# Updating Users About Time Critical Events[⋆]

Qi Guo[1], Fernando Diaz[2], and Elad Yom-Tov[3]

[1] Microsoft Corporation, One Microsoft Way, WA, 98052, qiguo@microsoft.com
[2] Microsoft Research, 102 Madison Ave, New York, NY 10016, fdiaz@microsoft.com
[3] Microsoft Research, 13 Shenkar St, Herzliya 46275, Israel, eladyt@microsoft.com

**Abstract.** During unexpected events such as natural disasters, individuals rely on the information generated by news outlets to form their understanding of these events. This information, while often voluminous, is frequently degraded by the inclusion of unimportant, duplicate, or wrong information. It is important to be able to present users with only the novel, important information about these events as they develop. We present the problem of updating users about time critical news events, and focus on the task of deciding which information to select for updating users as an event develops. We propose a solution to this problem which incorporates techniques from information retrieval and multi-document summarization and evaluate this approach on a set of historic events using a large stream of news documents. We also introduce an evaluation method which is significantly less expensive than traditional approaches to temporal summarization.

## 1 Introduction

A *time-critical news event* refers to an unexpected news event where information about the topic is rapidly developing. Examples include natural disasters (e.g. earthquakes) and human catastrophes (e.g. airplane crashes). When such an event occurs, local reporters provide information to national and international news agencies, which, in turn, disseminate this information to primary news sources such as TV channels, radio stations, and newspapers. These news sources also depend on citizen journalists, reports from official channels, and on social media, to form a picture of the event, which they then publish. Unfortunately, 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 and crippling of infrastructure, it is harder to collect authoritative news, causing rumors and unsubstantiated information to propagate [7].

Time-critical news events are very important topics for users. Oftentimes, users need information urgently and cannot afford to wait for comprehensive reports to materialize. This is especially true for individuals close to the event, or who have acquaintances there [14].

[⋆] Work conducted while all three authors were at Yahoo! Research New York.

Unfortunately, current solutions do not satisfy users interested in receiving updates about an event. In the context of Twitter, users can either follow specific authoritative accounts (e.g. @BBC) or specific event-related hashtags (e.g. #japanearthquake). However, there is no support for only presenting the user with *novel* content (i.e. new to the user) and updates can suffer from poor coverage, especially for smaller events, and unreliable information. For increased coverage, users can request keyword-based alerts from news aggregators which collate the output of tens of thousands of primary news sources. However, the granularity of clustering is relatively coarse.

In our work, we formalize the problem of online updating for time-critical news events. This task can be seen as a variation of previous work in information retrieval and multi-document summarization. As such, both our problem definition, evaluation, and algorithms are grounded in those results. Nevertheless, we stress that our problem definition, evaluation, and algorithms are significant extensions to previous work. We also note that the corpus used in our experiments is orders of magnitude larger than those found in existing multi-document summarization work; this reflects a real world challenge for this new problem.

## 2 Related Work

*Topic detection and tracking* (TDT) refers to the document-level tasks associated with detecting and tracking news events [1]. Although detecting on-topic news articles is a fundamental part of our problem, we are interested in more granular sub-event decision-making. Allan *et al.* suggested studying the selection of novel and relevant sentences from a stream of news articles [2]. Referring to this task as *temporal summarization*, the authors develop metrics and algorithms to retrospectively select sentences. Although similar to our setting, temporal summarization is also different in two ways. First, our work studies online decision-making, not retrospective summarization. Second, the editorial costs involved in the temporal summarization work are significantly higher than our techniques require. The annotation effort would be impossible for a data set of the size we consider.

Multi-document summarization (MDS) refers to the task of generating a text summary of a pool of documents on the same topic [8]. Our work is most similar to *extractive summarization* where the summary consists of sentences extracted from the pool of documents. Broadly, MDS methods can be classified as unsupervised or supervised. Unsupervised methods score candidate sentences according to a signal believed to be correlated with quality. For example, LexRank is a method for computing sentence importance based on the eigenvector of a graph representation of sentences [5]. Supervised methods score candidate sentences according to a model which directly predicts the evaluation metric. For example, Ouyang *et al.* compute sentence importance by modeling the relationship between sentence features and the target metric [11]. Regardless of the techniques used, almost all previous MDS work has focused on batch or retrospective summarization. Furthermore, experiments were conducted on relatively small document sets where $O(n^2)$ techniques (e.g. graph-based techniques) were acceptable; such approaches are not tractable for our data sets without some modification.

