

CrisisLTLSum: A Benchmark for Local Crisis Event Timeline Extraction and Summarization

Hossein Rajaby Faghihi^{1*}, Bashar Alhafni^{2*}, Ke Zhang³, Shihao Ran³,
Joel Tetreault³, Alejandro Jaimes³

¹Michigan State University, ²New York University Abu Dhabi,

³Dataminr, Inc.

rajabyfa@msu.edu, alhafni@nyu.edu,

{kzhang, sran, jtetreault, ajaimes}@dataminr.com

Abstract

Social media has increasingly played a key role in emergency response: first responders can use public posts to better react to ongoing crisis events and deploy the necessary resources where they are most needed. Timeline extraction and abstractive summarization are critical technical tasks to leverage large numbers of social media posts about events. Unfortunately, there are few datasets for benchmarking technical approaches for those tasks. This paper presents CrisisLTLSum, the largest dataset of local crisis event timelines available to date. CrisisLTLSum contains 1,000 crisis event timelines across four domains: wildfires, local fires, traffic, and storms. We built CrisisLTLSum using a semi-automated cluster-then-refine approach to collect data from the public Twitter stream. Our initial experiments indicate a significant gap between the performance of strong baselines compared to the human performance on both tasks. Our dataset, code, and models are publicly available.¹

1 Introduction

We present CrisisLTLSum, the first dataset on extraction and summarization of local crisis event timelines from Twitter. An example of an annotated timeline in CrisisLTLSum is shown in Figure 1. A timeline is a chronologically sorted set of posts, where each brings in new information or updates about an ongoing event (such as a fire, storm, or traffic incident). CrisisLTLSum supports two complex downstream tasks: **timeline extraction** and **timeline summarization**. As shown in Figure 1, the **timeline extraction** task is formalized as: given a seed tweet as the initial mention of a crisis event, extract relevant tweets with updates on the same event from the incoming noisy tweet stream. This task is crucial for real-time event tracking.

*Work done as Research Interns at Dataminr, Inc.

¹<https://github.com/CrisisLTLSum/CrisisTimelines>

Timeline	Part of the Timeline
June 21 00:04 #BREAKING Smoke from vegetation fire near Friant and Rice Road fills skies of Fresno amid strong winds (Seed Tweet)	
June 21 0:18 #BREAKING A large fire burning in northeast Fresno near Woodward Lake Thursday sent plumes of smoke into the air above the city.	✓
June 21 0:24 The season is upon us. Strong winds and fire. Stay safe and stay inside when you smell/see the smoke and dust. Word is that this is a grass fire at Friant and Rice.	✗ (Repetitive)
June 21 0:40 A large fire is burning along Friant Road. You can watch @VanessaABC30 giving an update on what we know right now.	✗ (Repetitive)
June 21 1:08 Winds have picked up immensely. Flames have reached Friant Rd. Windy conditions making it difficult for firefighters to contain the blaze.	✓
June 21 1:18 UPDATE: @VanessaABC30 talking to CAL FIRE about the fire burning along Rice Rd. & Friant Rd. Officials say the fire started as a commercial fire. Officials say there is a shelter in place order, but no evacuations at this point.	✓
June 21 1:31 Kevin Larrivee captured this video of a grass fire burning right now on Friant Rd, north of Woodward Park. The wind is crazy today. As you can see, that wind is pushing the fire quickly.	✓
June 21 1:33 Fire crews are working a vegetation fire near Friant and Rice roads. Please use caution when driving in the area and follow all directions from emergency personnel.	✓
Summary 1 Cal Fire officials have issued a shelter in place for residents northeast of Fresno near Woodward lake due to a vegetation fire. Strong winds have made it difficult for firefighters as it pushes it closer to Friant and Rice roads and officials urge motorists to use caution when driving in the area.	
Summary 2 A vegetation fire northeast of Fresno combined with windy conditions to cause firefighters problems. There was a shelter in place order given but no evacuation orders were given.	

Figure 1: This is a sample annotated timeline from CrisisLTLSum. The noisy timeline is the set of tweets sorted chronologically. The first tweet is the seed of the event. ✓ means that the tweet is annotated to be part of the timeline and ✗ indicates that the tweet is excluded. The reason for the exclusion is written under the mark.

The **timeline summarization** task aims to generate abstractive summaries of evolving events by aggregating important details from temporal and incremental information.

CrisisLTLSum can facilitate research in two directions: 1) NLP for Social Good (crisis domain), and 2) natural language inference and generation, i.e., timeline extraction and summarization tasks. Here, we discuss the importance and the differences of CrisisLTLSum compared to previous work for both of these aspects. Towards the first direction, the extraction of real-time crisis-relevant information from microblogs (Zhang and Eick, 2019; Mamo et al., 2021) plays a vital role in providing time-sensitive information to help first responders

understand ongoing situations and plan relief efforts accordingly (Sarter and Woods, 1991). CrisisLTLSum goes beyond the task of categorizing each single crisis-relevant post independently (Imran et al., 2013; Olteanu et al., 2014; Imran et al., 2016; Alam et al., 2018; Wiegmann et al., 2020; Alam et al., 2021a,b) and enables a more challenging task for extracting new updates of an ongoing crisis event from incoming posts and summarizing them with respect to the important event details. This can help provide time-sensitive updates while avoiding missing critical information in the bulk of the posts in microblogs due to the high volume of redundant and noisy information (Alam et al., 2021a). To the best of our knowledge, this is the first annotated dataset for such an extraction task, while this problem has been tackled before in unsupervised settings (Zhang et al., 2018).

Moreover, we focus on the extraction of **local** crisis events. The term “local” indicates that an event is bound to an exact location, such as a building, a street, or a county, and usually lasts for a short period. Building a corpus of local crisis events is particularly useful for first responders but also challenging because the timelines of these events are often not captured in existing knowledge sources. This means one has to design mechanisms for automatically detecting and tracking events directly from the Twitter stream, which is especially hard for existing clustering methods (Guille and Favre, 2015; Asgari-Chenaghlu et al., 2021) given the low number of available tweets for each local event.

For the second point, CrisisLTLSum enables NLP research on the complex tasks of timeline extraction and abstractive summarization. These tasks are particularly challenging in the context of social media. First, the process of identifying and extracting relevant updates for a specific event has to contend with the large volume of noise (Alam et al., 2021a) and informal tone (Rudra et al., 2018) compared to other domains such as news. Additionally, summarizing an on-going event helps toward a quick and better understanding of its progress. This requires a good level of abstraction with important details covered and properly presented (e.g., the temporal order of event evolution). CrisisLTLSum is the first dataset to provide human-written timeline summaries to support research in this direction.

CrisisLTLSum is developed through a two-step semi-automated process to create 1,000 local crisis timelines from the public Twitter stream. To our

best knowledge, this is the first timeline dataset focusing on “local” crisis events with the largest number of unique events. The contributions of this paper are as follows:

- We propose CrisisLTLSum, which is the largest dataset over local crisis event timelines. Notably, this is the first benchmark for abstractive timeline summarization in the crisis domain or on Twitter.
- We develop strong baselines for both tasks, and our experiments show a considerable gap between these models and human performance, indicating the importance of this dataset for enabling future research on extracting timelines of crisis event updates and summarizing them.

2 Related Work

Our work in this paper is related to two main directions of crisis domain datasets for NLP and timeline summarization.

