A systematic analysis of sentence update detection for temporal summarization

Gârbacea, C.; Kanoulas, E.

DOI
10.1007/978-3-319-56608-5_33

Publication date
2017

Document Version
Final published version

Published in
Advances in Information Retrieval

A Systematic Analysis of Sentence Update Detection for Temporal Summarization

Cristina Gârbacea[1] and Evangelos Kanoulas[2(✉)]

[1] University of Michigan, Ann Arbor, MI, USA
garbacea@umich.edu
[2] University of Amsterdam, Amsterdam, The Netherlands
e.kanoulas@uva.nl

Abstract. Temporal summarization algorithms filter large volumes of streaming documents and emit sentences that constitute salient event updates. Systems developed typically combine in an ad-hoc fashion traditional retrieval and document summarization algorithms to filter sentences inside documents. Retrieval and summarization algorithms however have been developed to operate on static document collections. Therefore, a deep understanding of the limitations of these approaches when applied to a temporal summarization task is necessary. In this work we present a systematic analysis of the methods used for retrieval of update sentences in temporal summarization, and demonstrate the limitations and potentials of these methods by examining the retrievability and the centrality of event updates, as well as the existence of intrinsic inherent characteristics in update versus non-update sentences.

Keywords: Temporal summarization · Content analysis · Event modeling

1 Introduction

Monitoring and analyzing the rich and continuously updated content in an online environment can yield valuable information that allows users and organizations gain useful knowledge about ongoing events and consequently, take immediate action. News streams, social media, weblogs, and forums constitute a dynamic source of information that allows individuals, corporations and government organizations not only to communicate information but also to stay informed on "what is happening right now". The dynamic nature of these sources calls for effective ways to accurately monitor and analyze the emergent information present in an online streaming setting.

TREC Temporal Summarization (TS) [5] facilitates research in monitoring and summarization of information associated with an event over time. Given an event query, the event type[1], and a high volume stream of input documents

[1] TREC TS focuses on large events with a wide impact, such as natural catastrophes (storms, earthquakes), conflicts (bombings, protests, riots, shootings) and accidents.

J.M. Jose et al. (Eds.): ECIR 2017, LNCS 10193, pp. 424–436, 2017.
DOI: 10.1007/978-3-319-56608-5_33

discussing the event, a temporal summarization system is required to emit a series of event updates, in the form of sentences, over time, describing the named event. An optimal summary covers all the essential information about the event with no redundancy, and each new piece of information is added to the summary as soon as it becomes available.

Temporal summarization systems typically use a pipelined approach, (a) filtering documents to discard those that are not relevant to the event, (b) ranking and filtering sentences to identify those that contain significant updates around the event, and (c) deduplicating/removing redundant sentences that contain information that has already been emitted; some examples of the afore-described pipeline constitute systems submitted to TREC TS in past years [5]. In this work we are only focusing on identifying potential update sentences and their retrieval from a large corpus for summarization purposes; that is, we assume that all incoming documents are relevant to the event under consideration, and we deliberately choose to ignore the past history of what event updates have been emitted by the summarization system. These assumptions, which are ensured by the construction of our experiments and evaluations, provide a decomposition of the temporal summarization problem and allow a focus on fundamental theories behind understanding what constitutes a potential event update (from now on simply referred as update) and what not. We leave the study of the interplay of the three components as future work.

Event update identification algorithms fall under one of the categories below, or apply a combination of these methods [10]:

1. **Retrieval algorithms** that consider event updates as passages to be retrieved given a event query;
2. **Event update centrality** algorithms that assume update sentences are central in the documents that contain them, and hence algorithms that can aggregate sentences should be able to identify them;
3. **Event update modeling** methods that consider events bear inherit characteristics that are not encountered in non-update sentences, and hence algorithms that model event updates should be able to predict whether a sentence is an update or not.