## 3 Problem Definition

A time-critical news event, $e$, is a topic with a clear onset time, $t_e$. A set of profile queries, $\mathcal{Q}_e$, represents the filtering key words which users submit to follow the event. The set of subtopics associated with the event, $\epsilon(e)$, represents the important information that should be included in the updates to deliver to the users (we will discuss our realization of $\epsilon(e)$ in Section 3.2). The system observes a temporally-ordered stream of documents, $[d_1, d_2, \ldots]$. 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 $d_t$ as well as those in an input buffer, $\mathcal{B}_I$, comprised of the most recent $k$ documents.[4] Upon receiving a document at time $t$, the set of delivered updates is $\tilde{\mathcal{S}}_{t-1}$, where $t-1$ is the arrival time of the previous document $d_{t-1}$.[5]

### 3.1 Metrics

To evaluate the overall performance of the online updating system during the time period of interest, we measure the relevance of the set of updates $\tilde{\mathcal{S}}$ delivered to the user using the ground truth subtopics $\epsilon(e)$. In particular, we measure the expected relevance of the delivered updates over time. The first two metrics we propose are *expected precision* and *expected recall.* The precision and recall of an update $s$ measure the quality of the subtopics discussed in $s$ relative to the set of true subtopics $\epsilon(e)$,

$$\mathrm{P}(s) = \frac{|\epsilon(s) \cap \epsilon(e)|}{|\epsilon(s)|}, \qquad \mathrm{R}(s) = \frac{|\epsilon(s) \cap \epsilon(e)|}{|\epsilon(e)|}$$

where $\epsilon(s)$ is the set of subtopics discussed in $s$. The expected precision and recall of the delivered updates is then defined as,

$$\mathrm{E}_s[\mathrm{P}(s)] = \frac{1}{|\tilde{\mathcal{S}}|}\sum_{s\in\tilde{\mathcal{S}}} \mathrm{P}(s), \qquad \mathrm{E}_s[\mathrm{R}(s)] = \frac{1}{|\tilde{\mathcal{S}}|}\sum_{s\in\tilde{\mathcal{S}}} \mathrm{R}(s)$$

[4] Wang *et al.* [13] consider the situation where documents arrive in a stream but requires storing all the input sentences in memory, which may not be feasible when the number of documents is of the scale encountered in web news media. In our work, we explicitly enforce a fixed-sized sentence buffer to respect this real world constraint.

[5] This task can be seen as the online, sequential analog of update summarization where the task is to create a summary given a new batch of documents assuming the information in an existing batch of documents has been consumed [4].

These can be thought of as macro-averaged measures with micro-averaged analogs, $\mathrm{P}(\tilde{\mathcal{S}})$ and $\mathrm{R}(\tilde{\mathcal{S}})$, treating individual updates as a single, long update. We refer to $\mathrm{P}(\tilde{\mathcal{S}})$ and $\mathrm{R}(\tilde{\mathcal{S}})$ as *cumulative precision* and *cumulative recall.*

To evaluate the amount of novel information the system delivers over time, we propose two metrics to measure the expected novelty of the emitted updates. At time $t$, the *incremental precision* of an update $s_t$ measures the fraction of subtopics discussed in $s_t$ that belongs to the set of true subtopics $\epsilon(e)$ but *not* in the set of previously delivered updates $\tilde{\mathcal{S}}_{t-1}$. The *incremental recall* of an update $s$ measures the fraction of true subtopics $\epsilon(e)$ that are discussed in $s$ but *not* in the set of previously delivered updates $\tilde{\mathcal{S}}_{t-1}$. These are defined as,

$$\delta\mathrm{P}(s_t, \tilde{\mathcal{S}}_{t-1}) = \frac{|(\epsilon(s_t) - \epsilon(\tilde{\mathcal{S}}_{t-1})) \cap \epsilon(e)|}{|\epsilon(s_t) - \epsilon(\tilde{\mathcal{S}}_{t-1})|}, \quad \delta\mathrm{R}(s_t, \tilde{\mathcal{S}}_{t-1}) = \frac{|(\epsilon(s_t) - \epsilon(\tilde{\mathcal{S}}_{t-1})) \cap \epsilon(e)|}{|\epsilon(e) - \epsilon(\tilde{\mathcal{S}}_{t-1})|}$$

*Expected incremental precision* and *expected incremental recall* are defined as,

$$\mathrm{E}_t[\delta\mathrm{P}(s_t, \tilde{\mathcal{S}}_{t-1})] = \frac{1}{|\tilde{\mathcal{S}}|}\sum_{t=1}^{|\tilde{\mathcal{S}}|} \delta\mathrm{P}(s_t, \tilde{\mathcal{S}}_{t-1}), \quad \mathrm{E}_t[\delta\mathrm{R}(s_t, \tilde{\mathcal{S}}_{t-1})] = \frac{1}{|\tilde{\mathcal{S}}|}\sum_{t=1}^{|\tilde{\mathcal{S}}|} \delta\mathrm{R}(s_t, \tilde{\mathcal{S}}_{t-1})$$

Finally, to measure the timeliness of the delivered updates, we also consider a metric which promotes performance for early delivery of relevant and novel updates. Specifically, we measure the recall of the delivered updates, $\tilde{\mathcal{S}}_t$, as a function of the time after event onset. Because $\tilde{\mathcal{S}}_t$ is cumulative, this recall curve will monotonically increase with time. As a result, we can compare algorithms according to the area under the recall curve. The integral is taken over a range of 10 days, beginning at $t_e$.

