Industrial Paper

# WILSON: A Divide and Conquer Approach for Fast and Effective News Timeline Summarization


Yiming Liao
The Pennsylvania State University
State College, PA, US
yiming@psu.edu

Shuguang Wang
The Washington Post
Washington, D.C., US
shuguang.wang@washpost.com

Dongwon Lee
The Pennsylvania State University
State College, PA, US
dongwon@psu.edu



## ABSTRACT

Major news media frequently uses the method of *news timeline summarization* to summarize important daily news over major events across the timeline. While various sophisticated methods have been proposed to generate both concise and complete news timelines, in practice, generating timelines from a large number of news articles not only faces quality issues but also encounters the challenge of generation speed, which all existing methods have neglected. To mitigate these issues, in this work, we propose to speed up timeline generation by dividing the whole summarization task into sub-summarization tasks, adopting the "divide and conquer" philosophy: (1) date selection and (2) text summarization.

Furthermore, since existing methods in news timeline summarization pay less attention to the date selection than text summarization, in this paper, we re-examine the role of date selection in news timeline summarization and demonstrate that accurate date selection "alone" can significantly contribute to the task of news timeline summarization. Leveraging on the explicit date selection, then, we propose a simple yet fast and effective news timeline summarization method, named WILSON (neWs tImeLine SummarizatiON). Experimented on two widely used timeline summarization benchmark datasets, *timeline17* and *crisis*, empirical evaluation shows that WILSON outperforms state-of-the-art approaches in both speed and ROUGE scores, significantly improving ROUGE-2 F1 scores by 9.5%~17.7% and reducing generation time by two orders of magnitude. A further user study with professional journalists also validates the superiority of WILSON. Finally, we build a real-time news timeline summarization system and achieve encouraging results on an industrial-level corpus.


## 1 INTRODUCTION

Along with the rapid development of web services, an increasing number of news articles are published daily, describing both major and minor events worldwide. Due to the tremendous amount of news articles being produced every day, readers easily get lost in this information flood. Fortunately, *news timeline*, which summarizes each event with primary messages in a chronological order, makes it easy for readers to gain key insights and understand the evolution of news events. As such, many major news media has adopted the idea and have frequently produced news timelines of major news events. For example, Table 1 describes how 2018 North Korea-United States Singapore summit finally became a reality. Note that as the example illustrates, creating a news timeline requires the resolution of two sub-problems: (1) choosing of an ideal number of days among hundreds or thousands of candidate days, and (2) generating succinct text summaries per days.

| |
|---|
| Feb. 25, 2018<br>North Korea is "willing to have talks" with the United States, South Korea says, as the PyeongChang Winter Olympics close in a burst of fireworks and diplomacy. |
| Mar. 8, 2018<br>Trump agrees to meet Kim for talks, an extraordinary development after months of heightened tension. Kim commits to stopping nuclear and missile testing. |
| Mar. 27, 2018<br>Kim makes a clandestine visit to Beijing to discuss the negotiations with South Korea and the United States. |
| ... |
| Jun. 1, 2018<br>Trump says the summit will take place June 12 as planned. During a visit to the White House, a North Korean hands Trump a large letter from Kim. |

**Table 1: An example timeline by The Washington Post about the summit between United States and North Korea.**[1]

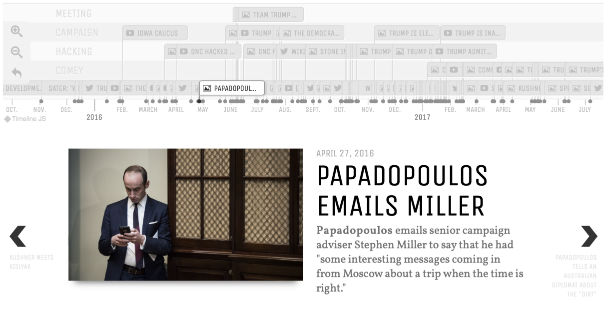
### 1.1 Industrial Use Case

Combined with visual or interactive interfaces, news timelines can provide a convenient way to compress overloaded news to audience. Figure 1 illustrates two real timeline example on two major US newspapers. Figure 1 (a) is an interactive timeline summarization about Trump-Russia investigation from The Washington Post, while Figure 1 (b) is a text-based timeline summarization about China-US Trade War from The New York Times. To help readers better understand the evolution of each news event, journalists take time to collect and organize related news articles, figure out major events and story lines, and "manually" summarize them in a chronological order. As events such as natural disasters and political issues can span from several months to multiple years and involve thousands of news articles, such a manual process cannot scale well. As this process is both time-consuming and labor-intensive, currently, despite the popularity of the concept, not all newspapers are able to quickly produce such news timelines.

To address this challenge, several automatic news timeline summarization methods have emerged in recent years [4, 12, 21, 22, 25, 27, 29]. By and large, there are mainly two categories of news timeline generation methods. One is aimed at separating



[1] https://www.washingtonpost.com/graphics/2018/national/trump-kim-jong-un-timeline/




(a) Trump-Russia investigation (The Washington Post)


(b) China-US Trade War (The New York Times)

**Figure 1: News timeline summarization examples on major news media**

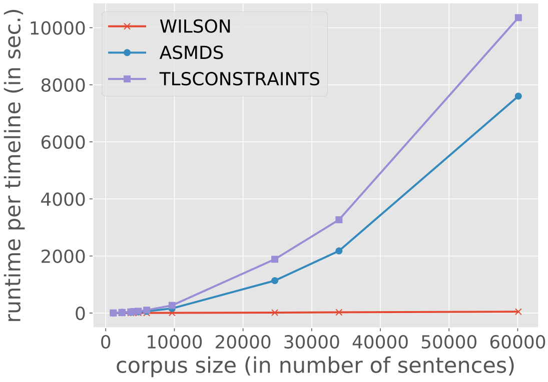



**Figure 2: Running times over varying corpus sizes using *Timeline*17 and *Crisis*. ASMDS and TLSCONSTRAINTS are the two state-of-the-art methods that use submodular framework. Note that, as both of them are not sufficiently scalable to the large corpus, we followed [12] to use a reduced corpus here.**

different stories from a whole news corpus, such as using variants of topic modeling [8, 31] and neural networks [30]. Another category focuses on generating a series of chronological summaries for one specific event from only relevant news articles [12, 22, 28], where the first categories can serve as pre-processing to find relevant news articles for each event. In this paper, our focus is on the latter category in an unsupervised manner.

However, majority of existing methods focus only on the quality of generated timelines and neglect the generation speed. For example, the state-of-the-art unsupervised approach adopts submodular framework [12] and requires the pairwise similarities for all tokenized sentences, which could be over 100,000 per timeline. This yields extremely slow running time, as clearly demonstrated in the comparison of running times in Figure 2. As the compression rates of timeline summarization vary with events and journalists may not know the exact value beforehand, iterative trials with different values are necessary, which makes faster timeline generation even more important. Therefore, in this work, we are greatly motivated by real industrial use cases to speed up the timeline generation by dividing the whole summarization tasks into multiple small summarization tasks by date separation.

News timelines are composed of both salient dates and daily summaries, but previous studies mainly focus on modeling relationships among article contexts while paying less attention to date selection. For example, some models [14, 24, 26, 27] just treat date information the same as text information and include it as one of the features, while others [4, 19] simply use date frequency to resolve events. Although simply modeling text correlation shows good performance on both timeline summarization and date selection [12], it is not clear how date selection will contribute to news timeline summarization. In addition, existing state-of-the-art unsupervised approaches mostly include global optimization, which helps daily summaries to be relevant to the topic. However, using global optimization also makes daily summaries less specific per each day and very time-intensive to generate timelines. Therefore, considering both the quality and speed of news timeline summarization, this paper makes the following main contributions:

(1) We re-examine the role of date selection in timeline summarization and show that, even without considering contextual correlation across different dates, accurate date selection is sufficient to generate high-quality news timelines. More importantly, although ignoring contextual correlation across dates leads to a lower empirical upper bound than other models, all of the previous approaches still fail to reach this lower upper bound, and they are not even close.
(2) Leveraging the explicit date selection, we propose a simple but fast and effective unsupervised news timeline summarization method, named WILSON. Experimented on two widely used timeline summarization datasets, WILSON outperforms state-of-the-art approaches in both speed and ROUGE scores, significantly improving ROUGE-2 F1 score by 9.5%~17.7% and reducing generation time by two orders of magnitude.
(3) To our best knowledge, WILSON is the first work to include an evaluation by professional journalists in news timeline summarization. Through manually comparing the machine-generated news timelines with corresponding human-generated ones, journalists confirm that our approach produces better timelines than competing methods.
(4) Based on the proposed WILSON, we build a real-time news timeline summarization system on an industrial-level news corpus.