In this work we present a systematic analysis of the limitations and potentials of the three approaches. We do **not** devise any new algorithm towards temporal summarization; our goal is to obtain a deeper **understanding of how and why** the aforementioned approaches fail, and what is required for a successful temporal summarization system. We believe that such an analysis is necessary and can shed light in developing more effective algorithms in the future.

The remainder of this paper is organized as follows: Sect. 2 describes prior initiatives and methods for temporal summarization of news events, Sect. 3 discusses the experimental design of our study, Sect. 4 describes the experimental results, and provides an analysis of these results around the limitations of the methods being tested, and last Sect. 5 outlines the conclusions of our work as well as future directions informed by these conclusions.

2 Related Work

Events play a central role in many online news summarization systems. Topic Detection and Tracking [2] has focused on monitoring broadcast news stories and issuing alerts about seminal events and related sub-events in a stream of news stories at document level. To retrieve text at different granularities, passage retrieval methods have been widely employed; see TREC HARD track [1] and INEX adhoc [11] initiatives for an overview. Passages are typically treated as documents, and existing language modeling techniques that take into account contextual information, the document structure or the hyperlinks contained inside the document are adapted for retrieval.

Single and multi-document summarization have been long studied by the natural language processing and information retrieval communities [4,10]. Such techniques take as input a set of documents on a certain topic, and output a fixed length summary of these documents. Clustering [20], topic modeling [3], and graph-based [7,14] approaches have been proposed to quantify the salience of a sentence within a document. McCreadie et al. [13] combine traditional document summarization methods with a supervised regression model trained on features related to the prevalence of the event, the novelty of the content, and the overall sentence quality. Kedzie et al. [12] also employ a supervised approach to predict the salience of sentences. Features combined include basic sentence quality features, query features, geotags and temporal features, but also features that represent the contrast between a general background corpus language model and a language model per event category. Gupta et al. [9] use background and foreground corpora to weight discriminative terms for topic-focused multi-document summarization. Finally, Chakrabarti et al. [6] and Gao et al. [8] combine evidence from event news and social media to model the different phases of an event using Hidden Markov Models and Topic Models.

3 Experimental Design

In this section we describe the experimental design used for our analysis. We consider three different approaches that have been adopted so far towards detecting event updates: (1) retrieval algorithms, (2) event update centrality algorithms, and (3) event update modeling methods.

1. Retrieval Algorithms: The primary goal of the experiments is to identify the limitations of retrieval algorithms towards temporal summarization of events. In the designed experiments we want to be as indifferent as possible to any particular retrieval algorithm; hence we focus on the fundamental component of any such algorithm which is the overlap between the language of an event query and the language of an event update in terms of shared vocabulary. If an event update does not contain any query term for instance, it is impossible to be retrieved by any lexical-based relevance model. This can give us a theoretical upper bound on the number of event updates that are at all retrievable. Clearly,

even if an event update contains the query terms (we call that event update *covered*) it is still likely that it may not be retrieved, if for instance the query terms are not discriminative enough to separate the update from non-updates. Hence, we focus in our analysis on discriminative terms. To identify such terms, we compute word likelihood ratios. The log-likelihood ratio (LLR) [9,19] is an approach for identifying discriminative terms between corpora based on frequency profiling. To extract the most discriminative keywords that characterize events, we construct two corpora as follows. We consider all relevant annotated sentence updates from the gold standard as our foreground corpus, and a background corpus is assembled of all the non-update sentences from the relevant documents. Afterwards, for each term in the foreground corpus we compute its corresponding LLR score. In order to quantify which are the most discriminative terms in our collection, we rank the terms in descending order of their LLR scores and consider the top-N most discriminative in the rest of our experiments.

(Query Expansion with Similar Terms). We further want to understand the fundamendal reason behind any language mismatch between query and event updates. A first hypothesis is that such a mismatch is due to *different lexical representation* for the same semantics. Hence, in a second experiment we expand queries in two different ways: (a) we select a number of synonym terms using WordNet [16], and (b) we use a Word2Vec [15] model trained on the set of relevant gold standard updates from TREC TS 2013 and 2014; then similar to the previous experiment we test the limitations of such an approach examining whether the expanded query terms are also event update discriminative terms.

