

[4] http://code.google.com/p/cleartk/.

[5] Document Creation Time (DCT) can be calculated from document name.

1. Apply the LDA topic model on the corpus of documents for a fixed number[6] of topics K.
2. Compute the probability of topic T_j by Eq. (1) and sort the topics in the descending order of their probabilities.
3. Pick the topic T_j from the sorted list in the order of the probabilities of T_j, i.e., $P(T_1), .., P(T_k)$.
4. For topic T_j, compute the score of all of the sentences by Eq. (3) where $P(t)$ is the unigram probability distribution obtained from the LDA topic model.
5. For topic T_j, pick up the sentence with the highest score and include it in the summary. If it is already included in the summary or it dissatisfies other requirements (cosine score between candidate sentence and already-included summary sentences crosses the certain range), then pick up the sentence with the next highest score for this topic T_j.
6. Each selected sentence is compressed according to the method described in Sect. 3.3.
7. If the summary reaches its desired length then terminate the operation, else continue from step 3.

3.3 Sentence Compression

The quality of a summary can be improved by sentence compression [13,18]. Consider the sentence *"The Amish school where a gunman shot 10 girls last week, killing five of them, is expected to be demolished on Thursday, a fire department official said"*. Here we can see the subclause "a fire department official said" does not have any significance in a summary. Removing this type of long unnecessary subclauses will improve summary quality and provide extra space to include new information in a fixed length summary. We mainly consider widely used reporting verbs such as *said*, *told*, *reported* etc., to find out subclauses like in the above example. In our experiments, we use the Stanford dependency parser [4] to parse each selected sentence. Sentences containing a reporting verb are always parsed following a fixed rule where the reporting verb is always the 'root' of the dependency tree. Then we traverse the parse tree to find the subclause related to that reporting verb.

3.4 Update Summarization

Time End Point Normalization. Time expression identification and normalization are integral parts for the temporal processing of raw text. We use Stanford SUTime [5], which is a rule-based temporal tagger, to extract all of the temporal expressions. SUTime is one of the best systems in capturing temporal expressions from a natural language text. It follows TimeML [31] formats (TIMEX3) for normalizing time expressions.

Consider the sentence: *"The Amish school where a gunman shot 10 girls* ***last week***, *killing five of them, is expected to be demolished* ***Thursday***, *a fire*

[6] Total 4 topics are taken into account, i.e. K is 4.

department official said". Here *last week* and *Thursday* are the time expressions of the sentence. SUTime output of the above text is mentioned below, where October 11^{th}, 2006 is a reference date:

"The Amish school where a gunman shot 10 girls <TIMEX3 tid="t2" type="DATE" ***value="2006-W40"*** *> last week </TIMEX3>, killing five of them, is expected to be demolished <TIMEX3 tid="t3" type="DATE"* ***value= "2006-10-12"*** *>Thursday </TIMEX3 >, a fire department official said."*

SUTime extracts *2006-W40* and *2006-10-12* as the normalized date of *last week* and *Thursday*, respectively. We convert them into an absolute time end point on a universal timeline. We follow standard date and time format (*YYYY-MM-DD hh:mm:ss*) for the time end point. For example, after conversion of **2006-W40** and **2006-10-12**, we get 2006-09-23 23:59:59 and 2006-10-12 23:59:59, respectively.

Temporal Ordering of Events and Time Expressions. In update summarization, knowing the relative order of the events is very useful for merging and presenting information from various news sources [25]. Information, such as event occurrence time or what events occurred prior to a particular event, presuppose the ability to infer an event's temporal ordering in discourse [25]. Inferring relations of temporal entities and events is a crucial step towards update summarization task.