It is worth comparing our problem definition to temporal summarization, the problem previously studied by Allan *et al.* [2]. The temporal summarization metrics bear some resemblance to our own with a few subtle differences. Allan *et al.*, instead of measuring the expectation of $\mathrm{P}(s)$, focus on the expectation of $\lceil \mathrm{P}(s) \rceil$ with their 'u-precision' metric; that is, sentences are considered *relevant* if they contain *any* event subtopics. Similarly, Allan *et al.* focus on the expectation of $\lceil \delta\mathrm{P}(s, \tilde{\mathcal{S}}) \rceil$ with their 'nu-precision' metric. $\mathrm{E}_s[\mathrm{R}(s)]$ is *exactly* the 'nu-recall' metric used in Allan *et al.* The remaining metrics used by Allan *et al.* do not have clear interpretations in our problem definition.

### 3.2 Defining Event Subtopics

As mentioned earlier, to evaluate the quality of a candidate update, the associated subtopics need to be determined. Allan *et al.* use editors to manually define a set of subtopics associated with each event as well as which of these subtopics are associated with each sentence [2]. As the number of events and documents grows, this annotation approach becomes problematic since the annotation effort scales linearly with the number of sentences.

In order to address scaling issues with the annotation approach used by Allan *et al.*, we adopt an evaluation approach inspired by work in multi-document summarization. For each event, we define a *target summary*, $\mathcal{S}^*$. The target summary is a retrospective, manually written description of the event. Because it is retrospective, we can assume that the author(s) have full knowledge of the accuracy and importance of different subtopics of the event. Instead of explicitly defining subtopics for an event, we can consider each $n$-gram in $\mathcal{S}^*$ as an individual subtopic. Although this may seem awkward, this is the principle underlying the Rouge metric used in multi-document summarization [6]. In fact, under this definition of $\epsilon(s)$, our precision and recall metrics are precisely the Rouge metrics. As is common in the MDS literature, we adopt unigram-based evaluation although, in theory, we could extend our results to higher order $n$-grams.

Using subtopics derived from target summary $n$-grams is attractive for several reasons. First, our editorial cost does not increase as we consider more documents. If editors explicitly define the subtopics for an event, as was done by Allan *et al.*, then they also have to assign these subtopics to sentences manually. Because our approach defines subtopics using only the text of the gold standard summaries and because all sentences naturally have $n$-grams defined, we incur no additional annotation cost if we wish to consider more or different documents. Second, target summaries, especially for relatively large events, can be found freely available on the Web. In our work, we use Wikipedia-based target summaries for our events, as Wikipedia has provided target summaries for previous work in multi-document summarization [3, 12].

In order to support our decision to represent subtopics in this way, we conducted a Mechanical Turk study to confirm the association between the precision $\mathrm{P}(s)$ of a delivered update $s$ and human judgments. In our study, 600 sentences were randomly selected from the references of a Wikipedia article to represent a pool of candidate updates about an event. Workers were presented with pairs of randomly selected sentences with similar lengths and were asked to determine which sentence of each pair was more appropriate to be delivered as an update. Each pair of the updates was assigned to 5 workers to judge. For each pair of the updates, $s_i$ and $s_j$, we computed the correlation between $\mathrm{P}(s_i) - \mathrm{P}(s_j)$ and the number of workers who pick the second sentence in the pair. Because difference in P is continuous while the the voting of workers is discrete, in addition to measuring the Pearson correlation, we compute the polyserial correlation [10]. We observed a $\rho$ of 0.45 ($p < 10^{-15}$) using polyserial correlation and a $\rho$ of 0.43 ($p < 10^{-12}$) with Pearson correlation. This is consistent with levels of association strength reported in the MDS literature [6].

## 4 AGK Model

Allan *et al.* model $\mathrm{E}_s[\mathrm{P}(s)]$ and $\mathrm{E}_t[\delta\mathrm{P}(s_t, \tilde{\mathcal{S}}_{t-1})]$ by decomposing the task as modeling the sentence-wise measures, $\lceil \mathrm{P}(s) \rceil$, and $\delta\mathrm{P}(s, \tilde{\mathcal{S}})$, respectively. We refer to this as the AGK model. After predicting $\lceil \mathrm{P}(s) \rceil$ and $\delta\mathrm{P}(s, \tilde{\mathcal{S}})$, sentences are selected greedily to achieve effective performance. We point out that the authors study signals which are rank-correlated with $\lceil \mathrm{P}(s) \rceil$ and $\delta\mathrm{P}(s, \tilde{\mathcal{S}})$ and do not explicitly model the relationship between their predictors and the target metrics.