(Query Expansion with Relevance Feedback). A second hypothesis is that a vocabulary mismatch is due to a *topical drift* of the event updates. Imagine the case of the *"Boston Marathon Bombing"*. Early updates may contain all the query words, however when the topic drifts to the trial of the bombers or the treatment of the injured, it is expected that there will be a low overlap between the event query and the event updates due to the diverging vocabulary used. Such a vocabulary gap would be hard to fill by any synonym or related terms. However, if one were to consider how the vocabulary of the updates changes over time, one might be able to pick up new terms from past updates that could help in identifying new updates. This is a form of relevance feedback. To assess this hypothesis, given an update, we consider all the sentence updates that have appeared in documents prior to this update. Then we examine the vocabulary overlap between this current update and discriminative terms from past updates. A high overlap would designate that one can actually gradually track topical drift.

2. Event Update Centrality: Here we devise a set of experiments to test whether an event update is central in the documents that contain it. If this is the case, algorithms that can aggregate sentences should be able to identify relevant and informative updates. Graph-based ranking methods have been proposed for document summarization and keyword extraction tasks [7,14]. These methods construct a sentence network, assuming that important sentences are linked to

many other important sentences. The underlying model on which these methods are based is a random walk model on weighted graphs: an imaginary walker starts walking from a node chosen arbitrarily, and from that node continues moving towards one of its neighbouring nodes with a probability proportional to the weight of the edge connecting the two nodes. Eventually, probabilities of arriving at each node on the graph are produced; these denote the popularity, centrality, or importance of each node.

(Within Document Centrality). In this first experiment we are interested in testing whether an event update is central within the document that contains it. This scenario would be the ideal, since if this is the case, centrality algorithms running on incoming documents could emit event updates in a timely manner. To this end, we use LexRank [7], a state-of-the-art graph-based summarization algorithm, and examine the ranking of event updates within each document.

We pick LexRank to assess the salience of event updates as it is one of the best-known graph-based methods for multi-document summarization based on lexical centrality. Words and sentences can be modeled as nodes linked by their co-occurrence or content similarity. The complexity of mining the word network only depends on the scale of the vocabulary used inside the documents; it is often significantly reduced after applying term filtering. LexRank employs the idea of random walk within the graph to do prestige ranking as PageRank [17] does. We rely on the MEAD summarizer [18] implementation to extract the most central sentences in a multi-document cluster.

(Across Documents Centrality). Here we perform a maximal information experiment in which we are interested in assessing the ranking of sentence updates across documents. If this is the case, it signifies that even though sentence updates appear not to be central inside single documents, they become central as information is accumulated. We are aware that devising such an algorithm would not be providing users with timely updates, however in this experiment we want to identify the upper bound of centrality-based algorithms towards event summarization. Therefore we purposefully ignore the temporal aspect.

3. Event Update Modeling: We test the hypothesis that event updates bear inherent characteristics which are not encountered in non-update sentences. If this is indeed the case, then one might be able to devise a method that uses these inherent characteristics to predict whether a sentence is an update or not. We model the inherent characteristics of a general event update as the set of terms with high log-likelihood ratio, i.e. the set of the most discriminative event terms. Since extracting the most discriminative terms for an event at hand from the gold standard annotations would result in a form of overfitting – we learn from and predict on the same dataset – we devise two experiments.

(General Event Update Modeling). In the first experiment we test the hypothesis that an event update can be distiguished from a non-update independent of the event particulars or the event type. We define a general event as any event in our collection irrespective of the event type. We use the log-likelihood ratio test to identify the most discriminative terms in event updates

vs. non-updates[2]. Afterwards we examine the degree of overlap between the extracted discriminative terms with the annotated updates for each test event.