Unlike Denis and Muller [8], we anchor events to one time point only, which is the upper end point. We are concerned only about the relative ordering of the events. We use ClearTK-TimeML tool to extract events and temporal relations [2]. In ClearTK-TimeML, four types of temporal relations are predicted. They are **B**EFORE, **A**FTER, **I**NCLUDES, and **N**ORELATION. Our main goal is to solve the novelty problem by using relative events' anchored values. In order to saturate the event-event and event-time relations, we use Allen's [1] transitive closure rules. Some of them are given below:

$$A \textit{ before } B \textit{ and } B \textit{ before } C \implies A \textit{ before } C$$
$$A \textit{ includes } B \textit{ and } B \textit{ includes } C \implies A \textit{ includes } C$$
$$A \textit{ after } B \textit{ and } B \textit{ after } C \implies A \textit{ after } C$$

We anchor all of the events to absolute times based on the *'includes'* and *'is-included'* relations of the event-time links. The remaining events are anchored approximately, based on other relations which are *'before'* and *'after'*.

Temporal Score. To obtain temporal score, we use ClearTK system [2] for initial temporal relation extraction and some transitive rules as described earlier. First, we relax the original event time association problem by anchoring the event to an approximate time. Then, we calculate the temporal score of a sentence by taking an average time score of all of the events' anchored time. Then, all of the sentences are ordered in the descending order of their temporal scores except for the first sentence of each document. Then, we calculate the temporal position score (tp_s) of the temporally ordered sentences. The tp_s of the first sentence of a document is considered to be one. Temporal position scores of the remaining sentences can be calculated by the Eq. (6). D_s is the number of sentences in document D and the temporally ordered sentence position index, $i \in \{0, \ldots, D_s - 1\}$.

$$tp_s = 1 - \frac{\gamma \times i}{D_s} \tag{6}$$

The parameter (γ) is used to tune the weight of the relative temporal position of the sentences.

Sentence Ranking. From the Latent Dirichlet Allocation (LDA) topic model, we obtain a unigram (event or named entity) probability distribution, $P(t)$. For each topic, the sentence score can be computed using Eq. (7).

$$Score(s) = tp_s \times (\sum_{t \in S}(P(t) \times \alpha \times W_g) + \sum_{t \in S}(P(t) \times \beta \times W_g)) \tag{7}$$

In Eq. (7), W_g can be calculated using Eq. (5), tp_s is the temporal position score of a sentence obtained from Eq. (6), and α and β are the weight factors of the new terms and the topic title terms, respectively, which are learnt from TAC'2010 dataset. For each topic, one sentence is taken as a summary sentence from the ordered list of sentences (descending order of their score, $Score(s)$). We use cosine similarity score to remove the redundancy of the summary. Additionally, we use the same sentence compression technique as in the generic summarization.

4 Evaluation

4.1 ROUGE Evaluation: Generic Summarization

We use the DUC 2004 dataset to evaluate our generic MDS system. We perform our experiment on 35 clusters of 10 documents each. DUC 2004 Task-2 was to create short multi-document summaries no longer than 665 bytes. We evaluate the summaries generated by our system using the automatic evaluation toolkit ROUGE[7] [22]. We compare our system with some recent systems including, the best system in DUC 2004 (Peer 65), a conceptual units based model [34], and G-FLOW, a recent state-of-the-art coherent summarization system [6]. As shown in Table 1, our system outperforms those three systems. It also scores better than the recent submodular functions based state-of-the-art system[8] [23]

[7] ROUGE runtime arguments for DUC 2004:
ROUGE -a -c 95 *-b* 665 *-m -n* 4 *-w* 1.2.

[8] We do not compare our system with the recent topic model based system [14] because that system is significantly outperformed by Lin and Bilmes's [23] system in terms of both ROUGE-1 recall and f_1-measure.

Table 1. Evaluation on the DUC 2004 dataset (The best results are **bolded**)

Systems	R-1	F_1
Peer 65	0.3828	0.3794
Takamura and Okumura [34]	0.3850	-
G-FLOW	0.3733	0.3743
Lin	0.3935	0.3890
Our generic MDS System	**0.3953**	**0.3983**

Table 2. ROUGE-2 and ROUGE-SU4 scores with 95 % confidence on the DUC'2004 dataset

Systems	*ROUGE-2*	*ROUGE-SU4*
Our generic	**0.1017**	**0.142**
MDS System	**(0.0975-0.11)**	**(0.135-0.149)**

in terms of ROUGE-1 recall and f_1-measure. We also include ROUGE-2 and ROUGE-SU4 scores in Table 2 with 95 % confidence. For the DUC 2004 dataset, on average we find weight factors of the groups like events, named-entities, and others, which are 3, 1.14, and 1, respectively. That means that our summarization system assigns the highest priority to the events group and the lowest priority to the normal terms. Hence, it explains the importance of the semantic events in a summarization system. This also explains the importance of named entities over other tokens during summary generation.