Modeling $\lceil \mathrm{P}(s) \rceil$ involves predicting the *relevance* of $s$. Because the system does not have access to relevant sentences at runtime, the AGK model uses a pseudo-relevant sentence, $\tilde{s}$, in the same spirit that document retrieval systems use pseudo-relevant documents for feedback. The authors propose two methods for generating the pseudo-relevant sentence. First, the authors hypothesize that concatenating all of the sentences in $\mathcal{B}_I$, under the assumption that the language in the input buffer is sufficiently filtered and consistent, provides a good relevance model. As an alternative, the authors hypothesize that, if $s$ is extracted from $d$ and we assume $d$ is relevant, then concatenating all of sentences in $d$ might also provide a good relevance model. Given $s$ and $\tilde{s}$, a system can compute the relevance of $s$ using any text similarity measure.

Modeling $\delta\mathrm{P}(s, \tilde{\mathcal{S}})$ involves predicting the *novelty* of $s$ with respect to $\tilde{\mathcal{S}}$. Again, a system can compare $s$ and $\tilde{\mathcal{S}}$ using any text similarity measure. Specifically, the AGK model uses the geometric mean *dissimilarity* to all sentences in $\tilde{\mathcal{S}}$. The AGK model considers $s$ as novel if it is dissimilar to *every* sentence in $\tilde{\mathcal{S}}$.

## 5 Proposed Model

As with the AGK model, we model sentence precision and incremental sentence precision. However, we propose directly modeling $\mathrm{P}(s)$ and $\delta\mathrm{P}(s, \tilde{\mathcal{S}})$. Moreover, instead of using individual predictors and a rank correlation, we propose learning the relationship between a large number of signals, known as *features*, likely to be correlated with our targets. As a result, our model can be considered a generalization of the AGK model and the MDS literature.

Let $\phi$ be the features associated with the candidate sentence, $s$. Features can be thought of as properties of a sentence likely to be indicative of our prediction target, either sentence precision or incremental sentence precision. For example, we might believe that earlier sentences in a document are more relevant; therefore, the integer-valued position of a sentence in the document would be considered as a feature. Similarly, any of the predictors in the AGK model can be considered as a feature. A detailed discussion of features follows in Section 6.

We adopt a linear model of $\mathrm{P}(s)$ and $\delta\mathrm{P}(s, \tilde{\mathcal{S}})$. That is, given a vector of features values, $\phi$, of a candidate sentence, our model is represented by a vector of weights, $\theta$. A sentence is scored by the inner product, $\langle \theta, \phi \rangle$. Because we have two models, we have one set of parameters for each target, $\mathrm{P}(s)$ and $\delta\mathrm{P}(s, \tilde{\mathcal{S}})$. Given a set of example sentences and target values, we find the values of our parameters that minimize the squared loss between the predictions and the targets. Specifically, we will learn the values of these parameters using training data of the form, $\{\langle \phi^s, y^s \rangle\}$ where $y$ is the target value, either $\mathrm{P}(s)$ or $\delta\mathrm{P}(s, \tilde{\mathcal{S}})$.

Our algorithm uses cascaded predictions for each candidate sentence. We first make a prediction of the sentence precision. Sentences with a predicted value below $\tau^{\tilde{\mathrm{P}}}$ are filtered out of consideration. This ensures that sentences considered have a high percentage of relevant sub-events. We then make a prediction of the incremental precision for each of the remaining sentences. Sentences with a predicted value above $\tau^{\delta\tilde{\mathrm{P}}}$ are added to $\tilde{\mathcal{S}}_{t-1}$ delivered to the user as an update.

The values of $\tau^{\tilde{\mathrm{P}}}$ and $\tau^{\delta\tilde{\mathrm{P}}}$ are learned on a separate validation set. We omit the specifics of the training algorithm due to space constraints.

## 6 Features

As stated earlier, features of the candidate update sentence play an important part in our models. We define features likely to be correlated with our targets, $\mathrm{P}(s)$ and $\delta\mathrm{P}(s, \tilde{\mathcal{S}})$. When possible, we incorporate concepts from temporal summarization and multi-document summarization. We divide our features into *stationary features*, those which do not change as new documents are observed, and *non-stationary features*, those which change as new documents are observed. For each of these types of features, we consider both sentence-level features and document-level features.