(Event-Type Update Modeling). Given that different event types may be expressed using a different vocabulary, we repeat the experiment described above considering only events that have the same event type in common (as already mentioned, event types can be natural catastrophes, conflicts, accidents, etc.). Our goal is to learn discriminative LLR terms that are specific to a particular type of event. We use the annotated sentences from the gold standard for each event type in building our foreground corpus; the background corpus is made up of all non-update sentences from the relevant documents per event type.[3]

4 Results and Analysis

4.1 Datasets

In all our experiments we use the TREC KBA 2014 Stream Corpus[4] used by the TREC 2014 TS track. The corpus (4.5 TB) consists of timestamped documents from a variety of news and social media sources, and spans the time period October 2011–April 2013. Each document inside the corpus has a timestamp representing the moment when the respective document was crawled, and each sentence is uniquely identified by the combination document identifier and positional index of the sentence inside the document.

We run our experiments on two pre-filtered collections released by the TREC TS organizers based on the KBA corpus that are more likely to include relevant documents for our events of interest. The testsets provided contain 10 event queries for the TREC TS 2013 collection (event ids 1–10), and 15 event queries for the TREC TS 2014 collection (event ids 11–25). For each event, sentences in documents have been annotated as either updates or non-updates through an in depth-pooling experiment. Each event update contains one or more critical units of information, called information nuggets. The goal of a temporal summarization system is to emit event updates that cover all information nuggets. Information nuggets were extracted from update sentences by the TREC TS co-ordinators, and were used to further identify sentence updates not included in the original pool. In our evaluation we use this extended set of updates.

4.2 Retrieval Algorithms: Are Event Updates Retrievable?

The first question we want to answer is to what extent there is a language overlap between the event queries (and query expansions) with the event updates. To get a theoretical upper bound, we first examine how many event updates

[2] In total we extract 8,471 unigrams and 1,169,276 bigrams using the log-likelihood ratio weighting scheme.

[3] We discard event types for which there is not enough annotated data available.

[4] http://trec-kba.org/kba-stream-corpus-2014.shtml.

contain: *(i)* at least one query term, *(ii)* at least one query term after WordNet and Word2Vec query expansion, and *(iii)* at least one query term after query expansion with all the terms from event updates found in documents prior to the current update (relevance feedback). We observe that on average *24.4%* of event updates are guaranteed never to be retrieved by a traditional retrieval algorithm; this percentage remains unchanged when the query is expanded by WordNet synonyms, while it drops to *22.7%* of updates by a query expanded with Word2Vec[5]. Examples of expansion terms are shown in Table 1. Relevance feedback when using all query terms in past event updates lowers the amount of uncovered updates to *16%* on average across all event queries. Therefore, this also signifies that the upper bound performance for retrieval algorithms reaches approximately *84%* update coverage on average. Hence, retrieval algorithms with relevance feedback might be able to account for vocabulary gap and topic drift in the description of sub-events.

Table 1. Expansion terms and their rank on the basis of the log-likelihood ratio value (−1 designates that the term does not appear in the list of extracted LLR terms).

Event id	Query term rank	Any WordNet synonym rank	Any similar Word2Vec term rank
9	(guatemala, 2), (earthquak, 24)	(guatemala, 2), (earthquak, 24)	(guatemala, 2), (quak, 16), (earthquak, 24), (philippin, 36), (hit, 64), (7.4-magnitud, 112), (strong, 138), (struck, 201), (magnitud, 238), (strongest, 368), (caribbean, 586), (temblor, 5451), (tremor, 8303)
11	(concordia, 157), (costa, 183)	(concordia, 157), (costa, 183), (rib, −1)	(concordia, 157), (costa, 183), (shipwreck, 3636), (liner, 4793), (keel, 6252), (ill-fat, 6856), (vaus, 7721), (wreck, 8070), (genoa-bas, −1), (lean, −1), (7:13, −1), (raze, −1), (rica, −1)
22	(protest, 1), (bulgarian, 96)	(protest, 1), (bulgarian, 96)	(protest, 1), (resign, 74), (bulgarian, 96), (demonstr, 132), (bulgaria, 133),(dhaka, 167), (amid, 182), (ralli, 186), (shahbag, 235), (shahbagh, 478), (finmin, 547), (borissov, 630), (revok, 1183), (borisov, 1197), (boyko, 3284), (tender, 3469), (gerb, 4517), (activist, 8055)