## 2 THE PROPOSED METHOD: WILSON

In this section, we introduce our proposed method, named WILSON (neWs tImeLine SummarizatiON), also illustrated in Figure 3. Besides the pre-processing modules such as temporal tagging

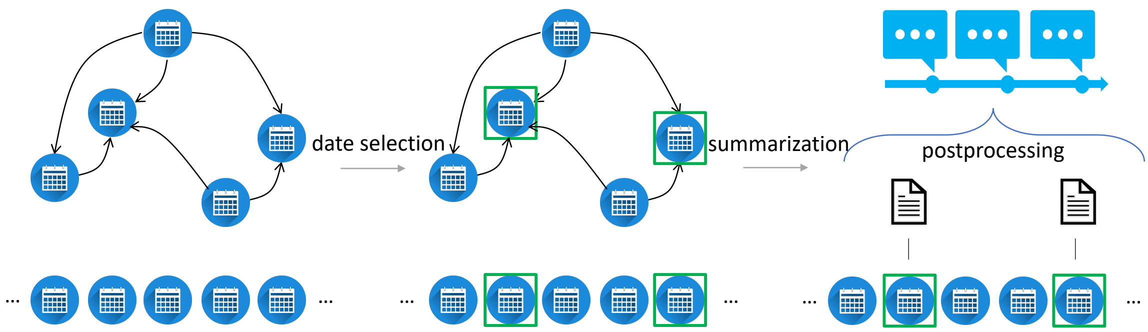



**Figure 3: Workflow of our proposed method**–WILSON.

and search engine indexing, WILSON mainly consists of two components – explicit date selection and text summarization for each selected date.

## 2.1 Problem Formulation

A news timeline can be viewed as a series of chronologically ordered daily summaries over main events, denoted by $(d_i, S_i)$, where $d_i$ and $S_i$ stand for $i_{th}$ date and $i_{th}$ summary. Thus, news timeline summarization can be formulated as:

DEFINITION 1 (NEWS TIMELINE SUMMARIZATION). *Given a corpus of articles $C_q$, which is associated with a topic query $q$ and a time window $[t1, t2]$, the process of automatic timeline generation is to produce a series of daily summaries $(d_1, S_1), ..., (d_T, S_T)$, where $t1 \leq d_i \leq t2$.*

For both readability and reliability of generated news timelines, we follow existing works and utilize extractive summarization, which directly selects sentences from the corpus as summaries. More specifically,

DEFINITION 2 (EXTRACTIVE NEWS TIMELINE SUMMARIZATION). *Given a corpus of articles $C_q$, which is associated with a topic query $q$ and a time window $[t1, t2]$, the corpus is first tokenized to dated sentences $\{(date_i, sentence_i) | date_i \in [t1, t2], sentence_i \in C_q\}$ by a date expression in the sentence and/or by the publication date, then the timeline generation is to produce a series of daily summaries $(d_1, S_1), ..., (d_T, S_T)$, where $t1 \leq d_i \leq t2$ and $S_i = (sentence_{i1}, ..., sentence_{iN})$.*

The number of selected dates $T$ and sentences $N$ are hyper-parameters and chosen by users to control the compression rate of the generated timelines. Date selection is evaluated by f1 scores and summaries are evaluated by ROUGE [10].

## 2.2 Date Selection

We use HeidelTime [20] to tag temporal expressions in sentences during pre-processing stage and start with an unsupervised date selection algorithm [23] to select the most salient dates: (1) we build a date reference graph with dates as nodes and reference relationships as edges; (2) then, we run the PageRank algorithm [16] on the graph and select the top $T$ ranked nodes as the most salient dates. Date references refer to the sentences $s_{ij}$ that are published on $date_i$ while mentioning $date_j$. We experiment with 4 types of edge weights as follows:

- W1: the number of reference sentences $|s_{ij}|$
- W2: temporal distance $|date_j - date_i|$ in days
- W3: W1 $*$ W2, which considers both frequency and temporal distance.
- W4: We adopt BM25 [18] to estimate the relevance of sentences to the query, and use max $BM25(s_{ij}, q)$ as edge weight for each reference.

| Edge Weight | Date F1 | Rouge-1 F1 | Rouge-2 F1 |
|---|---|---|---|
| timeline17 | | | |
| W1 | 0.5512 | 0.3905 | 0.0969 |
| W2 | 0.5528 | 0.4029 | 0.1002 |
| W3 | 0.5628 | 0.4009 | 0.0995 |
| W4 | 0.5068 | 0.3934 | 0.0934 |
| crisis | | | |
| W1 | 0.3022 | 0.3476 | 0.0715 |
| W2 | 0.2838 | 0.3604 | 0.0715 |
| W3 | 0.2710 | 0.3575 | 0.0738 |
| W4 | 0.2925 | 0.3509 | 0.0726 |

**Table 2: Performance of different edge weights**

For example, considering $date_i$=2018-06-01, $date_j$=2018-06-12, and $s_{ij}$ composed of only two sentences, i.e. *Trump says summit with North Korea will take place on June 12* and *The summit will take place on June 12.* Then, W1 is the number of sentences and equals 2, while W2 is the difference between 2018-06-01 and 2018-06-12 in days and equals 11. Accordingly, W3 equals W1 $*$ W2 and is 22. For W4, we treat each sentence as a document, use topic query $q$ to score each document with BM25, and take the maximum BM25 score as W4.

As Table 2 shows that all four edge weights yield comparable results, date reference relationship alone can extract as accurate date selections as topical information. Since constructing topical relationships across dates takes extra time, we finally adopt W3 as the edge weight to select the most salient dates in the rest of this paper. Note that, for completeness, we also generate daily summaries to obtain a complete news timeline per each date selection and evaluate the summaries by ROUGE scores in Table 2. The details about daily summarization is introduced in the next subsection.

Although the occurrence of an event signals its importance within the news timeline [4] and is well leveraged in existing timeline summarization algorithms, we note that the occurrence of events is also correlated with the recency of events, where past events occur earlier and are more heavily reported than recent events. Consequently, existing approaches may suffer from this issue. For example, approaches that optimize the summaries to

| Date Selection | Date Coverage (±3) | Date F1 | ROUGE-1 | ROUGE-2 | ROUGE-S* |
|---|---|---|---|---|---|
| | | Timeline17 | | | |
| Uniform | 0.8398 | 0.4475 | 0.3896 | 0.0917 | 0.1598 |
| W3 | 0.7828 | 0.5668 | 0.4000 | 0.0995 | 0.1676 |
| W3 + Recency | 0.8111 | 0.5542 | 0.4036 | 0.1005 | 0.1702 |
| | | Crisis | | | |
| Uniform | 0.5932 | 0.1325 | 0.3387 | 0.0570 | 0.1138 |
| W3 | 0.5459 | 0.2726 | 0.3573 | 0.0738 | 0.1246 |
| W3 + Recency | 0.5885 | 0.2748 | 0.3597 | 0.0760 | 0.1270 |

**Table 3: Performance on date coverage**

recover the whole corpus, such as ETS [29] and TILSE [12], will generate more summaries on past events.

In addition, as most references in articles refer to past events, the current date selection algorithm tends to give too much weight on old dates and will also result in timelines that lack recent dates. For a better illustration, we present the Cumulative Distribution Function (CDF) of the date duration between selected dates and the start date in Figure 4. As expected, both TILSE (Submodular) and date selection via PageRank (Tran et al.) tends to select more old dates, while the date distribution of ground-truth timelines is generally more uniform. Thus, we use the standard deviation of differences between consecutive dates to measure the uniformity of date distribution:

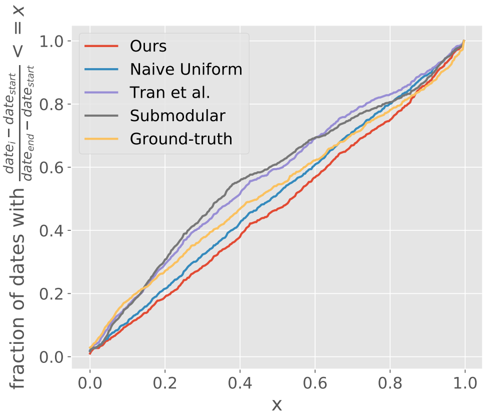
Definition 3 (Uniformity of Date Selection). *Given a series of selected dates $\{d_1, d_2, ..., d_T\}$ in chronological order, we regard the differences between consecutive dates as $\{diff_i = d_{i+1} - d_i\}$, then define its standard deviation $\sigma = \sqrt{\frac{1}{T}\sum_{i=1}^{T}(diff_i - \bar{diff})^2}$, as the uniformity of date selection.*

*2.2.1 Recency Adjustment.* To add more weights on recent dates, we leverage the Personalized PageRank algorithm [1], where the restart distribution is not uniform. More specifically, we weight each date node $date_i$ by $W_i = \alpha^{-d_i}$, where $d_i = |date_i - date_{start}|$. $\alpha$ ranges from 0 to 1 and is used to control the restart distribution. In practice, we use a grid search to find the $\alpha$ that gives the most uniform distribution in the date selection, then use the chosen dates for news timeline generation.