4.2 ROUGE Evaluation: Update Summarization

To evaluate our update MDS system, we use the TAC'2011 dataset. TAC'2011 dataset contains two groups of data, A and B. Group A contains the old dataset. Group B contains the new dataset of the same topic as group A. We perform our experiment on 28 clusters of 10 documents each. TAC'2011 guided update summarization task was to create short multi-document summaries no longer than 100 words with the assumption that the user has already read the documents from group A. Table 3 tabulates ROUGE scores of our system and best performing systems in TAC'2011 update summarization task. Our model outperforms the current state-of-the-art system, which is h-uHDPSum, as well as the best update summarization system (peer 43) of TAC'2011 summarization track. 95 % confidence intervals in Table 4 show that our system obtains significant improvement over the two systems (h-uHDPSum and Peer 43) in terms of ROUGE-2 and ROUGE-SU4. The performance of our event and temporal relation based summarizer changes according to the type of documents we are considering to be summarized. Our system gets very high recall and f-measures for the documents that are well constituents of events. Our temporal relation based system reveals

Table 3. Evaluation on the TAC'2011 dataset

Systems	*ROUGE*-2	*ROUGE*-*SU*4
Our update MDS System	**0.1120**	**0.1460**
h-uHDPSum	0.1017	0.1364
Peer 43	0.0959	0.1309

Table 4. 95 % confidence for various systems on the TAC'2011 dataset

Systems	*ROUGE*-2	*ROUGE*-*SU*4
Our update MDS System	**0.1016-0.1244**	**0.1356-0.1587**
h-uHDPSum	0.0910-0.1034	0.1265-0.1473
Peer 43	0.0894-0.1029	0.1251-0.1366

all of the hidden novel information. At the same time, our event and named entity based scoring scheme ensures the saliency in update summarization.

4.3 Manual Evaluation

ROUGE evaluation is not enough to measure the quality of a summary properly. Human evaluation is necessary to get an accurate score of quality. In generic MDS, we use relevancy, non-redundancy, and overall responsivenes criteria to manually evaluate our generic summary. We randomly select 24 clusters from the DUC 2004 dataset and assign a total of 3 human assessors for the evaluation purpose. Each assessor examines the summaries from all 24 clusters that are generated by our system and gives a score of 1 (Very Poor) to 5 (Very Good). Finally average scores are calculated. Table 5 tabulates average scores of manual evaluation on DUC 2004 dataset.

Our event-based summarization system chooses the high relevance sentences as the summary sentences. We observe that the cosine similarity checking performs poorly in removing redundancy. It has been shown in the literature that highly responsive summary "would have redundancy to some extent" [34]. This may be the reason why our summarization system does not perform well in checking redundancy as we had hoped.

In update MSD, we use the following criteria to manually evaluate our update summaries: novelty (containing update information), readability/fluency,

Table 5. Manual evaluation on the DUC 2004 dataset

Relevancy	3.92
Non-redundancy	3.50
Overall responsiveness	3.70

Table 6. Manual evaluation on the TAC 2011 dataset

Novelty	4.13
Fluency	3.92
Overall responsiveness	4.07

and overall responsiveness (overall focus and content). We randomly select 21 clusters from TAC 2011 dataset. Table 6 tabulates average scores of manual evaluation on TAC 2011 dataset.

Our temporal summarization system chooses highly novel sentences as summary sentences without losing fluency and responsiveness.