In order to be realistic though, we compute likelihood ratios for words in our corpus. We first consider annotated updates as our foreground corpus, and non-updates as our background corpus. We rank terms on the basis of their

[5] Word2Vec was trained on the set of gold standard updates from the TREC TS 2013 and TREC TS 2014 collections.

discriminative power. In Table 1, in Column 1 we report on the query terms and their rankings among the most discriminative LLR terms extracted from the TREC TS 2013 and TREC TS 2014 collections. We observe that in general, query terms appear to be ranked high up in the list of discriminative terms. We repeat the same experiment after expanding query terms with WordNet and Word2Vec synonyms, and report on the ranks of the expanded query terms inside the list of LLR terms with high discriminative power in Table 1, in Columns 2 and 3. We observe that these query expansion terms are not very discriminative in general, although Word2Vec (trained on the test set) is able to pick up some discriminative terms.

Conclusion: Event query terms are central in event updates, however they cannot cover all updates, nor are they the most discriminative terms (e.g. see *"costa concordia"* in Table 1). A temporal analysis is necessary to identify whether the language gap is more evident as the event develops, however we leave this as future work. Further, based on the afore-described observations the language gap is not due to a lexical mismatch between the query and the updates, but rather due to topic drifting. Therefore, a dynamic algorithm that can adapt the lexical representation of a query – possibly by the means of relevance feedback – could bridge this gap.

4.3 Event Update Centrality: Do Event Updates Demonstrate Centrality?

Summarization methods applied at document level assume that event updates demonstrate centrality inside the documents they appear in. In the next set of experiments we test whether it is the case that event updates demonstrate centrality characteristics. Ideally, update sentences are central and salient inside the documents they are found in. This would allow a summarization algorithm to identify updates as soon as a document has streamed in.

To assess the within-document centrality of updates we run LexRank on each incoming document. We process the LexRank output to infer rankings inside documents for the set of relevant event updates. After ranking each sentence within a document, we compute three measures: precision at rank cut-off 1, precision at 10, and R-precision, where R is the number of update sentences within the document. The results of the experiment are shown as a heatmap in Fig. 1 – the first three columns, denoted as (A)[6]. The average precision values across the two collections can be found bellow the heatmap, while Table 2 shows the average values for each collections separately. For the TREC TS 2013 collection (events 1–10), we can see that it is rarely the case that event updates make it to the top of the ranking inside single documents. However, for the TREC TS 2014 dataset (events 11–25) we observe higher precision scores, especially in the top-10 positions.

[6] No documents were released for event 7, hence the white row in the heatmap.

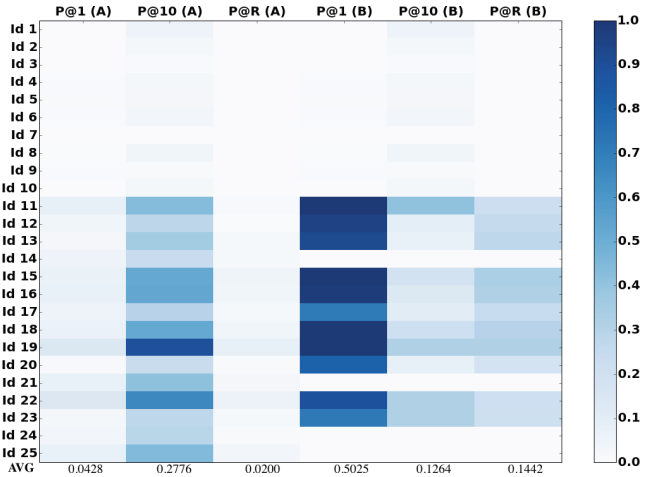

Fig. 1. Within – (A) – and across – (B) – document centrality scores based on LexRank.