Stationary sentence-level features consider only basic characteristics of the sentence independent of other documents in the stream. Simple features include the sentence position, the length of the sentence, and the presence of patterns such as numbers, capitalization, and temporal expressions. More complicated sentence-level features consider the similarity to other content in the document. At a coarse level, we can consider comparing a sentence $s$ to the words present in its source document $d$. To accomplish this, we adapted the frequency-based $Sum_{CF}$ algorithm [9]. While these scores provide a comparison of $s$ to $d$ as a whole, we also derived features that describe the novelty of $s$ compared to individual sentences in $d$ by measuring maximum and average similarity to other sentences in the document. This feature is inspired by the novelty signal in the AGK model.

Stationary document-level features are the same for all constituent sentences. These include the BM25F retrieval score of the document, the number of sentences in the document, the fraction of query terms in the document, as well as pattern-matching features.

We consider two types of non-stationary features, those based on the contents of $\mathcal{B}_I$ and those based on the contents of the delivered updates $\tilde{\mathcal{S}}_{t-1}$. Input buffer features are likely to help with predicting the relevance score $\mathrm{P}(s)$. Output buffer features are likely to help with detecting redundancy and therefore are likely to help with predicting the novelty score $\delta\mathrm{P}(s_t, \tilde{\mathcal{S}}_{t-1})$.

There are several ways to predict $\mathrm{P}(s)$ given the contents of $\mathcal{B}_I$. The most straightforward way is to use the sumbasic features described earlier. In this case, we use statistics from $\mathcal{B}_I$ rather than $d$. We can also compute AGK novelty features based on sentences in $\mathcal{B}_I$ rather than $d$. Several algorithms in the MDS literature use graphs based on inter-sentence text similarity to detect important sentences [5]. In order to derive a graph from text similarity, the authors often threshold scores to establish edges between sentences. Graph-based properties of the sentences (e.g. LexRank, degree) can be used to infer the importance of a sentence. We incorporate these measures, constructing our graphs based on the contents of $\mathcal{B}_I$.

Sentence-level features based on $\tilde{\mathcal{S}}_{t-1}$ are, in principle, similar to those based on $\mathcal{B}_I$. At a coarse level, we compare $s_t$ to a centroid based on all sentences in $\tilde{\mathcal{S}}_{t-1}$. We also compute the average and maximum term-based similarity to all sentences in $\tilde{\mathcal{S}}_{t-1}$. Finally, we also consider the average and minimum difference in timestamps between $s_t$ and sentences in $\tilde{\mathcal{S}}_{t-1}$. Again, these features are based on those found in the AGK model.

Non-stationary document-level features include the age of the document compared to the time of decision-making (recall that older sentences remain in $\mathcal{B}_I$ and are candidates for presentation). We also include a document-level computation of novelty based on sentences in $\mathcal{B}_I$.

## 7 Methods and Materials

We defined a set of 197 time-critical news events based on pages in Wikipedia classified as referring to natural and human disasters (e.g. earthquakes, airliner accidents) which occurred between August 2009 and April 2011. For each event $e$, we derived the set of profile queries $\mathcal{Q}_e$ from the Wikipedia titles redirecting to the event page. For each query in $\mathcal{Q}_e$, we selected all documents which matched more than 60% of the query terms from Yahoo! News, a corpus consisting of syndicated and crawled feeds from news sources including both local news providers and international news agencies. Documents were only considered if their publication date occurred between the event onset time and ten days thereafter. Ordering these documents by publication time provided us with our stream of source documents. After splitting documents into sentences, a total of 811,582,157 sentences were extracted for the entire set of events. We note that the scale of our data is orders of magnitude larger than the dataset used by Allan *et al.* (22 events and a total of 17,049 sentences).

We divide our set of events into training and testing sets. We hold out a set of 27 events for testing. Our model requires independent training of three components: the $\mathrm{P}(s)$ model, the $\delta\mathrm{P}(s, \tilde{\mathcal{S}})$ model, and the $\langle \tau^{\tilde{\mathrm{P}}}, \tau^{\delta\tilde{\mathrm{P}}} \rangle$ thresholds. The remaining 152 training events are divided to train these different components: 70 for training and validation of $\mathrm{P}(s)$ model parameters, 65 for training and validation of $\delta\mathrm{P}(s, \tilde{\mathcal{S}})$ model parameters, and 17 for tuning $\langle \tau^{\tilde{\mathrm{P}}}, \tau^{\delta\tilde{\mathrm{P}}} \rangle$ thresholds.