Crisis Datasets for NLP: Prior research has investigated generating datasets from online social media (e.g., Twitter) on large scale crisis events, while providing labels for event categories (Wiegmann et al., 2020; Imran et al., 2013), humanitarian types and sub-types (Olteanu et al., 2014; Imran et al., 2016; Alam et al., 2018; Wiegmann et al., 2020; Arachie et al., 2020; Alam et al., 2021a,b), actionable information (McCreadie et al., 2019), or witness levels (Zahra et al., 2020) of each crisis related post. While existing datasets on crisis event timelines (Binh Tran et al., 2013; Tran et al., 2015; Pasquali et al., 2021) are limited to a small set of large-scale events, CrisisLTLSum covers a thousand timelines compared to only tens of events covered by each of the existing datasets. Additionally, we further go beyond the simple tweet categorization by enabling the extraction of information that include updates over the events’ progress.

Timeline Summarization: Timeline summarization (TLS) was firstly proposed in Allan et al. (2001), which extracts a single sentence from the news stream of an event topic. In general, the TLS task aims to summarize the target’s evolution (e.g., a topic or an entity) in a timeline (Martschat and Markert, 2018; Ghalandari and Ifrim, 2020). Existing approaches of TLS are mainly based on **extractive** methods, which are often grouped into several categories. For instance, Update Summarization (Dang et al., 2008; Li et al., 2009) aims to

update the previous summary given new information at a later time, while Timeline Generation (Yan et al., 2011; Tran et al., 2015; Martschat and Markert, 2018) aims to generate itemized summaries as the timeline, where each item is extracted by finding important temporal points (e.g., spikes, changes or clusters) and selecting representative sentences. Another category, Temporal Summarization, was first proposed in the TREC shared task (Aslam et al., 2013) with follow-up work (Kedzie et al., 2015), which targets extracting sentences from a large volume of news streams and social media posts as updates for large events. Temporal Summarization is close to the first task (Timeline Extraction) proposed in CrisisLTLSum.

There have been a few recent works on **abstractive** timeline summarization across different domains, e.g., biography (Chen et al., 2019), narratives (Barros et al., 2019), and news headlines (Steen and Markert, 2019), where the human-written summaries are directly collected from the web. The abstractive summarization goal is to generate a set of sentences summarizing the context of interest without taking the exact words or phrases from the original text but rather by combining them and summarizing the important content. To our best knowledge, CrisisLTLSum is the first to provide human-written summaries for crisis event timelines collected from noisy social media stream. Recent research (Nguyen et al., 2018) has also investigated the summarization task based on tweets in other domains which essentially do not reflect the challenges in the summarization of an evolving event.

3 CrisisLTLSum Collection

This section presents our semi-automated approach to collect CrisisLTLSum. We first extract clusters of tweets as noisy timelines and then refine them via human annotation to get clean timelines that only include non-redundant, informative, and relevant tweets.

3.1 Noisy Timeline Collection

Figure 2 shows the process for generating a set of noisy timelines starting from the Twitter stream and followed by pre-processing and knowledge enhancement steps, the online clustering method, and post-processing & cleaning steps.

Location, Time, and Keywords Filtering We limit the incoming tweets to specific geographical areas, periods, and domains of interest.

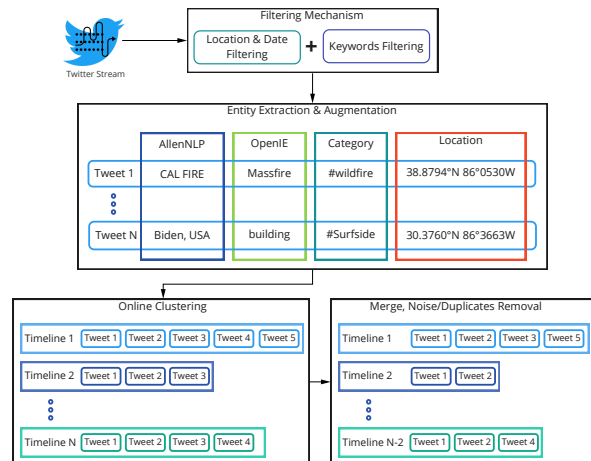


Figure 2: The process of noisy timeline collection. The output of this step are noisy clusters which are used to create the dataset.

The location filtering relies on a list of location candidates created by gathering cities, towns, and famous neighborhoods in a big area of interest. A tweet is considered relevant to our area of interest if 1) the text mentions one of the candidates, 2) the geo-tag matches the area of interest, or 3) the user location matches the area of interest. To limit the tweets to a specified crisis domain, we curate domain-specific keywords and only select tweets with phrases matching one of the keywords. This approach is not comprehensive or exhaustive but somewhat representative of each crisis domain. Improving this method to be more encompassing is an area for future research. The combinations of (area a , domain d , time t) are manually selected so that the events of type d at location a are more frequent during time period t . For instance, wildfire events are most likely to happen in California from May through August, while the same type of event is more likely from December to March in Victoria (Australia). More details with examples of curated keywords can be found in Appendix A.

Entity Extraction This step aims to extract entity mentions from the tweet text and provide additional information that can be used to help identify related tweets. We use three different modules. First, we use a pre-trained neural model from AllenNLP (Gardner et al., 2018), trained on CoNLL03 (Tjong Kim Sang and De Meulder, 2003), to extract entities with types of people, location, and organization. Although this module extracts some important entities in the text, it fails to extract uncommon entities or special mentions

such as the name of a wildfire. To address this, similar to prior research (Zheng and Kordjamshidi, 2020), we further exploit the extractions from OpenIE (Stanovsky et al., 2018) and select the noun arguments with less than ten characters as entities. Lastly, we add the tweet’s hashtags to the entity set. Since location mentions are crucial in extracting local events and existing models have low performance detecting them from noisy tweets text, we further developed a BERT-based NER model tuned on Twitter data to detect location mentions.

Location Augmentation We use OpenStreetMap API to map location mentions to physical addresses.² This step provides complementary information about each location while reducing the noise introduced by the entity extraction module through removing location mentions that are wrongly detected or are not located in the area of interest. This is especially important since our focus is on local events happening at specific locations.

Online Clustering This step aims to mimic the real-life scenario where tweets are sequentially fed into a clustering algorithm (Wang et al., 2015). We further choose this method since this is a lot faster than the retro-respective (all data available at the same time) clustering methods for a large pool of input data. Here, the clustering objective is to group tweets related to the same local event, such as a “fire in building A” or a “wildfire in a specific area”. The online clustering method utilizes a custom similarity metric that combines the similarity of the entities, the closeness of locations in the real world, and the existence of shared hashtags. Algorithm 1 shows the similarity computation between two tweets. The *smallest_distance* computes the minimum physical distance between location mentions given their augmented real-world location (the output of the location augmentation step). As the distance between higher-level location mentions such as state/city/country is always zero, we simply ignore those location types. The *find_matching_entities* function follows the ideas in Faghihi et al. (2020) on creating a unique matching matrix, which we use for extracting the top matching pairs of entities from tweets. Here, each entity can only be paired once with the highest matching-score entity from the other tweet. The *min_{dist}*, *max_{dist}*, *shashtag*, and *s_{dist}* are hyper-

²<https://www.openstreetmap.org/>

Algorithm 1 Find Similarity of tweet t_i and t_j

```

similarity = 0
if  $t_1.hashtags \cap t_2.hashtags \neq \emptyset$  then
    similarity = similarity + shashtag
end if
distmin = smallest_distance( $t_1, t_2$ )
if  $dist_{min} \geq max_{dist}$  then
    return 0
else if  $dist_{min} \leq min_{dist}$  then
    similarity = similarity + sdist
end if
top_pairs = find_matching_entities( $t_1, t_2$ )
norm_factor =
     $\min(len(t_1.entities), len(t_2.entities), top\_pairs)$ 

sentity =  $\sum top\_pairs.similarity / norm\_factor$ 

similarity = similarity + sentity
return similarity

```

parameters of the clustering algorithm. We have only used heuristics and a small set of executions to tune these hyper-parameters.