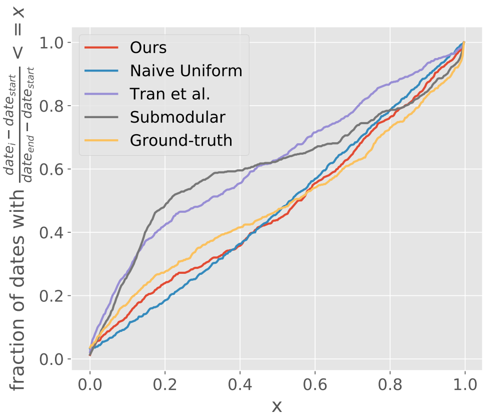
*2.2.2 Date Coverage.* To better check the coverage of generated timelines, besides f1 score on date selection, we also measure the date coverage, e.g., if any day of ground-truth date $g_i \pm 3$ days lies in the generated timeline, $g_i$ will be considered to be covered and we will measure what percentage of ground-truth dates are covered per timeline. For comparison, we also generate news timelines on truly uniformly distributed dates and present the results in Table 3. As we can see, although truly uniformly distributed dates cover the most ground-truth dates, due to the low accuracy in the date selection, the generated daily summaries are poor. However, adding recency adjustment with uniformity contributes to date selection in coverage, thus yields better timeline summarization.

## 2.3 Daily Summarization

Having selected the most salient dates, next, we divide timeline summarization into sub-summarization tasks. Although daily summarization tasks can be accomplished by any supervised or unsupervised document summarization algorithms, we intend to use a simple daily summarization method to validate the effectiveness of our explicit date selection, as complicated summarization techniques may introduce extra improvements in the performance. Specifically, we utilize the classic TextRank




**(a) Timeline17**




**(b) Crisis**

**Figure 4: Distribution of selected dates among different approaches.**

[13] to generate daily summaries. Similar to the task of date selection, TextRank constructs a sentence graph with sentences as nodes and similarity scores as edge weights. In particular, we use BM25 [18] to compute edge weights [2] and run PageRank on this directed graph to select the most important sentences as daily summaries.

*2.3.1 Post-processing.* Dividing large text summarization tasks into smaller ones greatly speeds up timeline generation, and these sub-tasks can naturally be further accelerated through parallel processing. Conducting text summarization on a daily basis rather than on the whole corpus, however, ignores temporal correlation and thus introduces redundancy in summarization. To remove redundancy across dates, therefore, we incorporate post-processing to re-rank daily summaries based on the whole summarization. Similar to MMR [3], instead of directly using daily summaries, we add sentences into timeline summarization by their daily ranks and only accept sentences whose maximum cosine similarity with selected ones is smaller than a threshold (e.g., < 0.5).

## 2.4 News Timeline Generation Algorithm

The generation algorithm of WILSON is summarized in Algorithm 1. First, we build a date reference graph based on the date

**Algorithm 1:** Algorithm for WILSON

**Input** : temporally tagged sentences $C = \{(date_i, sent_i)\}$
preset number of dates $T$
preset number of daily sentences $N$
**Output** : a series of daily summaries $(d_1, S_1), ..., (d_T, S_T)$

Build a date reference graph based on date co-occurrence $\{(date_i, date_j)|(date_i, sent_k) \in C \,\&\, (date_j, sent_k) \in C\}$ ;
Compute edge weight according to W3 in Section 2.2 ;
selected_dates $\leftarrow \emptyset$ ;
**for** *Grid search* $\alpha \in (0, 1)$ **do**
    *Compute personalized node weight for each date* $date_i$ *using* $W_i = \alpha^{-|date_i - \min_k(date_k)|}$ *;*
    *Run personalized PageRank to select the top* $T$ *ranked dates as a date selection candidate ;*
    *Based on Definition 3, compute the uniformity score of this date selection candidate ;*
    *selected_dates* $\leftarrow$ *save the date selection with the best uniformity score as* $(d_1, d_2, ..., d_T)$ *;*
**end for**
**for** $d_i \in$ *selected_dates* **do**
    *Find all sentences on* $d_i$ $C_i \leftarrow \{sent_k|(date_k, sent_k) \in C \,\&\, date_k = d_i\}$ *;*
    *Run TextRank on* $C_i$ *to rank all sentences by importance score in a max heap* $H_i$ *;*
    *Initialize selected sentences* $S_i \leftarrow \emptyset$ *;*
**end for**
**repeat**
    *Currently selected sentences* $S \leftarrow \bigcup_{i=1}^{T} S_i$ *;*
    *Top ranked sentence per day* $H \leftarrow \bigcup_{i=1}^{T} H_i[0]$ *;*
    *Remove top sentences: heap_pop(*$H_i$*) for* $i \in [1, T]$*;*
    *Remove sentences from* $H$ *that have maximum similarity > 0.5 with existing sentences in* $S$ *;*
    *Add remaining sentences in* $H$ *to the corresponding daily summary* $S_i$ *only if* $|S_i| < N$*;*
**until** *(all* $|S_i| = N$*) or (all* $|H_i| = 0$*);*
*return* $(d_1, S_1), ..., (d_T, S_T)$

pairs that appear in the same sentences. Second, we extract features to compute weights for the graph edges and run personalized PageRank to pick the most salient $T$ dates. More specifically, we include the recency adjustment strategy to improve the date coverage of selected dates. Then, we use TextRank to rank all sentences on each selected date. According to the sentence ranks per selected date, we post-process the sentences in batch and remove sentences that could introduce redundant information on existing selections. Finally, our algorithm produces a series of compact daily briefs as the summarized news timeline to help people better understand the evolution of the corresponding news event.

## 2.5 Complexity Analysis

In this section, we briefly provide a time complexity analysis of our approach with a comparison to the submodular framework. Denote $T$ as the total number of dates, $N$ as the average number of sentences per date, $t$ as the desired number of dates and $n$ as the desired number of sentences per date in the summarized timeline. According to PageRank on the dense graph, date selection takes $O(T^2)$ while $t$ daily summarization tasks take $O(t * N^2)$. Thus the total time complexity of WILSON is $O(T^2 + t * N^2)$.

For submodular framework [12], which conducts global summarization, it takes $O((TN)^2)$ to obtain pair-wise similarities for all sentences and takes $O(t * n * T * N)$ to iterate $t * n$ times to select each individual sentence in a greedy manner. Therefore, the total time complexity is $O((TN)^2 + t * n * N * T)$. In Figure 2, the corpus size is defined as the total number of sentences (i.e. $T * N$). As expected, the submodular frameworks show quadratic growth with a time complexity $O((TN)^2)$, while our approach is almost linear to the corpus size with a time complexity $O(T^2 + t * N^2)$.

Given the approximation that $T$ and $N$ are in the same order of magnitude (based on Table 4), WILSON runs faster than the submodular framework by a factor of $O(\frac{T^2}{t})$. Given around 10% date compression rate ($\frac{t}{T}$) and $T$ in hundred level, theoretically, our approach could gain over three orders of magnitude improvement in generation speed. Note that, due to the scalability issue of the submodular framework, [12] filtered sentences with predefined keywords to reduce $N$ by over one order of magnitude, reducing the time complexity in practice. Given ~10% filtering rate, our approach could still gain about two orders of magnitude in generation speed, which is consistent with experiments in Table 7.

| Dataset | # of topics | # of timelines | average per timeline | | |
|---|---|---|---|---|---|
| | | | # of doc | # of sents | duration days |
| Timeline17 | 9 | 19 | 739 | 36,915 | 242 |
| Crisis | 4 | 22 | 5,130 | 173,761 | 388 |

**Table 4: Dataset overview**

# 3 EMPIRICAL VALIDATION

## 3.1 Set-Up

### 3.1.1 *Datasets.*

We run experiments on *timeline17* [24, 25] and *crisis* [22]. Both datasets[2] consist of journalist generated timelines from major news media such as CNN, BBC and Reuters, and a corresponding corpus of articles per topic (e.g. H1N1 flu and Egypt war). More specifically, *timeline17* contains 19 timelines from 9 topics, while *crisis* involves 22 timelines from 4 topics. An overview of the two datasets is shown at Table 4.