We evaluate performance of the final delivered updates according to the expected precision/recall and incremental precision/recall as well as cumulative precision and recall measures. Gold standard summaries are derived from a version of the Wikipedia summary downloaded in Spring of 2011. We evaluate our performance on each event-query pair, $\langle e, q \rangle$, then aggregate first over all queries in an event and then aggregating across events. We perform event-level aggregation to prevent events with many queries from dominating the evaluation. Statistical significance is measured using the Student's $t$-test paired on event.

We consider two methods for constructing candidate sets of sentences for a document. We expect sentences earlier in the document to be more likely to contain novel information. A conservative approach would consider only the titles from documents in the stream as candidates for presentation to users. Titles are attractive because they are, by design, intended to communicate the most important information in the document. At the same time, considering only titles reduces the memory requirements of $\mathcal{B}_I$. However, there may be better sentences further in the document. Therefore, our second candidate pool considers the first ten sentences of the document, including the title.

In addition to exploring different candidate pools, we consider different subsets of features. In the first, we used only the stationary sentence and document features. This can be considered as an ensemble of buffer-insensitive MDS signals. This can also be thought of as an approach that maintains no input buffer. Our second feature subset only used non-stationary sentence and document features. This can be considered as an ensemble of buffer-sensitive MDS signals, including graph-based scores such as LexRank and buffer-sensitive sumbasic scores. Finally, we also consider using *all* features in aggregate. We attempted to use individual MDS and temporal summarization features as predictors but found that the performance was much worse to those using groups of stationary and non-stationary MDS and temporal summarization features.

## 8 Results

We present results for our algorithms in Table 1. Individual state-of-the-art predictors from temporal summarization and MDS techniques which combined evidence and were omitted due to poor performance.

**Table 1.** Experimental Results. Bold values indicate the best run for the evaluation measure. Superscripts indicate significant differences ($p < 0.05$) between the run and competing algorithms within each subtable (s: stationary; n: non-stationary; a: all).

(a) Sentences selected from the first 10 sentences of the document

| features | $\mathrm{E}_s[\mathrm{P}]$ | $\mathrm{E}[\delta\mathrm{P}]$ | $\mathrm{P}(\tilde{\mathcal{S}})$ | $\mathrm{E}_s[\mathrm{R}]$ | $\mathrm{E}[\delta\mathrm{R}]$ | $\mathrm{R}(\tilde{\mathcal{S}})$ | $\mathrm{AUC}_{\mathrm{R}(\tilde{\mathcal{S}})}$ |
|---|---|---|---|---|---|---|---|
| stat. | 0.4468$^{\mathrm{n,a}}$ | 0.2144$^{\mathrm{a}}$ | 0.2156$^{\mathrm{a}}$ | 0.0101$^{\mathrm{n}}$ | 0.0041$^{\mathrm{a}}$ | 0.2894$^{\mathrm{a}}$ | 0.2532$^{\mathrm{n,a}}$ |
| non-stat. | 0.5282$^{\mathrm{s}}$ | 0.2855$^{\mathrm{a}}$ | 0.2814$^{\mathrm{a}}$ | **0.0163**$^{\mathrm{s}}$ | 0.0056$^{\mathrm{a}}$ | 0.2846$^{\mathrm{a}}$ | 0.2521$^{\mathrm{s,a}}$ |
| all | **0.5548**$^{\mathrm{s}}$ | **0.4136**$^{\mathrm{s,n}}$ | **0.4129**$^{\mathrm{s,n}}$ | 0.0133 | **0.0128**$^{\mathrm{s,n}}$ | **0.3496**$^{\mathrm{s,n}}$ | **0.3034**$^{\mathrm{s,n}}$ |

(b) Sentences selected from title of the document

| features | $\mathrm{E}_s[\mathrm{P}]$ | $\mathrm{E}[\delta\mathrm{P}]$ | $\mathrm{P}(\tilde{\mathcal{S}})$ | $\mathrm{E}_s[\mathrm{R}]$ | $\mathrm{E}[\delta\mathrm{R}]$ | $\mathrm{R}(\tilde{\mathcal{S}})$ | $\mathrm{AUC}_{\mathrm{R}(\tilde{\mathcal{S}})}$ |
|---|---|---|---|---|---|---|---|
| stat. | 0.5400$^{\mathrm{n}}$ | 0.3004$^{\mathrm{a}}$ | 0.2950$^{\mathrm{a}}$ | 0.0123$^{\mathrm{n}}$ | 0.0062$^{\mathrm{a}}$ | **0.3233**$^{\mathrm{n,a}}$ | **0.2810**$^{\mathrm{n,a}}$ |
| non-stat. | 0.4546$^{\mathrm{s,a}}$ | 0.2272$^{\mathrm{a}}$ | 0.2340$^{\mathrm{a}}$ | 0.0097$^{\mathrm{s,a}}$ | 0.0052$^{\mathrm{a}}$ | 0.2549$^{\mathrm{s}}$ | 0.2143$^{\mathrm{s}}$ |
| all | **0.5459**$^{\mathrm{n}}$ | **0.4097**$^{\mathrm{s,n}}$ | **0.4067**$^{\mathrm{s,n}}$ | **0.0132**$^{\mathrm{n}}$ | **0.0102**$^{\mathrm{s,n}}$ | 0.2772$^{\mathrm{s}}$ | 0.2425$^{\mathrm{s}}$ |