The pre-processed set of tweets is passed to the online clustering method, one tweet at a time. For each new tweet, similarity scores are computed between the new tweet and all cluster heads. The new tweet is added to the highest matching-score cluster where the similarity score is higher than *sim_{threshold}* and the time elapsed between the new tweet and the last update of the cluster is less than *time_{threshold}*. If the previous criteria are met for none of the clusters, a new cluster is created based on the new tweet. During this process, we remove inactive clusters whose last update was at least *expiration_{threshold}* minutes ago and have less than *tweet_{threshold}* number of tweets available. A cluster head is always the tweet with the most entity mentions; In case of a tie, the more recent tweet becomes the head of the cluster. The hyper-parameters of this method is noted in Appendix A.

Cluster Post-Processing We apply three post-processing steps to improve the quality of the generated clusters. First, we manually merge pairs of clusters with a cluster head similarity higher than a threshold *head_{min}*. This step compensates for some of the errors from missing entities in the pre-processing step, which affects the intermediate similarity scores in the clustering algorithm. Second,

we use a simple fuzzy sequence-matching technique to remove identical or similar tweets inside each cluster. Third, we train a BERT-based (Devlin et al., 2019) binary classifier to detect informative content, which can be used to prune out the noisy tweets that do not include crisis-relevant information. This classifier is trained on the available labeled data (Alam et al., 2021a) on tweets’ informativeness. Since most of the available tweets in Alam et al. (2021a) are specific to storm and wildfire domains and there are no representative subsets for our other domains of interest (traffic, local fire), we only apply this step to the clusters that are generated for those categories. These post-processing steps aim not to prune out all the noisy information but rather to provide a better starting point for our next steps.

3.2 CrisisLTLSum Human Annotation

Taking the noisy timelines generated from the previous step, we leverage human annotations to refine and generate clean timelines and summarize them. We, authors of this work, manually selected 1,000 clusters that contain enough tweets describing how a crisis event evolves, while specifying the “seed tweet” (i.e. the first observed post that describes the ongoing event) of each timeline. The detailed process is presented in Appendix B. The selected clusters cover events mainly from four crisis domains, including wildfire, local fire, storm, and traffic. More data statistics are shared in Section 4.

Procedure We use the Amazon Mechanical Turk (MTurk) platform to label and refine the noisy clusters to generate a clean timeline and collect the summaries. We split the annotation into multiple batches of Human Intelligence Tasks (HITs), where each batch contains timelines from the same domain. Each HIT contains three noisy timelines, and we collect annotations from 3 different workers on each. The workers are given the seed tweet and the subsequent tweets sorted by time, and they were asked to read the tweet one by one and answer i) whether the tweet should be part of the timeline, and ii) what is the reason if not.

A tweet is labeled as part of the timeline only if it satisfies all the following three conditions:

- *relevant*: talks about the same event indicated in the seed tweet
- *informative*: provides facts about the event but not only contains personal points of view
- *not repetitive*: brings in new information

Domain	Timelines	Tweets	✓	✗
Wildfire	423	4,829	1,961	2,868
Traffic	287	2,340	831	1,509
Fire	155	1,469	640	829
Storm	109	1,767	789	978
Other	26	205	82	123
	1,000	10,610	4,303	6,307

Table 1: Data statistics across different crisis domains in terms of the number of timelines and tweets. ✓ indicates tweets that are part of the timeline and ✗ indicates tweets that are not part of the timeline.

about the ongoing event

After reviewing all the tweets, the worker is finally asked to write a concise summary to describe how the event progresses over time. Detailed instructions and annotation workflows are presented as Figures 8-14 in Appendix E.

Annotation Workflow & Quality Control Following prior quality control practices (Briakou et al., 2021), we use multiple quality control (QC) steps to ensure the recruitment of high-quality annotators. First, we use location restriction (QC1) to limit the pool of workers to countries where native English speakers are most likely to be found. Next, we recruit annotators who pass our qualification test (QC2), where we ask them to annotate 3 timelines. We run several small pilot tasks, each with a replication factor of nine. We check annotators’ performance on timeline extraction task against experts’ labels and have experts manually review (QC3) annotators’ summary quality. Only workers passing all the quality control steps contribute to the final task. During the final task, we perform regular quality checks (QC4), and only use workers who consistently perform well.

Compensation We compensate the workers at a rate of \$3 per HIT for the task. Each batch of tasks is followed by a one-time bonus that makes the final rate over \$10 per hour.

4 CrisisLTLSum Statistics & Analysis

In this section, we cover comprehensive statistics and analysis of CrisisLTLSum to further elaborate on the statistical characteristics of our dataset.

4.1 Dataset Statistics

Out of these 1,000 annotated timelines (10,610 tweets) in CrisisLTLSum, 423 (42%) are about

wildfire, 287 (29%) are about traffic, 155 (15%) are about local fire, 109 (11%) are about storm, and 26 (3%) are about other types of events (e.g., building collapse). To make the ground-truth on whether each tweet is part of the timeline or not, we take the majority label among the three workers. 4,303 (41%) tweets are labeled as part of the timeline, whereas 6,307 (59%) are not. Out of all timelines, 110 (11%) only include tweets that are part of the timeline, whereas 68 (7%) did not include any. Table 1 presents the statistics across different event crisis domains.

4.2 Dataset Analysis

Timelines by Lengths Table 2 presents the aggregated length (i.e., number of tweets) distribution of the timelines. The majority of the timelines (447 or 45%) have five tweets or less. This is observed across all crisis domains except for storm events, where the majority of the timelines (43 out of 109) are 6 to 12 tweets in length. It is worth noting that our dataset includes long timelines of 26 or more tweets, constituting 9% (94 timelines) across all domains. Figure 3 presents the average percentage of part-of-timeline tweets based on the aggregated length distributions of the timelines. We notice an interesting trend across the domains: the longer the timeline, the lower the average percentage of tweets to be part of the timeline.

Annotation Quality To measure the agreement rate between workers on the timeline extraction task, we consider the annotations of two out of three workers who agree most per timeline. The average timeline-level agreement between those two workers is **90.06%**. We also explore a more profound analysis by comparing the workers’ annotations on 20% of the timelines against the annotations of experts. To do so, we evaluate the majority label provided by the three workers against the label provided by the experts. The average timeline-level agreement rate of this analysis is **91.77%**.

4.3 Dataset Splits

To aid reproducibility when using our dataset for various research experiments, we divided our dataset into: training (TRAIN: 70% or 706 timelines), development (DEV 10% or 86 timelines), and testing (TEST 20% or 208 timelines) splits. The splits are created via stratified sampling based on the event crisis domains and the length of timelines (i.e., number of tweets) as shown in Table 2.

Domain	Aggregated Timelines Lengths			
	[1 – 5]	[6–12]	[13–25]	[26 –]
Wildfire	175	140	59	49
Traffic	158	95	20	14
Fire	76	45	22	12
Storm	21	43	27	18
Other	17	8	0	1
	447	331	128	94

Table 2: Dataset statistics based on the aggregated timelines lengths (i.e., number of tweets) across different crisis domains.