### 3.1.2 *Competing methods.*

- **Random**: The system generates daily summaries by randomly selecting sentences from the corpus.
- **MEAD** [17]: a classic centroid-based multi-document summarization system.
- **Chieu et al.** [4]: a multi-document summarization system that uses date related TFIDF scores to measure sentence importance among corpus.
- **ETS** [29]: an unsupervised timeline summarization algorithm via simultaneously optimizing multiple heuristic metrics, including relevance, coverage, coherence, and diversity.
- **Tran et al.** [25]: a supervised timeline summarization algorithm, which extracts various features from sentences and leverages learning to rank techniques.
- **Regression** [26]: a supervised approach that formulates sentence selection as a linear regression problem.
- **Wang et al.** [27]: a supervised approach that formulates sentence selection as a matrix factorization problem.

[2] http://l3s.de/~gtran/timeline/

| Methods | ROUGE-1 | ROUGE-2 | ROUGE-S* |
|---|---|---|---|
| Random | 0.128 | 0.021 | 0.026 |
| Chieu et al. | 0.202 | 0.037 | 0.041 |
| MEAD | 0.208 | 0.049 | 0.039 |
| ETS | 0.207 | 0.047 | 0.042 |
| Tran et al. | 0.230 | 0.053 | 0.050 |
| Regression | 0.303 | 0.078 | 0.081 |
| Wang et al. (Text) | 0.312 | 0.089 | 0.112 |
| Wang et al. (Text + Vision) | 0.331 | 0.091 | 0.115 |
| Liang et al. | 0.334 | **0.105** | 0.103 |
| WILSON **(Ours)** | **0.370** | 0.083 | **0.141** |

**Table 5: Results on Timeline17**

| Methods | ROUGE-1 | ROUGE-2 | ROUGE-S* |
|---|---|---|---|
| Regression | 0.207 | 0.045 | 0.039 |
| Wang et al. (Text) | 0.211 | 0.046 | 0.040 |
| Wang et al. (Text + Vision) | 0.232 | 0.052 | 0.044 |
| Liang et al. | 0.268 | 0.057 | 0.054 |
| WILSON **(Ours)** | **0.352** | **0.074** | **0.123** |

**Table 6: Results on Crisis**

- **Liang et al.** [9]: a dynamic evolutionary framework leveraging distributed representation for timeline summarization.
- **ASMDS (TILSE)** [12]: TILSE is a state-of-the-art unsupervised timeline summarization approach, which incorporates submodularity-based multi-document summarization framework with temporal criteria.
- **TLSCONSTRAINTS (TILSE)** [12]: as a variant of TILSE, this method uses the same objective funtion as ASMDS but adopt different temporal constraints.

*3.1.3 Measurement.* Among all the baselines, TILSE is the only one with source code available. Consequently, for all the other baselines, we follow the existing works [9, 25, 27], which conduct experiments on *timeline*17 with settings mentioned at the beginning of Section 5.2 in [25] and directly report the baseline results from previous papers. More specifically, in the generated timeline, the number of selected dates $T$ is set to the number of dates in each ground-truth timeline, while the number of sentences per day $N$ is forced to be the rounded value of the average number of sentences per date from the ground-truth timeline.

To fairly compare with TILSE, we re-run the their code, follow all their pre-processing, including text cleaning and keywords filtering, and conduct experiments on exactly the same sentence corpus per timeline generation. Note that, [12] used a slightly different setting from previous papers: 1) for Timeline17 dataset, they mixed articles of the same topic from different news agencies together, which yields a bit higher ROUGE scores in timeline generation; 2) it suffers from the scalability issue and thereby uses filtered sentence corpus for both datasets. Thus, we followed their settings for a fair comparison with TILSE in Table 7. Wall time is measured on a 24-core 128GB machine.

*3.1.4 Evaluation Metrics.* The commonly used summarization metrics, ROUGE scores [10], including ROUGE-1, ROUGE-2 and ROUGE-S* F1 scores, are adopted to evaluate the agreement between machine-generated and journalist generated timelines. Moreover, to be consistent with TILSE comparison, we also include time-sensitive ROUGE scores as additional measurements [11]. More specifically, *concat* ROUGE scores totally ignore the time information by directly concatenating all texts together, while *agreement* ROUGE scores only consider the generated daily summaries on the groundtruth dates, and *align* ROUGE scores discount the quality of generated daily summaries by their distance to the corresponding groundtruth date. Last but not least, we test for significant improvements using an approximate randomization test [15] with a p-value of 0.05.

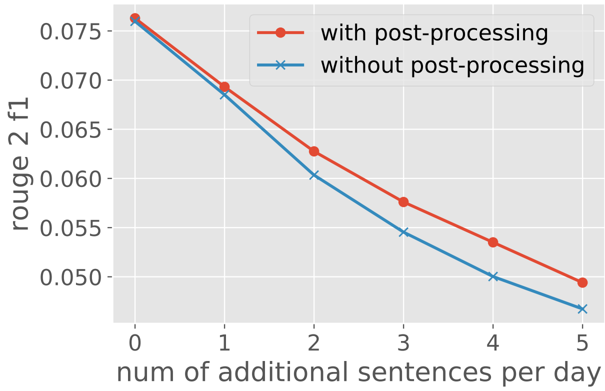



**Figure 5: Concat Rouge 2 f1 scores when adding more sentences on each date on Crisis.**

## 3.2 Performance Comparison

Table 5 and 6 shows that our unsupervised approach WILSON outperforms all baselines in ROUGE-1 and ROUGE-S* f1 scores by a significant margin, and is only second to [27], a supervised approach, and [9] in ROUGE-2 f1 score on Timeline17 dataset.

In addition, Table 7 illustrates that WILSON outperforms the state-of-the-art unsupervised framework TILSE in all ROUGE metrics. Averagely, our method outperforms the submodular approaches by 12.9% in concatenate ROUGE-2 scores, by 58.3% in agreement ROUGE-2 scores, and by 40.1% in alignment ROUGE-2 scores. More importantly, our method also gains two orders of magnitude improvement in generation speed, making it possible to generate news timelines in a real-time manner.

In Table 7, We also include multiple variants of WILSON for ablation analysis. WILSON-uniform simply adopts uniform date selection, while WILSON-Tran directly uses W3 as edge weight without recency adjustment. As expected, selecting uniformly distributed dates results in the worst summarization, while including recency adjustment improves time-sensitive ROUGE-2 scores by 9.0%~21.6%.

Overall, comparing with all competing approaches, the performance improvement of our method is higher in *Crisis* dataset. One explanation is that *Crisis* dataset contains more articles and spans a longer period, making it difficult for those competing approaches to correctly identify the long-term event dependencies, while our method mainly focuses on local dependencies.