To better understand the difference between the two collections we considered the case of a random algorithm that simply shuffles all sentences within a document, and ranks them by this permuted order. The intuition behind this experiment is that differences in document lengths betwen the two collections may affect the precision numbers observed - in short documents it is easier to achieve a higher precision. The *mean precision of the random algorithm at 10* for the 2013 collection is *0*, while for the 2014 collection is *0.028* - statistically significantly worse than the corresponding centrality scores. Hence, there is no clear reason for the observed differences between the two collections, and further investigation is required, that may also extend to missing judgement of sentences in the 2013 collection.

Table 2. Mean precision values for within – (A) – and across – (B) – document centrality for TREC TS datasets.

Average	P@1 (A)	P@10 (A)	P@R (A)	P@1 (B)	P@10 (B)	P@R (B)
2013	0.0045	0.0279	0.0003	0.0045	0.0279	0.0003
2014	0.0667	0.4366	0.0326	0.7151	0.1667	0.2028

We then take a retrospective look at the centrality of sentences by considering centrality scores across all relevant documents in each collection. The LexRank algorithm is now run over the entire corpus (multi-document sentence centrality), and sentences are then ranked with respect to the output scores. To make the two algorithms (within and across documents) comparable, we examine each document separately. First we rank the sentences within each document in accordance to the overall document ranking produced by LexRank, and then

we compute the same three measures. The values can be seen in the form of a heatmap in Fig. 1 – the last three columns, denoted as (B).[7] First, we observe that the same pattern preserves for these two different collections. While computing centrality across documents does not change the precision values for the TREC TS 2013 dataset at all, for the TREC TS 2014 collection we can see a considerable increase. TREC TS 2014 annotated updates demonstrate centrality within and across documents, rendering them central in the development of the events under consideration. Furthermore, across-document centrality appears to bring some rather central updates at the very top of the ranked list, but within-document centrality appears to have a better effect on lower - up to 10 - ranks. Across-document centrality of sentences can also increase R-precision, demonstrating a robust behaviour.

Conclusion: Sentence centrality, when computed within a single document, does not appear to be a strong signal that can designate whether a sentence is an update or not. When computed across all documents, it consistently improves all measures. Such an algorithm, however, is not particularly useful since it has to wait for all documents to be streamed in before identifying any update sentences. One could, however, examine the minimum number of documents it takes for such a summarization algorithm before salient updates make it to the top of the ranking. We leave the construction of such an algorithm for future work.

4.4 Event Update Modeling: Do Event Updates Present Inherent Characteristics?

Given the results of the previous experiment, a hypothesis to test is whether knowing beforehand event discriminative terms can help in retrieving event updates. Clearly, different event types may have different inherent characteristics; for instance, it is likely that an event of type *accident* does not share the same characteristics as an event of type *protest*. Hence, we perform our analysis on different slices of the data.

First we create a general model of event updates by considering non-update sentences as a background corpus and update sentences as a foreground corpus. Then we compute the overlap between discriminative terms from this general model across all events and their types with the update sentences of the event under consideration. One can see in Fig. 2 – Column 1 that discriminative terms belonging to the general model appear on average in *95%* of the event updates. Note that this is not a theoretical upper bound, but rather an average case analysis, since terms with high LLR scores should in general be able to discriminate update from non-update sentences.

We repeat the same experiment, this time for each event type separately. We compute the overlap between the discriminative terms from the event type model

[7] For events 14, 21, 24 and 25 we cannot report on any centrality scores across relevant documents due to the size of the data and the inability of LexRank to handle it - hence the white rows in the heatmap in columns (B). The average values for the precision measures below the heatmap are computed excluding these events.