5 Conclusion and Future Work

In this paper, we have shown a simple yet effective way of approaching the task of generating multi-document generic summaries. The importance of semantic events and named entities in generating summaries has been deeply analyzed using the LDA topic model. By dividing terms into different groups we achieve high ROUGE-1 recall and f_1 scores for generic MDS task. Our update summarization model can identify novel information based on temporal ordering of events. Our system outperforms the state-of-the-art update summarization system based on ROUGE-2 and ROUGE-SU4 recall measures. There is still much room to improve event-event and event-time ordering. Ordering temporal entity considering all possible 12 relations is an NP-complete problem. Denis and Muller [8] reduce the complexity of the problem by converting relations into end points, but they get only 41 % F1-score. By increasing the recall and precision of event-event and event-time relation extraction, it is possible to get better temporal ordering of sentences. This will eventually provide better update summarization. We believe that some recent works on temporal relation classification using dependency parses [27] and discourse analysis framework [28] can further improve our update summarization system performance.

References

1. James, F.: Allen.: maintaining knowledge about temporal intervals. Commun. ACM **26**(11), 832–843 (1983)
2. Bethard, S.: Cleartk-timeml: a minimalist approach to tempeval. In: Second Joint Conference on Lexical and Computational Semantics (* SEM), vol. 2, pp. 10–14 (2013)
3. Boudin, F., El-Bèze, M., Torres-Moreno, J. M.: A scalable MMR approach to sentence scoring for multi-document update summarization. COLING (2008)
4. Cer, D.M., De Marneffe, M.-C., Jurafsky, D., Manning, C.D.: Parsing to stanford dependencies: trade-offs between speed and accuracy. In: LREC (2010)
5. Chang, A.X., Manning, C.D.: Sutime: a library for recognizing and normalizing time expressions. In: Language Resources and Evaluation (2012)

6. Christensen, J., Mausam, S.S., Etzioni, O.: Towards coherent multi-document summarization. In: Proceedings of NAACL-HLT, pp. 1163–1173 (2013)
7. Delort, J.-Y., Alfonseca, E.: Dualsum: a topic-model based approach for update summarization. In: Proceedings of the 13th Conference of the European Chapter of the Association for Computational Linguistics, pp. 214–223 (2012)
8. Denis, P., Muller, P.: Predicting globally-coherent temporal structures from texts via endpoint inference and graph decomposition. In: Proceedings of the Twenty-Second International Joint Conference on Artificial Intelligence, vol. 3, pp. 1788–1793. AAAI Press (2011)
9. Pan, D., Guo, J., Zhang, J., Cheng, X.: Manifold ranking with sink points for update summarization. In: Proceedings of the 19th ACM International Conference on Information and Knowledge Management, pp. 1757–1760 (2010)
10. Erkan, G., Radev, D.R.: Lexrank: graph-based lexical centrality as salience in text summarization. J. Artif. Intell. Res. (JAIR) **22**(1), 457–479 (2004)
11. Filatova, E., Hatzivassiloglou, V.: Event-based extractive summarization. In: Proceedings of ACL Workshop on Summarization, vol. 111 (2004)
12. Fisher, S., Roark, B.: Query-focused supervised sentence ranking for update summaries. In: Proceeding of TAC 2008 (2008)
13. Gillick, D., Favre, B., Hakkani-Tur, D., Bohnet, B., Liu, Y., Xie, S.: The icsi/utd summarization system at tac. In: Proceedings of the Second Text Analysis Conference, Gaithersburg, Maryland, USA. NIST (2009)
14. Haghighi, A., Vanderwende, L.: Exploring content models for multi-document summarization. In: Proceedings of Human Language Technologies: The Annual Conference of the North American Chapter of the Association for Computational Linguistics, pp. 362–370. Association for Computational Linguistics (2009)
15. Kullback, S.: The kullback-leibler distance (1987)
16. Li, J., Li, S., Wang, X., Tian, Y., Chang, B.: Update summarization using a multi-level hierarchical dirichlet process model. In: COLING (2012)
17. Li, L., Heng, W., Jia, Y., Liu, Y., Wan, S.: Cist system report for acl multiling 2013-track 1: multilingual multi-document summarization. In: MultiLing 2013, p. 39 (2013)
18. Li, P., Wang, Y., Gao, W., Jiang, J.: Generating aspect-oriented multi-document summarization with event-aspect model. In: Proceedings of the Conference on Empirical Methods in Natural Language Processing, pp. 1137–1146 (2011)
19. Li, W., Mingli, W., Qin, L., Wei, X., Yuan, C.: Extractive summarization using inter-and intra-event relevance. In: Proceedings of the 21st International Conference on Computational Linguistics and the 44th Annual Meeting of the Association for Computational Linguistics, pp. 369–376. Association for Computational Linguistics (2006)
20. Li, X., Liang, D., Shen, Y.-D.: Graph-based marginal ranking for update summarization. In: SDM, pp. 486–497. SIAM (2011)
21. Li, Xuan, Liang, Du, Shen, Yi-Dong: Update summarization via graph-based sentence ranking. IEEE Trans. Knowl. Data Eng. **25**(5), 1162–1174 (2013)
22. Lin, C.-Y.: Rouge: a package for automatic evaluation of summaries. In: Text Summarization Branches Out: Proceedings of the ACL-2004 Workshop, pp. 74–81 (2004)
23. Lin, H., Bilmes, J.: A class of submodular functions for document summarization. In: ACL, pp. 510–520 (2011)
24. Mani, I.: Automatic Summarization, vol. 3. John Benjamins Publishing, Amsterdam (2001)