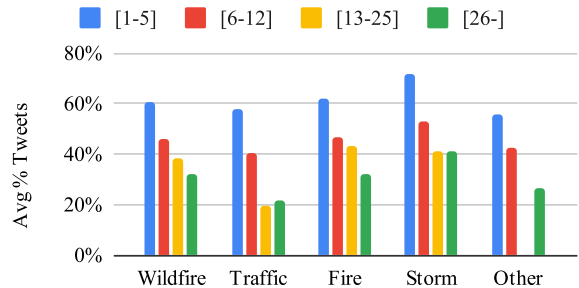


Figure 3: Average % of tweets that are part of the timelines based on the aggregated timelines lengths across different crisis domains.

Detailed statistics are available in Appendix D.

5 Experiments

Here, we first formalize the definitions of the downstream tasks. We then propose a set of existing naive and advanced models to serve as a baseline for these tasks. Next, we analyze the performance of the baselines on each task and provide error analysis for the best performing baseline to indicate the remaining major challenges of this dataset for future research.

5.1 Task Definitions

Given a crisis event timeline consisting of n tweets $T = [t_0, t_1, \dots, t_n]$, where t_0 is the seed tweet, i.e. the first observed post that describes the ongoing event, we define two tasks: **Timeline Extraction** and **Timeline Summarization**.

Timeline Extraction: Given an initial seed tweet t_0 , and following chronologically sorted tweets $t_i, i = 1, \dots, n$, the goal of the timeline extraction task is to determine whether tweet t_i is part of the timeline, i.e., related to the same event and adding new information compared to all prior tweets $[t_0, \dots, t_{i-1}]$. Up to time n , a clean timeline $T_{extracted} = [t_0, t_e^1, t_e^2, \dots, t_e^m], m \leq n$ is extracted

to describe how the event progresses over time.

Timeline Summarization: Similar to previous work by [Chen et al. \(2019\)](#), this task aims to generate a summary $\hat{Y} = \{\hat{w}_1, \dots, \hat{w}_k\}$ with sequence of words \hat{w}_i to concisely describe the crisis event and its evolution given the output from the timeline extraction task $T_{extracted}$. In particular, it tries to optimize the parameters that maximize the probability $P(Y/T_{extracted})$ given ground-truth summaries $Y = \{w_1, \dots, w_k\}$.

5.2 Modeling

5.2.1 Timeline Extraction

Naive Baseline We employ a simple and naive baseline that assigns the label of the majority class observed in TRAIN to detect whether a tweet is part of the timeline or not.

Sequence Classification We leverage pretrained language models to build sentence-level classifiers. We construct a list of tweet sequences S by concatenating all the tweets that are part of the timeline from t_1 to t_i : $S = [t_1, t_1 + t_2, \dots, t_1 + \dots + t_n]$. Therefore, each timeline of n tweets would yield n sequences. For our first sentence-level classification model, we fine-tune BERT ([Devlin et al., 2019](#)) on the tweet sequences where every training example would have the form of “ t_0 [SEP] s_i ”; where $s_i \in S$. We fine-tune this model using Hugging Face’s Transformers ([Wolf et al., 2020](#)).

Sequence Labeling Given the sequential nature of timelines, we treat the timeline extraction task as a sequence labeling problem. We create a model by adding a GRU ([Cho et al., 2014](#)) on top of BERT. Given a timeline $T = [t_0, t_1, t_2, \dots, t_n]$, we feed every tweet t_i to BERT and get its contextual representation corresponding to the [CLS] token. We then concatenate the representations of all the tweets and feed them to the GRU to predict if each tweet is part of the timeline or not.³

5.2.2 Timeline Summarization

Naive Baselines We define three simple models as the first set of timeline summarization baselines: 1) *first-tweet*: uses the first tweet of the timeline as the output summary; 2) *last-tweet*: uses the last tweet of the timeline as the output summary; and 3) *random-tweet*: uses a random tweet from the timeline as the output summary.

³It is important to note that the loss corresponding to the seed tweet t_0 is masked out during training.

Seq2Seq Models We further benchmark two pretrained sequence-to-sequence (Seq2Seq) models: BART ([Lewis et al., 2020](#)) and DistillBART ([Shleifer and Rush, 2020](#)). We chose these two models as they achieve strong results on various summarization datasets. For BART, we train it from scratch, and for DistillBART, we use a version that has been fine-tuned on both XSum ([Narayan et al., 2018](#)) and CNN/DM ([Hermann et al., 2015](#)) datasets. We use the fine-tuned DistillBART model in two settings: 1) further fine-tuning on CrisisLTL-Sum; 2) zero-shot setting. We use HuggingFace’s Transformers to fine-tune both of these models.

The input to the Seq2Seq models is the concatenation of all tweets that are part of the timeline. Since each timeline in our dataset is annotated by three workers (§3.2), it includes three summaries. To adapt the timeline summarization task to this setting, we pick the two summaries written by the two workers who agree the most based on the timeline extraction labels they assign to the tweets in each timeline. This reduces the variance between the summaries regarding their coverage of the crisis event described in the tweets that are part of the timeline. During fine-tuning, we double the examples in TRAIN by repeating each timeline twice, once for each summary. We describe all training settings and hyperparameters in Appendix C.

6 Results

6.1 Timeline Extraction

Table 3 presents timeline extraction results on the DEV set. For evaluation, we use the average timeline-level accuracy. The naive majority class baseline marks every tweet as not part of the timeline and achieves 48.57 average timeline accuracy. This is expected since 10 timelines in the DEV set only contain tweets that are part of the timeline (§4.1). The BERT-based sequence classification model achieves 74.64 average timeline accuracy, beating the BERT-GRU model, which achieves 65.86. Although the BERT-based sequence classification model has a limitation when extracting long timelines due to its limited size of 512 positional embeddings, it performs better than the BERT-GRU sequence labeling model. We attribute this to: 1) the careful preprocessing we did when constructing the sequences used to train the BERT-based sequence classification model, and 2) the limited data size, which might not enable the BERT-GRU model to fully capture the sequential relation-

ship across the tweets in the timelines.

For timeline extraction results on the TEST set, we compare both the human-level performance and the best model’s performance against experts’ annotations.⁴ To get the human-level performance on the TEST set, we average the performance of the two workers who agree most across all timelines in TEST. The best timeline extraction model achieves **73.51** average timeline accuracy, whereas the human-level performance is **88.98**. This highlights the difficulty of the timeline extraction task and indicates that more involved models are needed to close the gap between model performance and human-level performance.

6.2 Timeline Summarization

We evaluate all timeline summarization models with ROUGE (Lin and Och, 2004). The summarization output of each model is evaluated in a multi-reference setting against the two summaries written by the two workers who agree the most based on the timeline extraction labels. We use SacreROUGE (Deutsch and Roth, 2020) to compute multi-reference ROUGE scores.

Table 4 presents the F1 scores of ROUGE-1 (R1), ROUGE-2 (R2), and ROUGE-L (RL) of the timeline summarization models on the DEV set. The three naive baselines (first tweet, last tweet, and random tweet) achieve comparable performances, with R1 being the highest compared to R2 and RL. This is expected since the tweets in the timeline and the reference summaries (also tweets) all describe the same event, which leads to a considerable unigram overlap (i.e., R1).