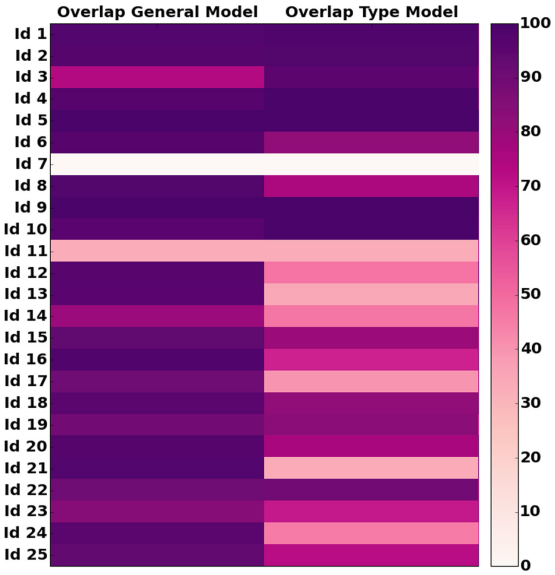

Fig. 2. Degree of overlap of discriminative terms with the TREC TS event updates.

and the annotated sentence updates, and present results for these experiments in Fig. 2 – Column 2. Interestingly, when mining event specific terms the degree of overlap drops to *72.84%* (it actually increases for the TREC TS 2013 collection to *94.28%* according to our intuition, but deteriorates for the TREC TS 2014 collection to *59.97 %*). This is against our hypothesis, as we were expecting that event specific discriminative terms will only increase the degree of overlap with the relevant sentence updates. We assume this happens due to the smaller size of the event type dataset used as a foreground corpus. The resulting event specific LLR terms are fewer but with a higher discriminative power, although we do not consider it when computing the overlap between the two models. In addition to this, we are using a fixed cut-off threshold (top 100) in our experiments for selecting terms from the discriminative list up until a specific rank. It could be that if we chose another threshold results would look different, however we leave the exploration of optimal cut-offs as future work towards devising effective algorithms.

Conclusion: Modeling event updates bears great promises towards devicing temporal summarization algorithms. It appears from our experiments that there is a number of discriminative keywords that can indicate the presence of an update sentence. The models built in this experiments somewhat overfit the data (all events were used to develop the models). A follow up experiment should perform a leave-one-out cross-validation to also test the predictive power of these terms. Nevertheless, it is clear from the results above this third approach in temporal summarization reserves more attention.

5 Conclusions

In conclusion, we have presented a systematic analysis of sentence retrieval for temporal summarization, and examined the retrievability, centrality, and inherent characteristics of event updates. We designed and ran a set of experiments on the theoretical upper bounds where possible, and on more realistic upper bounds with the use of discriminative terms obtained through likelihood ratio calculations. Our experimental design decisions are driven by abstraction whenever feasible, and state-of-the-art work where not possible.

Our results suggest that retrieval algorithms with query expansion have a theoretical upper bound that does not allow for the identification of all relevant event updates. A topical drift can be partially captured by (pseudo-)relevance feedback, however its performance is still bounded below 100% coverage. Further, we assessed sentence centrality with the use of graph-based methods and observed that update sentences are also salient sentences when enough documents are accumulated. The question that remains unanswered is what is the amount of information that needs to flow into the system before such salience can be reliably assessed. Last, modeling event updates through discriminative terms looks like a promising step towards improving the performance of a temporal summarization system. One thing that was not analyzed in this study is the interplay across these three categories of algorithms, and whether one could complement the other, or in which cases one is better than the other; we leave this as future work.

Finally, we believe that we provide evidence that can guide future research on the topic, and that our analysis is unique and original in the enormous space of temporal summarization research. We consider that certain directions have been outlined by our work, and we intend to explore these further in the future.

Acknowledgements. This research was supported by the Dutch national program COMMIT. All content represents the opinion of the authors, which is not necessarily shared or endorsed by their respective employers and/or sponsors.