*3.2.1 Effectiveness of Post-processing.* In Table 7, we observed that considering correlation across different dates and reducing redundant daily summaries are seemingly minor, especially on *Crisis* datasets. Different from *Timeline17* datasets, *Crisis* datasets consist of more compact daily summaries, where more than 90% dates contain only 1 sentence. Although reducing redundancy across dates is not necessary for timelines with compact daily summaries, we intend to verify the effectiveness of

| Model | concat Rouge 1 | concat Rouge 2 | our impr. | agreement Rouge 1 | agreement Rouge 2 | our impr. | align+ m:1 Rouge 1 | align+ m:1 Rouge 2 | our impr. | Date F1 | Running Time Per timeline (sec.) |
|---|---|---|---|---|---|---|---|---|---|---|---|
| Timeline17 | | | | | | | | | | | |
| ASMDS | 0.3452 | 0.0890 | 13.8% | 0.0913 | 0.0270 | 20.0% | 0.1047 | 0.0299 | 17.1% | 0.5437 | 338.68 |
| TLSCONSTRAINTS | 0.3685 | 0.0916 | 10.6% | 0.0912 | 0.0242 | 33.9% | 0.1049 | 0.0270 | 29.6% | 0.5127 | 560.24 |
| WILSON-uniform | 0.3659 | 0.0848 | 19.5% | 0.0754 | 0.0191 | 69.6% | 0.0924 | 0.0218 | 60.6% | 0.4366 | **1.97** |
| WILSON-Tran | 0.4007 | 0.0993 | 2.0% | 0.1035 | 0.0293 | 10.6% | 0.1181 | 0.0321 | 9.0% | **0.5668** | 2.12 |
| WILSON w/o Post | 0.4036 | 0.1005 | 0.8% | 0.1057 | 0.0318 | 1.9% | 0.1202 | 0.0344 | 1.7% | 0.5542 | 5.63 |
| WILSON | **0.4075**$\star\dagger$ | **0.1013**$\star\dagger$ | | **0.1065**$\star\dagger$ | **0.0324**$\dagger$ | | **0.1211**$\star\dagger$ | **0.0350**$\dagger$ | | 0.5542 | 7.59 |
| Crisis | | | | | | | | | | | |
| ASMDS | 0.3066 | 0.0645 | 17.7% | 0.0415 | 0.0091 | 123.1% | 0.0658 | 0.0135 | 71.9% | 0.2435 | 3055.96 |
| TLSCONSTRAINTS | 0.3307 | 0.0693* | 9.5% | 0.0564 | 0.0130 | 56.2% | 0.0764 | 0.0166 | 39.8% | 0.2739 | 4098.07 |
| WILSON-uniform | 0.3314 | 0.0551 | 37.7% | 0.0235 | 0.0059 | 244.1% | 0.0392 | 0.0080 | 190.0% | 0.1251 | **4.68** |
| WILSON-Tran | 0.3575 | 0.0739 | 2.7% | 0.0621 | 0.0167 | 21.6% | 0.0798 | 0.0202 | 14.9% | 0.2726 | 5.69 |
| WILSON w/o Post | 0.3600 | 0.0756 | 0.4% | 0.0677 | 0.0201 | 1.0% | 0.0843 | 0.0230 | 0.9% | **0.2748** | 22.95 |
| WILSON | **0.3605**$\star\dagger$ | **0.0759**$\star\dagger$ | | **0.0679**$\star$ | **0.0203**$\star\dagger$ | | **0.0846**$\star$ | **0.0232**$\star$ | | **0.2748** | 30.14 |

**Table 7: Comparison with TILSE. We indicate our improvement on Rouge 2 f1 score for different metrics. For** WILSON**, we use an approximate randomization test to test the significance of its improvement over ASMDS and TLSCONSTRAINTS, and denote the significant improvement by $\star$ and $\dagger$ respectively.**

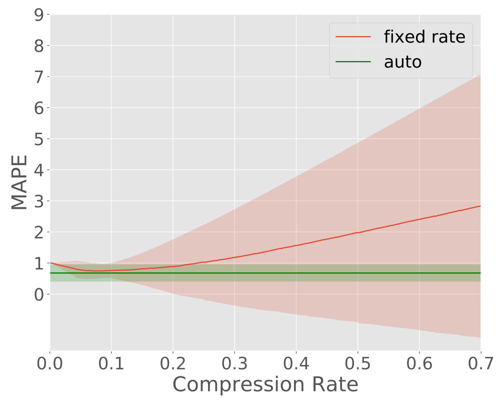
| Method | ROUGE-1 | ROUGE-2 |
|---|---|---|
| timeline17 | | |
| Submodularity framework [12] | 0.50 | 0.18 |
| Ground-truth date + Daily summary | 0.41 | 0.11 |
| Crisis | | |
| Submodularity framework [12] | 0.49 | 0.16 |
| Ground-truth date + Daily summary | 0.42 | 0.10 |

**Table 8: Empirical upper bound of submodularity framework and our two-stage method**

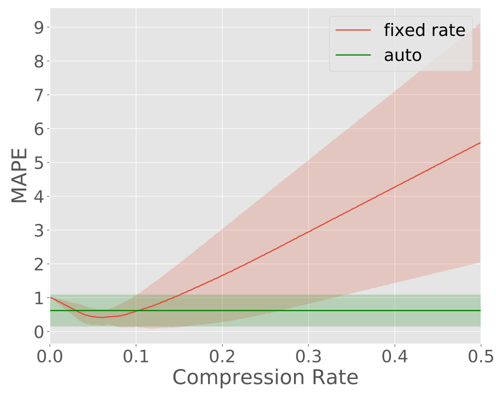
post-processing for timelines with abundant daily summaries. Instead of using the exact number of sentences per date in the ground-truth timelines, we generate timelines with more sentences per date, which is more practical as the true numbers are unknown. As demonstrated in Figure 5, simply adding daily summaries together suffers from the redundancy issue and using post-processing indeed helps. Note that we use the ROUGE-2 f1 score, so the overall scores going down with more sentences is because more generated texts lead to lower ROUGE accuracy.

3.2.2 *Empirical Bounds.* Empirical bounds of our two-stage method are given in Table 8, where we use ground-truth dates as date selections for daily summarization. Note that, besides using ground-truth dates, the upper bounds of the submodularity framework [12] also employ ground-truth summaries and are obtained by directly optimizing ROUGE f1 scores in a supervised way, but we only use ground-truth dates and never touch ground-truth summaries, making us aware of how date selection will contribute to news timeline summarization. As demonstrated, even without considering contextual correlation across different dates in text summarization, it is still possible to generate reasonable news timelines with accurate date selection. Although the upper bound of our two-stage framework is much lower than that of the submodular framework, it is worth mentioning that all existing approaches fail to reach our upper bound, not even close on the *Crisis* dataset.

3.2.3 *Automatic Date Compression.* As defined in Section 2.1, existing news timeline summarization works only use a preset number of dates and length of daily summaries to generate news timelines. Unlike the length of daily summaries, which only implies the compression rate for a single day and is usually set




**(a) Timeline17**




**(b) Crisis**

**Figure 6: Mean Absolute Percentage Error (MAPE) of predicted number of date selection**

to 2 or 3 sentences, determining the number of dates requires understanding for the whole corpus, making it difficult to select. To solve this issue, we aim at automatically detecting the number of dates for news timelines. Motivated by the fact that news timelines consist of major events within the duration, we propose to consider major event coverage to determine the number of dates. Specifically, we use the daily summarization to generate major events for each date and encode daily summaries with BERT

| Method | rank 1st | 2nd | 3rd | MRR | DCG |
|---|---|---|---|---|---|
| ASMDS | <u>4</u> | 3 | 3 | <u>0.72</u> | <u>7.39</u> |
| TLSCONSTRINTS | 1 | 6 | 3 | 0.56 | 6.29 |
| WILSON (Ours) | **5** | 1 | 4 | **0.76** | **7.63** |

**Table 9: Results of journalist evaluation on the quality of machine-generated timelines. Best and second best scores are highlighted by bold and underscore respectively.**

[5] into embedding vectors. Then, we use Affinity Propagation [6] to cluster encoded daily summaries into event clusters, and adopt the detected cluster number as date selection number. We compared our methods with fixed compression rates for date selection and presented the results in Figure 6. As shown, our automatic date compression method generally performs well on both datasets.

### 3.3 Evaluation by Journalists

In addition to ROUGE scores, we also consult two professional journalists at *the Washington Post*, which is one of the leading daily American newspapers, to manually evaluate the quality of machine-generated news timelines. Among 41 timelines from the two datasets, we sample 10 timelines (20%) from 6 topics, including H1N1 flu, BP oil spill, Egypt crisis, Libya war, Yemen war, and Syria war. For each sampled timeline, we present the human-generated ground-truth timeline and three machine-generated timelines from ASMDS, TLSCONSTRINTS, and WILSON (Ours) to journalists. The ground-truth timeline is labeled as a reference, while the other three are given in random order and the order is independent for each evaluation. The evaluation is based on the comprehensiveness and readability of the generated timelines compared with the ground-truth timelines.

For each evaluation, the two journalists are asked to review $\sim$ 80 daily summaries from $\sim$ 50 distinct dates, which adds up to $\sim$ 800 daily summaries from $\sim$ 500 distinct dates in total, and collaborate to provide one final ranking of the three machine-generated timelines. To measure the ranking performance of each method, we adopt two common rank-aware measurements, Mean Reciprocal Rank (MRR) and Discounted Cumulative Gain (DCG), and present the results in Table 9. As shown, when evaluated by professional journalists, our method outperforms the state-of-the-art unsupervised framework TLSCONSTRAINS and achieves slightly better or comparable results with ASMDS. Considering our method gains two orders of magnitude improvement in generation speed, the results are very encouraging. More interestingly, although TLSCONSTRAINS generally achieves higher ROUGE scores than ASMDS in table 7, TLSCONSTRAINS receives unexpectedly lower rank scores than ASMDS in this evaluation by journalists. This may imply a warning that automated measures may not be enough for news timeline summarization and human evaluation could be beneficial at times.