Stationary features alone did not seem to provide consistently strong performance. Stationary features were marginally better than non-stationary features when using only titles, and achieved statistically significant improvements for sentence-level precision and recall, and cumulative recall measures. We observed weaker performance when comparing stationary features alone to the combined feature set with title candidates; the exception again being the cumulative recall measures. Most of relative effectiveness of stationary features disappeared when we expanded the candidate pool to include the first 10 sentences. However, when comparing the stationary feature runs which selected titles to stationary feature runs which selected from the first 10 sentences, *none* of the performance differences were statistically significant, suggesting that the degradation resulted from improvements to the competing algorithms.

Non-stationary features alone consistently performed worst. The only experimental condition where it performed best was on sentence-level recall; here, the performance was only statistically significant compared to using stationary features alone. Nevertheless, when comparing non-stationary runs which selected titles to non-stationary runs which selected from the first 10 sentences, we observed statistically significant improvements for sentence-level precision and recall, and the area under the cumulative recall curve, suggesting that, even though the performance relative to other feature sets was weak, the addition of candidates improved performance compared to smaller candidate pools.

Combining all features performed strongest across almost all experimental conditions. Notably, this model underperforms when evaluating according to the recall of $\tilde{\mathcal{S}}$ selecting from titles alone. This weak performance disappears when selecting from the first 10 sentences. When comparing the runs which selected titles to runs which selected from the first 10 sentences, the *only* statistically significant performance difference was on the cumulative recall measures, suggesting that the gains resulted from an improvement in the performance of combined feature run rather than the degradation of competing runs.

Since we use linear predictors with normalized features for our models,we can inspect the magnitude of the weights, $\theta^{\tilde{\mathrm{P}}}$ and $\theta^{\delta\tilde{\mathrm{P}}}$, to see which features contributed most to our predictions. We have a total of six pairs of models for each of our experimental conditions. Due to space constraints, we only present analysis for weights of our models which considered all features. For $\theta^{\tilde{\mathrm{P}}}$ in the title candidate condition, the highest weights were assigned to sumbasic features, the document BM25F score, and the non-stationary novelty features. In the case of the first ten sentence candidate condition, the highest weights were assigned to the same features and also included document age, sentence position, and the LexRank score. For $\theta^{\delta\tilde{\mathrm{P}}}$ in the title candidate condition, the highest weights were assigned to sumbasic features, the temporal and term-based similarity to sentences in $\tilde{\mathcal{S}}_{t-1}$, and the document age. In the case of the first ten sentence candidate condition, the highest weights were assigned to document age, the size of $\tilde{\mathcal{S}}_{t-1}$, and term-based similarity to the sentences in $\tilde{\mathcal{S}}_{t-1}$.

## 9 Discussion

At a high level, the results of our experiments suggest that particular attention must be paid to buffering policies. In particular, techniques which do not use a buffer (i.e. only stationary features) under-perform those which add buffer features (i.e. all features). Buffer features alone, though, are not enough especially if we are conservative about the sentences under consideration (Table 1(b)).

The strong performance when using stationary features in terms of our cumulative recall measures (when selecting from titles) might be explained by the fact that the advantage of non-stationary features is limited by the size of the input buffer. When using titles alone, the number of sentences in $\mathcal{B}_I$ is small, leading to less accurate language and graph models. Recall that the gains achieved by the stationary feature model disappeared when we moved to considering the first 10 sentences because the models which used non-stationary features improved. This explanation is supported by the higher weight to LexRank and centrality features in the models based on candidates from first 10 sentences compared to those which only consider the titles. Also, we suspect that the difference in performance is most likely confined to recall-oriented metrics because our models are heavily tuned toward precision since we are modeling $\mathrm{P}(s)$ and $\delta\mathrm{P}(s, \tilde{\mathcal{S}})$.

The values of $\mathrm{E}_s[\mathrm{R}(s)]$ and $\mathrm{E}_t[\delta\mathrm{R}(s_t, \tilde{\mathcal{S}}_{t-1})]$ are consistently low across experimental conditions. Since our target summaries are lengthy documents, the denominators of these metrics will tend to be large compared to the relatively short sentences. In fact, one concern we have is that the variance in the length of the target summaries may be influencing our recall oriented metrics.