References

1. Allan, J.: HARD track overview in TREC 2003 high accuracy retrieval from documents. Technical report, DTIC Document (2005)
2. Allan, J., Carbonell, J.G., Doddington, G., Yamron, J., Yang, Y.: Topic detection and tracking pilot study final report (1998)
3. Allan, J., Gupta, R., Khandelwal, V.: Topic models for summarizing novelty. In: ARDA Workshop on LMIR, Pennsylvania (2001)
4. Allan, J., Papka, R., Lavrenko, V.: On-line new event detection and tracking. In: Proceedings of the 21st ACM SIGIR Conference, pp. 37–45 (1998)
5. Aslam, J.A., Diaz, F., Ekstrand-Abueg, M., McCreadie, R., Pavlu, V., Sakai, T.: TREC 2015 temporal summarization. In: Proceedings of the 24th TREC Conference 2015, Gaithersburg, MD, USA (2015)
6. Chakrabarti, D., Punera, K.: Event summarization using Tweets. ICWSM **11**, 66–73 (2011)

7. Erkan, G., Radev, D.R.: Lexrank: graph-based lexical centrality as salience in text summarization. J. Artif. Intell. Res. **22**, 457–479 (2004)
8. Gao, W., Li, P., Darwish, K.: Joint topic modeling for event summarization across news and social media streams. In: Proceedings of the 21st ACM CIKM Conference, pp. 1173–1182. ACM (2012)
9. Gupta, S., Nenkova, A., Jurafsky, D.: Measuring importance and query relevance in topic-focused multi-document summarization. In: Proceedings of the 45th ACL Interactive Poster and Demonstration Sessions, pp. 193–196. ACL (2007)
10. Imran, M., Castillo, C., Diaz, F., Vieweg, S.: Processing social media messages in mass emergency: a survey. ACM Comput. Surv. (CSUR) **47**(4), 67 (2015)
11. Kamps, J., Pehcevski, J., Kazai, G., Lalmas, M., Robertson, S.: INEX 2007 evaluation measures. In: Fuhr, N., Kamps, J., Lalmas, M., Trotman, A. (eds.) INEX 2007. LNCS, vol. 4862, pp. 24–33. Springer, Heidelberg (2008). doi:10.1007/978-3-540-85902-4_2
12. Kedzie, C., McKeown, K., Diaz, F.: Predicting salient updates for disaster summarization. In: Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics, ACL, pp. 1608–1617 (2015)
13. McCreadie, R., Macdonald, C., Ounis, I.: Incremental update summarization: adaptive sentence selection based on prevalence and novelty. In: Proceedings of the 23rd ACM CIKM Conference, pp. 301–310. ACM (2014)
14. Mihalcea, R., Tarau, P.: Textrank: bringing order into texts. ACL (2004)
15. Mikolov, T., Sutskever, I., Chen, K., Corrado, G.S., Dean, J.: Distributed representations of words and phrases and their compositionality. In: Advances in NIPS, pp. 3111–3119 (2013)
16. Miller, G.A.: Wordnet: a lexical database for english. Commun. ACM **38**(11), 39–41 (1995)
17. Page, L., Brin, S., Motwani, R., Winograd, T.: The pagerank citation ranking: bringing order to the web (1999)
18. Radev, D.R., Allison, T., Blair-Goldensohn, S., Blitzer, J., Celebi, A., Dimitrov, S., Drabek, E., Hakim, A., Lam, W., Liu, D., et al.: Mead-a platform for multidocument multilingual text summarization. In: LREC (2004)
19. Rayson, P., Garside, R.: Comparing corpora using frequency profiling. In: Proceedings of the Workshop on Comparing Corpora, pp. 1–6. ACL (2000)
20. Vuurens, J.B.P., de Vries, A.P., Blanco, R., Mika, P.: Online news tracking for ad-hoc information needs. In: Proceedings of the 2015 lCTIR Conference, MA, USA, 27–30 September 2015, pp. 221–230 (2015)