## 4 A CASE STUDY

In this section, we perform a qualitative analysis of the generated timelines of our approach. Since TILSE [12] is the only baseline with source code available, we also include its output in comparison. Table 10 presents a subset of timelines about the lawsuit of Michael Jackson's death in the *Timeline17* dataset, where the manually generated timeline was collected from BBC[3]. As different approaches generate timelines with different date selections, we only consider the dates that appear in all 4 timelines and show the first a few dates and their summaries in chronological order. We highlight the overlaps between manually generated and automatic generated timelines in colors and observe that the output of our approach is aligned better with the handcrafted one.

Interestingly, more summaries of our outputs are closer to the main events on each date than those of TILSE's, though they are all relevant to this topic. We think it may be because more important daily events are reported more heavily on each date, while existing models try but fail to effectively capture the evolution clues across dates, thus simple daily summarization can work well. Apparently, how to balance local and global summarization and effectively capture event evolution could be one potential direction for news timeline summarization.

## 5 REAL-TIME SYSTEM FRAMEWORK

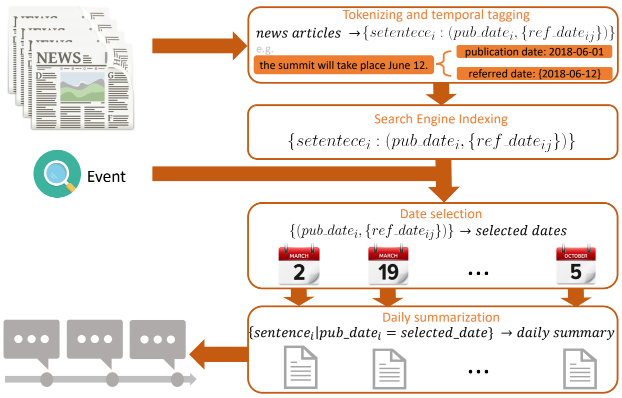



**Figure 7: Framework for Real-Time News Timeline Summarization**

The framework of our real-time news timeline generation system is shown in Figure 7. This framework applies our proposed method WILSON on a 4-year news corpus of over 1 million news articles[4] from the Washington Post and can generate timelines by event keywords in seconds. Firstly, we tokenize all the news articles into sentences and conduct temporal tagging to label each sentence. Then, to query relevant news content in real-time, we build a search engine on tagged sentences and index both date and content information. Specifically, we use ElasticSearch [7] as our backend search engine. Note that, we can easily include newly published news articles into our system by inserting them into the existing search engine. Finally, given both the keywords and duration time of a query event, our system will fetch related news sentences and run WILSON to generate a complete news timeline.

For example, we can generate a timeline about how the United States and North Korea reached the summit in seconds by setting query keywords to "trump, north korea, kim, summit, united states" and time duration between 2018-01-02 to 2018-06-12. We set the timeline length to 10 and present the output in Table 11. Taking journalist generated timeline[5] as a reference, we highlight

[3]https://www.bbc.com/news/entertainment-arts-15060651
[4]We excluded all news articles containing the keyword "timeline".
[5]https://www.washingtonpost.com/graphics/2018/national/trump-kim-jong-un-timeline/

| Groundtruth (From BBC) | TILSE (TLSCONSTRAINTS) | TILSE (ASMDS) | WILSON (Ours) |
|---|---|---|---|
| 2009-06-25<br>Dr Murray finds Jackson unconscious in the bedroom of his Los Angeles mansion .<br>Paramedics are called to the house while Dr Murray is performing CPR , according to a recording of the 911 emergency call .<br>He travels with the singer in an ambulance to UCLA medical center where Jackson later dies . | 2009-06-25<br>Jackson died at his Los Angeles home on 25 June aged 50 .<br>Jackson died at his home on 25 June last year at the age of 50 . | 2009-06-25<br>Michael Jackson died on 25 June 2009 from an overdose of the powerful anesthetic propofol . | 2009-06-25<br>Same drug class as morphine Given by tablets or injection Used post-surgery or for childbirth High doses can stop breathing or lead to delirium and seizures Jackson , who had a history of health problems , collapsed at his Los Angeles home around midday on Thursday .<br>Mr Martinez , who interviewed Dr Murray two days after Jackson 's death on 25 June 2009 , said the doctor told him the singer had stopped breathing shortly after 1100 . |
| 2009-06-28<br>Los Angeles police interview Dr Murray for three hours .<br>His spokeswoman insists he is " not a suspect " . | 2009-06-28<br>Jackson 's body was released to the family on Friday night .<br>Jackson 's body was released to the family on Friday night . | 2009-06-28<br>Jackson family left ' speechless and devastated ' by star 's death Relatives of Michael Jackson will seek a second autopsy on the star because they still have unanswered questions about his death , family friends say . | 2009-06-28<br>Michael Jackson 's family are said to be seeking a second autopsy because they still have questions about his death .<br>Earlier , veteran politician Rev Jesse Jackson , who has been counselling the family , said they had a flurry of questions of their own for Dr Murray . |
| 2009-07-28<br>Dr Murray 's home is also raided .<br>The search warrant allows " authorised investigators to look for medical records relating to Michael Jackson and all of his reported aliases " .<br>A computer hard drive and mobile phones are seized , and a pharmacy in Las Vegas is later raided in connection with the case . | 2009-07-28<br>Dr Conrad Murray , who police say is not a suspect , was at Jackson 's mansion and tried to revive him before he died .<br>Police raid Jackson doctor 's home Drug police are searching the Las Vegas home of Michael Jackson 's doctor as part of a manslaughter investigation into the singer 's death . | 2009-07-28<br>On Tuesday , police searched the Las Vegas home and offices of Jackson 's doctor , Conrad Murray , as part of a manslaughter investigation into the singer 's death . | 2009-07-28<br>Police raid Jackson doctor 's home Drug police are searching the Las Vegas home of Michael Jackson 's doctor as part of a manslaughter investigation into the singer 's death .<br>On Tuesday , police searched the Las Vegas home and offices of Jackson 's doctor , Conrad Murray , as part of a manslaughter investigation into the singer 's death . |
| 2010-06-25<br>Michael Jackson 's father , Joseph , files a wrongful death lawsuit against the physician . | 2010-06-25<br>Randy Jackson recently succeeded in stopping an unapproved tribute show to his brother Michael in Rome , which had been scheduled for 25 June , the anniversary of his death .<br>The suit was filed as fans around the world marked the first anniversary of Jackson 's death at the age of 50 . | 2010-06-25<br>Jackson died of a cardiac arrest at his home on 25 June last year . | 2010-06-25<br>25 June 2010 Michael Jackson 's father , Joseph , files a wrongful death lawsuit against the physician .<br>Fans sing outside the Jackson family home . |
| 2011-07-25<br>Rehearsal footage from Michael Jackson 's This Is It tour can not be used as evidence , the judge rules . | 2011-07-25<br>Judge Michael Pastor concluded on Monday that it would not help the defense and that " it was a waste of my time . "<br>25 July 2011 Rehearsal footage from Michael Jackson 's | 2011-07-25<br>But Judge Michael Pastor ruled on Monday that the film would not help the defense team and was a waste of his time . | 2011-07-25<br>Judge Michael Pastor concluded on Monday that it would not help the defense and that " it was a waste of my time .<br>25 July 2011 Rehearsal footage from Michael Jackson 's |
| 2011-08-30<br>Michael Jackson 's dermatologist is barred from giving evidence at the trial .<br>Dr Murray 's lawyers had planned to argue that Arnold Klein had administered the singer with painkillers for " no valid reason " but prosecutors said they were attempting to transfer responsibility for his death away from Dr Murray .<br>Testimony from five other doctors who treated Jackson is also disallowed . | 2011-08-30<br>Janet Jackson to miss concert Janet Jackson said she would find it " difficult " to attend the tribute concert in Cardiff Janet Jackson will not be attending her brother Michael Jackson 's tribute concert in Cardiff .<br>Because of the trial , the timing of this tribute to our brother would be too difficult for me , " Ms Jackson said in a statement . | 2011-08-30<br>Janet Jackson to miss concert Janet Jackson said she would find it " difficult " to attend the tribute concert in Cardiff Janet Jackson will not be attending her brother Michael Jackson 's tribute concert in Cardiff . | 2011-08-30<br>But Superior Court Judge Michael Pastor ruled that Arnold Klein would not be called to testify after prosecution lawyers said the defense wanted to transfer responsibility for Jackson 's death to the dermatologist .<br>Because of the trial , the timing of this tribute to our brother would be too difficult for me , " Ms Jackson said in a statement . |
| 2011-09-29<br>Jackson 's bodyguard , Alberto Alvarez , testifies that on the night Jackson died , Dr Murray ordered him to pick up vials of medicine before phoning for an ambulance .<br>" In my personal experience , I believed Dr Murray had the best intentions for Mr Jackson , " Mr Alvarez said . | 2011-09-29<br>29 September 2011 Last updated at 15:44 GMT Help Live coverage of the trial of Michael Jackson 's personal physician , Dr Conrad Murray , who is charged with involuntary manslaughter of the singer .<br>29 September 2011 Last updated at 04:16 GMT Help A key aide and a security guard have told the manslaughter trial of Michael Jackson 's doctor of events on the day the superstar died . | 2011-09-29<br>However , Jermaine and Randy Jackson said it should not go ahead because it would clash with the trial of Conrad Murray , the singer 's former doctor accused of his involuntary manslaughter . | 2011-09-29<br>Jackson 's bodyguard Alberto Alvarez claims Dr Murray " grabbed a handful of vials " and told him to put them in a bag Michael Jackson 's doctor told the performer 's bodyguard to pick up vials of medicine before phoning for help on the day he died , his trial has heard .<br>29 September 2011 Last updated at 04:16 GMT Help A key aide and a security guard have told the manslaughter trial of Michael Jackson 's doctor of events on the day the superstar died . |