25. Mani, I., Schiffman, B., Zhang, J.: Inferring temporal ordering of events in news. In: Proceedings of the Conference of the North American Chapter of the Association for Computational Linguistics on Human Language Technology: Companion Volume of the Proceedings of HLT-NAACL 2003-Short Papers, vol. 2, pp. 55–57. Association for Computational Linguistics (2003)
26. Mihalcea, R., Tarau, P.: Textrank: Bringing order into texts. In: Proceedings of EMNLP, vol. 4, p. 275, Barcelona, Spain (2004)
27. Ng, J.-P., Kan, M.-Y.: Improved temporal relation classification using dependency parses and selective crowdsourced annotations. In: COLING, pp. 2109–2124 (2012)
28. Ng, J.-P., Kan, M.-Y., Lin, Z., Feng, W., Chen, B., Jian, S., Tan, C.L.: Exploiting discourse analysis for article-wide temporal classification. In: EMNLP, pp. 12–23 (2013)
29. Page, L., Brin, S., Motwani, R., Winograd, T.: The pagerank citation ranking: Bringing order to the web (1999)
30. Martin, F.: Porter.: an algorithm for suffix stripping. Program Electr. Libr. Inf. Syst. **14**(3), 130–137 (1980)
31. Pustejovsky, J., Castano, J.M., Ingria, R., Sauri, R., Gaizauskas, R.J., Setzer, A., Katz, G., Radev, D.R.: Timeml: robust specification of event and temporal expressions in text. In: New Directions in Question Answering, vol. 3, pp. 28–34 (2003)
32. Steinberger, J., Ježek, K.: Update summarization based on novel topic distribution. In: Proceedings of the 9th ACM symposium on Document Engineering, pp. 205–213 (2009)
33. Steinberger, J., Kabadjov, M., Steinberger, R., Tanev, H., Turchi, M., Zavarella, V.: Jrcs participation at tac: Guided and multilingual summarization tasks. In: Proceedings of the Text Analysis Conference (TAC) (2011)
34. Takamura, H., Okumura, M.: Text summarization model based on maximum coverage problem and its variant. In: Proceedings of the 12th Conference of the European Chapter of the Association for Computational Linguistics, pp. 781–789. Association for Computational Linguistics (2009)
35. Teh, Y.W., Jordan, M.I., Beal, M.J., Blei, D.M.: Hierarchical dirichlet processes. J. Am. Stat. Assoc. **101**, 1566–1581 (2004)
36. Wenjie, L., Wei Furu, L., Qin, H.Y.: Pnr 2: ranking sentences with positive and negative reinforcement for query-oriented update summarization. In: Proceedings of the 22nd International Conference on Computational Linguistics, pp. 489–496 (2008)
37. Zhang, R., Li, W., Qin, L.: Sentence ordering with event-enriched semantics and two-layered clustering for multi-document news summarization. In: Proceedings of the 23rd International Conference on Computational Linguistics, pp. 1489–1497 (2010)