For the Seq2Seq pretrained models, we present results in two settings: 1) using the oracle timeline; and 2) using the extracted timeline from the best timeline extraction model. In the oracle setting, we use the gold labels in the DEV set to identify tweets that are part of each timeline to the summarization models. The results of this oracle experiment estimate an upper bound for the summarization models used in this task. As shown in Table 4, the fine-tuned BART model achieves the best performance in terms of R1, R2 and RL. When we apply the best timeline extraction models to identify the tweets that are part of the timeline, we observe consistent conclusions. On TEST set, the best summarization model (i.e., BART) achieves **47.05**, **25.40**, and

⁴It is worth noting that this is the same set that was used to estimate the annotation quality (§4.2)

	Accuracy
Majority Class	48.57
BERT	74.64
BERT-GRU	65.86

Table 3: Results of timeline extraction models on DEV.

		R1	R2	RL
First tweet	w/o TE	32.15	14.35	24.09
Last tweet	w/o TE	28.69	12.03	21.65
Random tweet	w/o TE	30.56	13.61	23.24
BART	Oracle	47.72	26.16	36.90
	TE	45.32	24.61	34.68
DistillBART (Zero-shot)	Oracle	44.87	24.37	32.00
	TE	42.97	21.82	29.49
DistillBART	Oracle	47.50	25.11	34.64
	TE	45.04	22.61	32.51

Table 4: Results of different timeline summarization models after timeline extraction and in oracle settings on DEV.

35.90 in R1, R2, and RL, respectively.

Human Evaluation We further conducted a human evaluation to better assess the quality of the summaries. We take the whole TEST set and we evaluate eight summaries per timeline: 1) the three human-written summaries; 2) three model-generated summaries using the three models we develop for timeline summarization in the oracle setting; 3) two summaries from random systems: a random tweet that is part of the timeline, and a summary from a randomly selected timeline that belongs to a different crisis domain. The random systems add naive baselines and also serve as additional check on the annotation quality. Following a similar process in 3.2, we recruit a group of workers from MTurk with the same Location restriction (QC1), who can pass our pilot study as a qualification test. Each summary was evaluated by five different workers on a scale from 1 to 5 across four axes: Coherence, Accuracy, Coverage, and Overall quality, as was done by Lai et al. (2022). Detailed instructions and annotation workflows are presented as Figures 15–21 in Appendix E.

Table 5 presents the results. Looking at the results over all the timelines, the human-written summaries are significantly better than the ones generated by models, in terms of overall quality with an average rating of 4.12. The human-written summaries are also better in terms of accuracy, however,

	[All]				[6 -]			
	Acc	Coherence	Cov	Overall	Acc	Coherence	Cov	Overall
Human	4.75	4.85	4.14	4.12*	4.73*	4.83	3.97*	3.98*
BART	4.71	4.86	3.91	3.93	4.59	4.81	3.43	3.50
DistillBART	4.33	4.68	4.17	3.90	4.56	4.70	3.82	3.74
DistillBART (Zero-shot)	4.60	4.60	4.13	3.92	4.67	4.83	3.75	3.7
Random Tweet	4.72	4.71	3.40	3.40	4.53	4.64	2.72	2.79
Random Summary	1.16	4.55	1.16	1.17	1.24	4.45	1.25	1.25

Table 5: Results of human evaluation on human and model generated summaries. [All] means all timelines in the TESTand [6 -] means ones with length ≥ 6 . * denotes human summary is significantly ($p < 0.05$) better or worse than the best performing model based on one-sided Mann–Whitney U test (Mann and Whitney, 1947). Bold numbers denote best systems.

this was not statistically significant compared to the average accuracy ratings assigned to the models generated summaries. The models generated summaries have higher average ratings in terms of coherence and coverage, but this is not statistically significant compared to the human-written summaries. Notably, the gap of the ratings becomes larger for longer timelines (i.e., timelines with 6 or more tweets). The human-written summaries are significantly better in terms of accuracy, coverage, and overall. In terms of coherence, they are on par with the models generated summaries. This highlights the shortcomings of the summarization models when it comes to long timelines.

6.3 Error Analysis

We conduct an error analysis over the outputs of the best models for timeline extraction and summarization on DEV. For timeline extraction, 66 timelines (77%) have at least one error, in which 272 out of 958 tweets (28%) are wrong.

Length	Extraction	Summarization		
	Accuracy	R1	R2	RL
[1–5]	76.97	49.43	29.48	39.25
[6–12]	74.24	43.92	23.19	33.62
[13–25]	73.59	43.71	22.93	32.65
[26–]	67.37	35.10	11.87	21.73

Table 6: Best timeline extraction and summarization models performances on DEV based on timelines’ lengths.

Table 6 presents timeline extraction and summarization results of the best models based on different timeline lengths over the DEV set. We observe that the model performance decreases with increasing timeline length for both timeline extraction and summarization tasks, with significant drops in accuracy and ROUGE scores when timelines get longer.

Moreover, we inspected some of the generated summaries manually and we noticed that most of the summarization errors were due to hallucinations or to copying specific sentences that are present in the timeline without covering all the important event details described in the timeline. For instance, for the timeline in Figure 1, the best baseline summarization model (i.e., BART) generated the following summary: “A large fire burning in northeast Fresno near Woodward Lake sent plumes of smoke into the air above the city. Officials say the fire started as a commercial fire”. The two sentences in the generated summary are copied verbatim from the tweets in the timeline. This highlights the need for better models to capture the important details mentioned in the timeline.

7 Conclusion

We presented CrisisLTLSum, the first dataset on local crisis timelines extracted from Twitter and the first to provide human-written summaries on information extracted from Twitter. We showed that CrisisLTLSum supports two downstream tasks: Timeline Extraction and Timeline Summarization. Our experiments with SOTA baselines indicate that both of these tasks are challenging and encourage future research. Our dataset further provides a resource for developing methods on utilizing microblogs toward aiding first-responders in evaluating ongoing crisis events. In future, we plan to explore models that can solve both tasks in a joint setting by extracting new information from each update point and then summarizing those. This dataset can also be expanded by additional annotation to enable abstractive entity-based understanding of the event flow (Mishra et al., 2018; Faghihi and Kordjamshidi, 2021).

Limitations

This study has some limitations on the dataset generation workflow. First, our noisy timeline collection process is not a comprehensive and exhaustive extraction to find all available information about a local crisis event. Here, our focus is to provide a representative dataset rather performing a comprehensive set containing all the local crisis events and their updates. Second, the proposed noisy timeline collection pipeline is highly dependent to the performance of the entity extraction modules, especially for the location extraction, and the accuracy of the OpenStreetMap API to find the correct physical address of each location mention. Accordingly, replicating the same process for other language or based on other locations may be hard because of such dependencies. Furthermore, the online clustering method used in this paper has a set of hyperparameters which are tuned heuristically from a small set of experiments. More comprehensive and large scale experiments on tuning those parameters could potentially impact the quality of the generated timelines. Generating similar results by following our noisy timeline collection process is in general limited by the users' access to the public Twitter stream and the changes in the available posts (they may become restricted or deleted).

Acknowledgements

We thank Aoife Cahill for providing detailed feedback on this paper, Alison Smith-Renner and Wenjuan Zhang for helpful discussions on MTurk annotations, our colleagues at Datamir, and the EMNLP reviewers for their helpful and constructive comments.