**Table 10: Summary examples on the Death of Michael Jackson. To save space, we only select the first a few dates in chronological order, which appear in all 4 timelines. Main coverage with groundtruth is colored: red texts highlight the overlaps between groundtruth and TILSE/ours while blue texts highlight the distinct overlaps between groundtruth and ours. Note that all summarization approaches use exactly the same sentence candidates pool.**

| | |
|---|---|
| 2018-03-08<br>Jason Aldag The Post reports : North Korea 's belligerent leader , Kim Jong Un , has asked President Trump for talks and Trump has agreed to meet him " by May , ... | 2018-04-01<br>CIA Director Mike Pompeo , left , and North Korean leader Kim Jong Un shake hands during a meeting in in Pyongyang , North Korea on Easter Weekend . |
| 2018-04-16<br>The only way the United States can persuade North Korea to peacefully give up its pursuit of these weapons is if Kim believes Trump 's threat of military force is credible . | 2018-04-20<br>7:30 a.m. Friday North Korea 's state media reports that leader Kim Jong Un has left Pyongyang for the North - South summit meeting with South Korean President Moon Jae - in . |
| 2018-04-27<br>... North Korean leader Kim Jong Un on Friday morning ... walking into South Korea for a historic summit with President Moon Jae - in that will lay the groundwork for a meeting between Kim and President Trump . | 2018-05-09<br>Their release came as Secretary of State Mike Pompeo visited North Korea on Wednesday to finalize plans for a historic summit meeting between Trump and the North 's leader , Kim Jong Un . |
| 2018-05-16<br>North Korea has taken repeated ... and threatening to scrap next month 's planned summit between Kim and U.S. President Donald Trump , saying it wo n't be unilaterally pressured into relinquishing its nuclear weapons . | 2018-05-24<br>After weeks of receiving and even appearing to encourage chants of " Nobel " ahead of a planned historic meeting with North Korea dictator Kim Jong Un , President Trump on Thursday abruptly canceled the June 12 summit . |
| 2018-06-05<br>... Donald Trump cast his Tuesday summit with North Korea 's Kim Jong Un as a " one - time shot " for the autocratic leader to ditch his nuclear weapons and enter the community of nations ... | 2018-06-12<br>President Trump said the U.S. will end its " war games " with South Korea after the historic summit with North Korean leader Kim Jong Un on June 12 . |

**Table 11: WILSON generated output of the timeline about how the U.S. and North Korea finally had the summit. Main coverage with journalist generated timeline is color-coded blue, and some trivial contexts are omitted by ellipsis for space.**

the coverage between our generated news timeline with the journalist-generated timeline in blue color and demonstrate that our output is aligned well with the journalist-generated timeline, showing the effectiveness of WILSON in practice.

## 6 RELATED WORK

Existing works in summarizing timelines for a specific topic from relevant news articles include both supervised and unsupervised approaches. As representatives of supervised approaches, [25] leverages learning to rank techniques based on sentence features, while [27] proposes a matrix factorization framework to predict importance scores of sentences. Unsupervised approaches usually optimize task-specific heuristic object functions, which measure relevance, coverage and diversity of daily summaries. For example, [28] solves the optimization problem by iteratively substituting sentences in summaries, while the state-of-the-art framework TILSE adapts the sub-modularity framework from multi-document summarization domains to optimize timeline summarization [12].

In addition to extractive methods, some recent works also utilize abstractive summarization methods to generate more compact sentences as daily summaries [19]. Although the generated sentences are empirically proved to be readable, the reliability of generated summaries are not guaranteed, probably leading to false information. Extractive summarization methods, however, directly borrow sentences from original news articles and do not encounter reliability issue. Thus, in this paper, we utilize extractive summarization for both readability and reliability of generated timelines.

Besides ROUGE scores, [19] is the only existing work to include human in the evaluation, but they just assess the readability of daily summaries as they utilize the abstractive summarization. Since none of the previous studies utilize user study to measure the generation quality of the whole news timelines, we are the first work to include user study in timeline evaluation and consult journalists to assess the generation quality of the entire news timelines.

## 7 CONCLUSION

This paper shows that, with accurate date selection, we can generate high-quality news timelines without considering the temporal correlation of text summarization. Leveraging the explicit date selection, we propose a fast and effective unsupervised timeline summarization method named WILSON. Specifically, WILSON outperforms state-of-the-art approaches in both ROUGE scores and speed, significantly improving concatenate ROUGE-2 F1 scores by 9.5%∼17.7%, time-sensitive ROUGE-2 F1 scores by 17.1%∼123.1% and reducing generation time by two orders of magnitude, which allows us to develop a real-time news timeline generation system for the news room. More importantly, a user study with professional journalists also confirms that the outputs of WILSON are closer to human-generated ones than outputs of other methods. Last but not least, this work also suggests two potential directions for future works, i.e. considering both occurrence and recency of events for better salient date selection and reducing contextual correlation across dates by balancing local and global summarization to improve daily summarization.

## ACKNOWLEDGMENTS

The authors would like to thank the anonymous referees for their valuable comments and helpful suggestions. This work was in part supported by NSF awards #1742702, #1820609, #1909702, #1915801, and #1934782. Any opinions, findings and conclusions or recommendations expressed in this material are the author(s) and do not necessarily reflect those of the sponsors. Last but not least, we truly appreciate Everdeen Mason, Sophie Ho and their journalist team at the Washington Post for the valuable feedback and support.

## REFERENCES

[1] Bahman Bahmani, Abdur Chowdhury, and Ashish Goel. 2010. Fast incremental and personalized pagerank. *Proceedings of the VLDB Endowment* 4, 3 (2010), 173–184.

[2] Federico Barrios, Federico López, Luis Argerich, and Rosa Wachenchauzer. 2016. Variations of the similarity function of textrank for automated summarization. *arXiv preprint arXiv:1602.03606* (2016).

[3] Jaime Carbonell and Jade Goldstein. 1998. The use of MMR, diversity-based reranking for reordering documents and producing summaries. In *Proceedings of the 21st annual international ACM SIGIR conference on Research and development in information retrieval.* 335–336.

[4] Hai Leong Chieu and Yoong Keok Lee. 2004. Query based event extraction along a timeline. In *Proceedings of the 27th annual international ACM SIGIR conference on Research and development in information retrieval.* ACM, 425–432.

[5] Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2018. Bert: Pre-training of deep bidirectional transformers for language understanding. *arXiv preprint arXiv:1810.04805* (2018).

[6] Brendan J Frey and Delbert Dueck. 2007. Clustering by passing messages between data points. *science* 315, 5814 (2007), 972–976.