It is worth reflecting on the relationship between our measures and the behavior with respect to the user. From the user's perspective, if each delivered update contains a fair amount of new content, then less time is spent reviewing redundant information. As a result, it would be fair to say that $\mathrm{E}[\delta\mathrm{P}]$ is one important metric when considering user experience. One potential caveat with $\mathrm{E}[\delta\mathrm{P}]$ is that later sentences may tend to have higher $\mathrm{E}[\delta\mathrm{P}]$ because subtopics (terms) already in $\tilde{\mathcal{S}}$ are subtracted from the denominator of this metric. This is important since the set difference is in terms of all subtopics, including those that are not in $\mathcal{S}^*$; 'irrelevant subtopics' may be removed and inflate the quality of $s$. This concern can be allayed by inspecting algorithm performance in terms of $\mathrm{E}_s[\mathrm{P}]$, where the denominator is independent of subtopics in $\tilde{\mathcal{S}}$. For our best performing run, we maintain a high expected precision, implying that, on average, more than half of each update contains content in the target summary.

In addition to receiving relevant and novel sentences, users are also interested in topical coverage which we measure with our recall-oriented measures. As we discussed earlier, the sentence-level recall measures, $\mathrm{E}_s[\mathrm{R}]$ and $\mathrm{E}[\delta\mathrm{R}]$, are problematic because of the large difference between sentence and target summary lengths. The cumulative recall metric provides some insight into the comprehensiveness of our delivered sentences. However, our best performing runs tended to only capture roughly a quarter of the target summaries on average. This could result from several factors. First, our data only included documents up to 10 days after the event onset while the target summaries we used were gathered from Wikipedia, potentially, more than a year after the event onset. We suspect that the absolute $\mathrm{R}(\tilde{\mathcal{S}})$ numbers would improve if we used a snapshot of the Wikipedia summary 10 days after the event. To address this issue, we plan on investigating the effect of using different versions of Wikipedia articles in future work. Second, there may be a vocabulary mismatch between Wikipedia and print news. Although this is an issue for any approach based on a target summary, we may be able to represent summaries with retrospective news analyses.

In order to measure the timeliness of delivered sentences, we included the $\mathrm{AUC}_{\mathrm{R}(\tilde{\mathcal{S}})}$ measure. The ordering of algorithms tends to be consistent with the ordering based on the final cumulative summary, $\mathrm{R}(\tilde{\mathcal{S}})$, suggesting algorithms which are strong shortly after the event onset tend to be strong later on.

## 10 Conclusion

We have introduced the online updating problem for time-critical news events, described and verified a scalable evaluation method, and demonstrated the effectiveness of the proposed algorithms. While rooted in prior work in information retrieval and text summarization, the online nature of our task makes our problem very different and unique.

## References

1. J. Allan, editor. *Topic Detection and Tracking.* Springer, 2002.
2. J. Allan, R. Gupta, and V. Khandelwal. Temporal summaries of new topics. In *Proceedings of the 24th ACM SIGIR*, pages 10–18, 2001.
3. N. Balasubramanian and S. Cucerzan. Topic pages: An alternative to the ten blue links. In *Fourth IEEE International Conference on Semantic Computing*, 2010.
4. H. T. Dang and K. Owczarzak. *Overview of the TAC 2008 Update Summarization Task*, pages 1–16. 2008.
5. G. Erkan and D. R. Radev. Lexrank: graph-based lexical centrality as salience in text summarization. *J. Artif. Int. Res.*, 22:457–479, December 2004.
6. C.-Y. Lin. Rouge: A package for automatic evaluation of summaries. In S. S. Marie-Francine Moens, editor, *Text Summarization Branches Out: Proceedings of the ACL-04 Workshop*, pages 74–81, July 2004.
7. 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*, 2010.
8. A. Nenkova and K. McKeown. Automatic summarization. *Foundations and Trends in Information Retrieval*, 5(2-3), 2011.
9. A. Nenkova, L. Vanderwende, and K. McKeown. A compositional context sensitive multi-document summarizer. In *Proceedings of the 29th ACM SIGIR*, 2006.
10. U. Olsson, F. Drasgow, and N. Dorans. The polyserial correlation coefficient. *Psychometrika*, 47(3):337–347, September 1982.
11. Y. Ouyang, W. Li, S. Li, and Q. Lu. Applying regression models to query-focused multi-document summarization. *Info. Processing and Management*, 47(2), 2011.
12. C. Sauper and R. Barzilay. Automatically generating wikipedia articles: a structure-aware approach. In *ACL 2009*, pages 208–216, 2009.
13. D. Wang and T. Li. Document update summarization using incremental hierarchical clustering. In *Proceedings of the 19th ACM CIKM*, 2010.
14. 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 ACM SIGIR*, 2011.