References

- Firoj Alam, Ferda Ofli, and Muhammad Imran. 2018. Crisismmd: Multimodal twitter datasets from natural disasters. In *Twelfth international AAAI conference on web and social media*.
- Firoj Alam, Umair Qazi, Muhammad Imran, and Ferda Ofli. 2021a. Humaid: Human-annotated disaster incidents data from twitter with deep learning benchmarks. In *ICWSM*, pages 933–942.
- Firoj Alam, Hassan Sajjad, Muhammad Imran, and Ferda Ofli. 2021b. Crisibench: Benchmarking crisis-related social media datasets for humanitarian information processing. In *ICWSM*, pages 923–932.
- James Allan, Rahul Gupta, and Vikas Khandelwal. 2001. Temporal summaries of new topics. In *Proceedings of the 24th annual international ACM SIGIR conference on Research and development in information retrieval*, pages 10–18.
- Chidubem Arachie, Manas Gaur, Sam Anzaroot, William Groves, Ke Zhang, and Alejandro Jaimes. 2020. Unsupervised detection of sub-events in large scale disasters. In *Proceedings Of The AAAI conference on artificial intelligence*, volume 34, pages 354–361.
- Meysam Asgari-Chenaghlu, Mohammad-Reza Feizi-Derakhshi, Leili Farzinvasht, Mohammad-Ali Balfar, and Cina Motamed. 2021. Topic detection and tracking techniques on twitter: a systematic review. *Complexity*, 2021.
- Javed A Aslam, Matthew Ekstrand-Abueg, Virgil Pavlu, Fernando Diaz, and Tetsuya Sakai. 2013. Trec 2013 temporal summarization. In *TREC*.
- Cristina Barros, Elena Lloret, Estela Saquete, and Borja Navarro-Colorado. 2019. Natsum: Narrative abstractive summarization through cross-document timeline generation. *Information Processing & Management*, 56(5):1775–1793.
- Giang Binh Tran, Mohammad Alrifai, and Dat Quoc Nguyen. 2013. Predicting relevant news events for timeline summaries. In *Proceedings of the 22nd International Conference on World Wide Web*, pages 91–92.
- Eleftheria Briakou, Di Lu, Ke Zhang, and Joel Tetreault. 2021. Olá, bonjour, salve! xformal: A benchmark for multilingual formality style transfer. In *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 3199–3216.
- Xiuying Chen, Zhangming Chan, Shen Gao, Meng-Hsuan Yu, Dongyan Zhao, and Rui Yan. 2019. Learning towards abstractive timeline summarization. In *Proceedings of the Twenty-Eighth International Joint Conference on Artificial Intelligence, IJCAI-19*, pages 4939–4945. International Joint Conferences on Artificial Intelligence Organization.
- Kyunghyun Cho, Bart van Merriënboer, Caglar Gulcehre, Dzmitry Bahdanau, Fethi Bougares, Holger Schwenk, and Yoshua Bengio. 2014. Learning phrase representations using RNN encoder–decoder for statistical machine translation. In *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 1724–1734, Doha, Qatar. Association for Computational Linguistics.
- Hoa Trang Dang, Karolina Owczarzak, et al. 2008. Overview of the tac 2008 update summarization task. In *TAC*.

- Daniel Deutsch and Dan Roth. 2020. [SacreROUGE: An open-source library for using and developing summarization evaluation metrics](#). In *Proceedings of Second Workshop for NLP Open Source Software (NLP-OSS)*, pages 120–125, Online. Association for Computational Linguistics.
- Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. [BERT: Pre-training of deep bidirectional transformers for language understanding](#). In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 4171–4186, Minneapolis, Minnesota. Association for Computational Linguistics.
- Hossein Rajaby Faghihi and Parisa Kordjamshidi. 2021. Time-stamped language model: Teaching language models to understand the flow of events. In *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 4560–4570.
- Hossein Rajaby Faghihi, Roshanak Mirzaee, Sudarshan Paliwal, and Parisa Kordjamshidi. 2020. Latent alignment of procedural concepts in multimodal recipes. In *Proceedings of the First Workshop on Advances in Language and Vision Research*, pages 26–31.
- Matt Gardner, Joel Grus, Mark Neumann, Oyvind Tafjord, Pradeep Dasigi, Nelson F. Liu, Matthew Peters, Michael Schmitz, and Luke Zettlemoyer. 2018. [AllenNLP: A deep semantic natural language processing platform](#). In *Proceedings of Workshop for NLP Open Source Software (NLP-OSS)*, pages 1–6, Melbourne, Australia. Association for Computational Linguistics.
- Demian Gholipour Ghalandari and Georgiana Ifrim. 2020. Examining the state-of-the-art in news timeline summarization. *arXiv preprint arXiv:2005.10107*.
- Adrien Guille and Cécile Favre. 2015. Event detection, tracking, and visualization in twitter: a mention-anomaly-based approach. *Social Network Analysis and Mining*, 5(1):1–18.
- Karl Moritz Hermann, Tomás Kociský, Edward Grefenstette, Lasse Espeholt, Will Kay, Mustafa Suleyman, and Phil Blunsom. 2015. [Teaching machines to read and comprehend](#). In *NIPS*, pages 1693–1701.
- Muhammad Imran, Shady Elbassuoni, Carlos Castillo, Fernando Diaz, and Patrick Meier. 2013. Practical extraction of disaster-relevant information from social media. In *Proceedings of the 22nd international conference on world wide web*, pages 1021–1024.
- Muhammad Imran, Prasenjit Mitra, and Carlos Castillo. 2016. Twitter as a lifeline: Human-annotated twitter corpora for nlp of crisis-related messages. *arXiv preprint arXiv:1605.05894*.
- Chris Kedzie, Kathleen McKeown, and Fernando Diaz. 2015. Predicting salient updates for disaster summarization. In *Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics and the 7th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, pages 1608–1617.
- Vivian Lai, Alison Smith-Renner, Ke Zhang, Ruijia Cheng, Wenjuan Zhang, Joel Tetreault, and Alejandro Jaimes-Larrarte. 2022. [An exploration of post-editing effectiveness in text summarization](#). In *Proceedings of the 2022 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 475–493, Seattle, United States. Association for Computational Linguistics.
- Mike Lewis, Yinhan Liu, Naman Goyal, Marjan Ghazvininejad, Abdelrahman Mohamed, Omer Levy, Veselin Stoyanov, and Luke Zettlemoyer. 2020. [BART: Denoising sequence-to-sequence pre-training for natural language generation, translation, and comprehension](#). In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 7871–7880, Online. Association for Computational Linguistics.
- Liangda Li, Ke Zhou, Gui-Rong Xue, Hongyuan Zha, and Yong Yu. 2009. Enhancing diversity, coverage and balance for summarization through structure learning. In *Proceedings of the 18th international conference on World wide web*, pages 71–80.
- Chin-Yew Lin and Franz Josef Och. 2004. [Automatic evaluation of machine translation quality using longest common subsequence and skip-bigram statistics](#). In *Proceedings of the 42nd Annual Meeting of the Association for Computational Linguistics (ACL-04)*, pages 605–612, Barcelona, Spain.
- Nicholas Mamo, Joel Azzopardi, and Colin Layfield. 2021. Fine-grained topic detection and tracking on twitter. In *KDIR*, pages 79–86.
- H. B. Mann and D. R. Whitney. 1947. [On a Test of Whether one of Two Random Variables is Stochastically Larger than the Other](#). *The Annals of Mathematical Statistics*, 18(1):50 – 60.
- Sebastian Martschat and Katja Markert. 2018. A temporally sensitive submodularity framework for timeline summarization. *arXiv preprint arXiv:1810.07949*.
- Richard McCreadie, Cody Buntain, and Ian Soboroff. 2019. Trec incident streams: Finding actionable information on social media.
- Bhavana Dalvi Mishra, Lifu Huang, Niket Tandon, Wen-tau Yih, and Peter Clark. 2018. Tracking state changes in procedural text: a challenge dataset and models for process paragraph comprehension. *arXiv preprint arXiv:1805.06975*.