[7] Clinton Gormley and Zachary Tong. 2015. *Elasticsearch: the definitive guide: a distributed real-time search and analytics engine.* " O'Reilly Media, Inc.".

[8] Jiwei Li and Claire Cardie. 2014. Timeline generation: Tracking individuals on twitter. In *Proceedings of the 23rd international conference on World wide web.* ACM, 643–652.

[9] Dongyun Liang, Guohua Wang, and Jing Nie. 2019. A Dynamic Evolutionary Framework for Timeline Generation based on Distributed Representations. *arXiv preprint arXiv:1905.05550* (2019).

[10] Chin-Yew Lin. 2004. Rouge: A package for automatic evaluation of summaries. *Text Summarization Branches Out* (2004).
[11] Sebastian Martschat and Katja Markert. 2017. Improving rouge for timeline summarization. In *Proceedings of the 15th Conference of the European Chapter of the Association for Computational Linguistics: Volume 2, Short Papers*. 285–290.
[12] Sebastian Martschat and Katja Markert. 2018. A Temporally Sensitive Submodularity Framework for Timeline Summarization. In *Proceedings of the 22nd Conference on Computational Natural Language Learning*. 230–240.
[13] Rada Mihalcea and Paul Tarau. 2004. Textrank: Bringing order into text. In *Proceedings of the 2004 conference on empirical methods in natural language processing*.
[14] Jun Ping Ng, Yan Chen, Min-Yen Kan, and Zhoujun Li. 2014. Exploiting timelines to enhance multi-document summarization. In *Proceedings of the 52nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*. 923–933.
[15] Eric W Noreen. 1989. *Computer-intensive methods for testing hypotheses*. Wiley New York.
[16] Lawrence Page, Sergey Brin, Rajeev Motwani, and Terry Winograd. 1999. *The pagerank citation ranking: Bringing order to the web.* Technical Report. Stanford InfoLab.
[17] Dragomir R Radev, Hongyan Jing, Małgorzata Styś, and Daniel Tam. 2004. Centroid-based summarization of multiple documents. *Information Processing & Management* 40, 6 (2004), 919–938.
[18] Stephen Robertson, Hugo Zaragoza, et al. 2009. The probabilistic relevance framework: BM25 and beyond. *Foundations and Trends® in Information Retrieval* 3, 4 (2009), 333–389.
[19] Julius Steen and Katja Markert. 2019. Abstractive Timeline Summarization. In *Proceedings of the 2nd Workshop on New Frontiers in Summarization*. 21–31.
[20] Jannik Strötgen and Michael Gertz. 2015. A Baseline Temporal Tagger for all Languages. In *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing*. Association for Computational Linguistics, Lisbon, Portugal, 541–547. http://aclweb.org/anthology/D15-1063
[21] Russell C Swan and James Allan. 2000. TimeMine: visualizing automatically constructed timelines.. In *SIGIR*. 393.
[22] Giang Tran, Mohammad Alrifai, and Eelco Herder. 2015. Timeline summarization from relevant headlines. In *European Conference on Information Retrieval*. Springer, 245–256.
[23] Giang Tran, Eelco Herder, and Katja Markert. 2015. Joint graphical models for date selection in timeline summarization. In *Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics*, Vol. 1. Association for Computational Linguistics, 1598–1607.
[24] Giang Binh Tran, Mohammad Alrifai, and Dat Quoc Nguyen. 2013. Predicting relevant news events for timeline summaries.. In *WWW (Companion Volume)*. 91–92.
[25] Giang Binh Tran, Tuan A Tran, Nam-Khanh Tran, Mohammad Alrifai, and Nattiya Kanhabua. 2013. Leveraging learning to rank in an optimization framework for timeline summarization. In *SIGIR 2013 Workshop on Time-aware Information Access (TAIA*.
[26] Lu Wang, Claire Cardie, and Galen Marchetti. 2015. Socially-Informed Timeline Generation for Complex Events. In *Proceedings of the 2015 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*. 1055–1065.
[27] William Yang Wang, Yashar Mehdad, Dragomir R Radev, and Amanda Stent. 2016. A low-rank approximation approach to learning joint embeddings of news stories and images for timeline summarization. In *Proceedings of the 2016 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*. 58–68.
[28] Rui Yan, Liang Kong, Congrui Huang, Xiaojun Wan, Xiaoming Li, and Yan Zhang. 2011. Timeline generation through evolutionary trans-temporal summarization. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing*. Association for Computational Linguistics, 433–443.
[29] Rui Yan, Xiaojun Wan, Jahna Otterbacher, Liang Kong, Xiaoming Li, and Yan Zhang. 2011. Evolutionary timeline summarization: a balanced optimization framework via iterative substitution. In *Proceedings of the 34th international ACM SIGIR conference on Research and development in Information Retrieval*. ACM, 745–754.
[30] Deyu Zhou, Linsen Guo, and Yulan He. 2018. Neural Storyline Extraction Model for Storyline Generation from News. In *Proceedings of NAACL-HLT*. 1727–1736.
[31] Deyu Zhou, Haiyang Xu, Xin-Yu Dai, and Yulan He. 2016. Unsupervised Storyline Extraction from News Articles.. In *IJCAI*. 3014–3021.

# A REPRODUCTION

We present the experiment details to reproduce our results.

**Datasets and pre-processing.** Both *timeline17* and *crisis* are available at http://l3s.de/~gtran/timeline/. We use spaCy [6] to tokenize news articles into sentences. For temporal tagging, we use HeidelTime [7] to detect all temporal expressions in each sentence. If one sentence contain multiple date expressions, we consider all distinct date-sentence pairs in generating dated sentences $\{(date_i, sentence_i)\}$. Besides, each sentence is also paired with the publication date of the article it appears in.

**Evaluation.** As suggested at the beginning of Section 5.2 in [25], we set the number of selected dates $T$ to the number of dates in each ground-truth timeline, and the number of sentences per day $N$ to the rounded value of the average number of sentences per date from the corresponding ground-truth timeline. In Table 4 and Table 5, we follow existing works and use ROUGE-1.5.5 to get concatenate ROUGE scores, including ROUGE-1, ROUGE-2 and ROUGE-S*, which ignores date selection in the generated summarization and concatenate all daily summaries together. For comparison with TILSE, we use the evaluation library from the authors [8] for time-sensitive ROUGE scores in Table 6. But different from previous papers, for *Timeline17* dataset, TILSE [11] mixed articles of the same topic from different news agencies together and uses filtered sentence corpus for both datasets. Thus, for a fair comparison, we dump their sentence candidate pool through TILSE code and run our daily summarization on the same sentence candidate pool for each timeline. In speed evaluation, we do not consider the temporal tagging in the pre-processing, and only measure the speed of generation on the tagged sentences for both TILSE and WILSON. The wall time is measured on a 24-core machine.

**Implementation details of WILSON.** For daily summarization, we group dated sentences $\{(date_i, sentence_i)\}$ by the date to obtain the sentence candidates for each date. Since one sentence can have multiple paired dates, it may appear in multiple daily summaries. When utilizing TextRank [13] to generate daily summaries, we use BM25 [18] scores as edge weight. More specifically, when calculating the edge weight of one sentence to other sentences, we treat the source sentence as query and other sentences as documents, and use its BM25 relevance scores as edge weights. BM25 weights are unsymmetrical, so we build a directed graph for each date and then run the PageRank algorithm to select top sentences as daily summaries. For PageRank algorithm in both date selection and daily summarization, we use the implementation of NetworkX [9] with default damping parameter $\alpha = 0.85$. Code is available at https://github.com/wilson-nts/WILSON.

**Implementation details of baselines.** Among all the baselines, TILSE is the only one with source code available. Therefore, for all the other baselines, we follow the existing works [9, 25, 27], adopt the conventional experiment setting and directly report the results from previous papers. For the news timeline outputs of TILSE [12] (both TLSCONSTRAINTS and ASMDS), we use the author implementation [10] and their provided configurations [11]. Note that, the TILSE implementation uses the same processing (e.g. caches sentence similarity calculation) to generate multiple timelines that use the same news corpus, therefore, we add the processing time back in measuring the generation time per timeline.

[6]https://spacy.io
[7]https://github.com/HeidelTime/heideltime
[8]https://github.com/smartschat/tilse/tree/master/tilse/evaluation
[9]https://networkx.github.io/
[10]https://github.com/smartschat/tilse
[11]https://github.com/smartschat/tilse/tree/master/configs