- Shashi Narayan, Shay B. Cohen, and Mirella Lapata. 2018. [Don't give me the details, just the summary! topic-aware convolutional neural networks for extreme summarization](#). In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 1797–1807, Brussels, Belgium. Association for Computational Linguistics.
- Minh-Tien Nguyen, Dac Viet Lai, Huy Tien Nguyen, and Minh Le Nguyen. 2018. Tsix: a human-involved-creation dataset for tweet summarization. In *Proceedings of the Eleventh International Conference on Language Resources and Evaluation (LREC 2018)*.
- Alexandra Olteanu, Carlos Castillo, Fernando Diaz, and Sarah Vieweg. 2014. Crisislex: A lexicon for collecting and filtering microblogged communications in crises. In *Eighth international AAAI conference on weblogs and social media*.
- Arian Pasquali, Ricardo Campos, Alexandre Ribeiro, Brenda Santana, Alípio Jorge, and Adam Jatowt. 2021. Tls-covid19: A new annotated corpus for timeline summarization. In *Advances in Information Retrieval*, pages 497–512, Cham. Springer International Publishing.
- Koustav Rudra, Niloy Ganguly, Pawan Goyal, and Saptarshi Ghosh. 2018. Extracting and summarizing situational information from the twitter social media during disasters. *ACM Transactions on the Web (TWEB)*, 12(3):1–35.
- Nadine B Sarter and David D Woods. 1991. Situation awareness: A critical but ill-defined phenomenon. *The International Journal of Aviation Psychology*, 1(1):45–57.
- Sam Shleifer and Alexander M. Rush. 2020. [Pre-trained summarization distillation](#).
- Gabriel Stanovsky, Julian Michael, Luke Zettlemoyer, and I. Dagan. 2018. Supervised open information extraction. In *NAACL-HLT*.
- Julius Steen and Katja Markert. 2019. [Abstractive timeline summarization](#). In *Proceedings of the 2nd Workshop on New Frontiers in Summarization*, pages 21–31, Hong Kong, China. Association for Computational Linguistics.
- Erik F. Tjong Kim Sang and Fien De Meulder. 2003. [Introduction to the CoNLL-2003 shared task: Language-independent named entity recognition](#). In *Proceedings of the Seventh Conference on Natural Language Learning at HLT-NAACL 2003*, pages 142–147.
- Giang Tran, Mohammad Alrifai, and Eelco Herder. 2015. Timeline summarization from relevant headlines. In *European Conference on Information Retrieval*, pages 245–256. Springer.
- Zhenhua Wang, Lidan Shou, Ke Chen, Gang Chen, and Sharad Mehrotra. 2015. [On summarization and timeline generation for evolutionary tweet streams](#). *IEEE Transactions on Knowledge and Data Engineering*, 27(5):1301–1315.
- Matti Wiegmann, Jens Kersten, Friederike Klan, Martin Potthast, and Benno Stein. 2020. Disaster tweet corpus 2020. *Disaster Tweet Corpus 2020 [Data Set]*.
- Thomas Wolf, Lysandre Debut, Victor Sanh, Julien Chaumond, Clement Delangue, Anthony Moi, Pierric Cistac, Tim Rault, Remi Louf, Morgan Funtowicz, Joe Davison, Sam Shleifer, Patrick von Platen, Clara Ma, Yacine Jernite, Julien Plu, Canwen Xu, Teven Le Scao, Sylvain Gugger, Mariama Drame, Quentin Lhoest, and Alexander Rush. 2020. [Transformers: State-of-the-art natural language processing](#). In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing: System Demonstrations*, pages 38–45, Online. Association for Computational Linguistics.
- 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*, pages 745–754.
- Kiran Zahra, Muhammad Imran, and Frank O Ostermann. 2020. Automatic identification of eyewitness messages on twitter during disasters. *Information processing & management*, 57(1):102107.
- Chao Zhang, Dongming Lei, Quan Yuan, Honglei Zhuang, Lance Kaplan, Shaowen Wang, and Jiawei Han. 2018. Geoburst+ effective and real-time local event detection in geo-tagged tweet streams. *ACM Transactions on Intelligent Systems and Technology (TIST)*, 9(3):1–24.
- Yongli Zhang and Christoph F Eick. 2019. Tracking events in twitter by combining an lda-based approach and a density–contour clustering approach. *International Journal of Semantic Computing*, 13(01):87–110.
- Chen Zheng and Parisa Kordjamshidi. 2020. Srlgrn: Semantic role labeling graph reasoning network. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 8881–8891.

A Noisy Timeline Collection

In this section, we provide some additional information on the noisy timeline collection process.

A.1 Location and Date Selection

We build CrisisLTLSum by limiting the search to four crisis domains: wildfire, local fire, traffic, and storm. We collect CrisisLTLSum across

seven states in the USA and two states of Australia during 2020 and 2021. The selected states in the USA are New York, Illinois, Massachusetts, California, Texas, and Florida. From Australia, we have selected the states of New South Wales and Victoria. These states are selected as they are some of the areas subject to the most events falling into our crisis domains of interest. California, Victoria, and New South Wales are mainly selected due to the abundance of wildfires in hot seasons. Texas and Florida have been selected as they are both subject to wildfires and storm events during different months. Massachusetts and Illinois are selected first because those areas are frequently subjected to bad weather, and second as they contain big cities subject to traffic and local fire events alongside New York.

A.2 Keyword Filtering

This step selects the subset of filtered tweets related to a specific crisis category of interest. Ideally, this task can be performed by a neural model trained on a large set of tweets with labeled data indicating the categories. However, as such a large amount of labeled data is unavailable, we rely on a common approach of designing keywords. We carefully curate a keyword list for each of our categories of interest by employing expert knowledge gathered through over-viewing crisis stories in news, social media, and related big disaster events of the past. Table 7 shows some example keywords used in each of the crisis domains in CrisisLTLSum. Please note that these lists are generic for each category and not specific to unique events. For instance, instead of defining keywords specific to an event called “Hurricane Ida”, our keywords list includes phrases such as “fallen tree”, “building collapse”, or “storm”. The quality of the keywords list is crucial to the final quality of the generated timelines, and we polish this list multiple times based on small sets of experiments before using them for the final task. Each keyword can be either a single word or multiple words. To avoid making a long list of keywords and ensure that different lexical formats of the same word are still considered in the keyword matching process, we maintain both a lower-cased and a lemmitized version of the keywords and the tweet’s text. If any of the keywords exist in the text and in any of these formats, then the tweet is considered related to the category. The multi-word keywords are not considered n-grams but as an in-

dications that all the words in the keyword should exist in the text, even if not as a sequence. This approach will not be comprehensive or exhaustive but rather representative of each crisis domain. Improving this method to be more encompassing is an area for future research.

A.3 Clustering Hyper-Parameters

We use the following parameters shared among all different domains: $s_{hashtag}$ is set to 0.2 and s_{dist} is 0.3. The $sim_{threshold}$ and $time_{threshold}$ are set to 0.7 and 15 hours respectively. The $time_{threshold}$ is reduced to 3 hours to avoid merging various accidents at different times but in the same location. The min_{dist} and max_{dist} are set to 0.4 kilometer and 4 kilometer for fire and traffic events while having the larger range of 0.4 and 10 kilometers for the wildfire and storm extraction. The $expiration_{threshold}$ is set to 15 hours and $tweet_{threshold}$ is 4.

A.4 Automatic and Manual Cluster Merge

The online clustering approach may fail to properly relate some tweets due to missing entities, clustering method hyper-parameters, or the difference in the description of various angles of the same story. To address this, we merge clusters by comparing all cluster heads with each other and combining those with a higher similarity score than a threshold s_{head} . Additionally, we use human feedback to merge clusters where their similarity score is below s_{head} but higher than a second hyper parameter s_{head}^{min} . This process can compensate for a portion of the missing entities due to entity extraction accuracy or tweets’ informal text.

A.5 Duplicate Removal

Here, the goal is to remove tweets from the same cluster with identical or similar text. The duplicate removal relies on a fuzzy string sequence-matching technique to compute the similarity between a pair of tweets. We simply go over each tweet sorted in chronological order and remove the ones that match any of the previous tweets in the same cluster with a matching score higher than d_{match} .

A.6 Noise Removal

Although the keyword filtering step would reduce the number of tweets unrelated to the category of interest, this step aims to further remove the unrelated clusters generated from the pipeline. To

Wildfire	wildfire, bushfire, campfire, volcano erupt, forest fire, ash fall, vegetation fire
Local fire	house fire, structure fire, building fire, gas leak, fire alarm, tower burned
Storm	tornado, shelter, storm damage, building collapse, storm collapse, roof damage, fallen tree
Traffic	traffic delay, road blocked, car fire, crash, injury, rollover, accident, stalled vehicle

Table 7: Sample keywords for each crisis domain in CrisisLTLSum.

do so, we use a neural model for detecting the informativeness of tweets’ text by training it on the available labeled data (Alam et al., 2021a). As the definition of informativeness in our domain differs from the data source, we further define a mapping between their label set and the informativeness as we refer to it. Hence, any fine-grained label related to personal emotions, prayers, or donations is removed from the informative set. Since this process is very category relevant and the existing labeled datasets do not cover all of our categories of interest, we can only apply this step to those categories where a representative subset exists for them in the available resources. We formulate this task as a Boolean tweet classification task to predict the tweet informativeness by applying a linear classification module on top of the aggregation token of a transformer-based language model.

B Noisy Clusters Selection for Human Annotation

This section gives more details on how we select noisy clusters for human annotation. Our goal is to select 1000 clusters that contain enough tweets describing an evolving crisis event. In particular, we first identify a “seed tweet” as the first observed post mentioning a local crisis event, then roughly check whether the following tweets in the cluster contain updates about the same event. Taking the example in Figure 1, we first see “a vegetation fire happened” in the seed tweet, then see “plumes of smoke”, “windy condition impacts fire control”. We select clusters across four crisis domains, including wildfire, fire, storm, and traffic, depending on how frequently each type of event happens in extracted noisy clusters.

B.1 Sample noisy timelines

Figure 4 shows the noisy timeline of the same annotated example from Figure 1. Figure 5, 6, and 7 show sample noisy timelines from other domains in CrisisLTLSum.

C Detailed Experimental Setup

C.1 Timeline Extraction

BERT We fine-tune BERT base uncased on a single GPU for 10 epochs with a learning rate of $5e-5$, batch size of 32, a seed of 42, and a maximum sequence length of 512. At the end of the fine-tuning, we pick the best checkpoint is based on the performance on the DEV set.

BERT-GRU For the BERT-GRU sequence labeling model, we use BERT base uncased to get the contextual representation for each tweet. The GRU has one layer with a hidden size of 128. The model was trained for 50 epochs with early stopping after five epochs if the performance did not improve on the DEV set. We use a learning rate of $5e-5$, a batch size of 16, and a seed of 42.

C.2 Timeline Summarization

We fine-tune BART based on a single GPU for ten epochs with a learning rate of $5e-5$, batch size of 16, a seed of 42, and a maximum target sequence length of 512. For DistillBART, we use the same hyperparameters. During inference, we use beam search with a beam size of 4. At the end of the fine-tuning, we pick the best checkpoint is based on the performance on the DEV set.

D Detailed Data Splits

Detail statistics on TRAIN, DEV, and TEST are shown in Table 8.

E Annotation Interface

Figure 8 - 14 show the annotation interface for the Timeline Extraction & Summarization task. Figure 15 - 21 show the annotation interface for the Summarization Quality Estimation (QE) task.

For the QE task, we first display the rating rubrics and examples to the workers as shown in Figure 15 - 17. To ensure the workers have a good understanding of the QE dimensions listed in the rubrics, as shown in Figure 18, the workers are asked to pass a screening test before they can access the quality rating part of the task interface.

Domain	Train				Dev				Test			
	[1 – 5]	[6–12]	[13–25]	[26 –]	[1 – 5]	[6–12]	[13–25]	[26 –]	[1 – 5]	[6–12]	[13–25]	[26 –]
Wildfire	125	101	42	35	14	11	5	4	36	28	12	10
Traffic	113	66	15	9	13	8	2	2	32	21	3	3
Fire	54	32	15	8	7	3	2	1	15	10	5	3
Storm	14	30	18	12	2	4	3	2	4	2	0	0
Other	11	5	0	1	2	1	0	0	5	9	6	4

Table 8: Aggregated data statistics based on crisis domains and timeline lengths of the TRAIN, DEV, and TEST splits.

Timeline	
June 21 00:04	#BREAKING Smoke from vegetation fire near Friant and Rice Road fills skies of Fresno amid strong winds
June 21 0:18	#BREAKING A large fire burning in northeast Fresno near Woodward Lake Thursday sent plumes of smoke into the air above the city.
June 21 0:24	The season is upon us. Strong winds and fire. Stay safe and stay inside when you smell/see the smoke and dust. Word is that this is a grass fire at Friant and Rice.
June 21 0:40	A large fire is burning along Friant Road. You can watch @VanessaABC30 giving an update on what we know right now.
June 21 1:08	Winds have picked up immensely. Flames have reached Friant Rd. Windy conditions making it difficult for firefighters to contain the blaze.
June 21 1:18	UPDATE: @VanessaABC30 talking to CAL FIRE about the fire burning along Rice Rd. & Friant Rd. Officials say the fire started as a commercial fire. Officials say there is a shelter in place order, but no evacuations at this point.
June 21 1:31	Kevin Larrivee captured this video of a grass fire burning right now on Friant Rd, north of Woodward Park. The wind is crazy today. As you can see, that wind is pushing the fire quickly.
June 21 1:33	Fire crews are working a vegetation fire near Friant and Rice roads. Please use caution when driving in the area and follow all directions from emergency personnel.

Figure 4: Sample noisy timeline of a wildfire event. This is the same example depicted in Figure 1.

Once they answer the test question with the correct answer, the rating interface will show up, as shown in Figure 19. In this page, we display the original tweet timeline on the left hand side of the rating interface. Before displaying all the summaries and the rating options, to make sure the workers go over the timeline carefully, we ask them to use the length of the timeline to answer the last screening question before conducting the rating task. The rating part of the interface is shown in Figures 20 and 21.

Timeline

May 29 10:21	US-92 is closed in both directions in Daytona Beach at Indian Lake Rd due to a crash. Motorists should seek an alternate route. https://t.co/WQwXpPupG9
May 29 10:23	#TRAFFICALERT: A vehicle fire is currently blocking the westbound lanes of ISB/US-92 at Indian Lake Rd in Daytona Beach. Westbound traffic is closed at LPGA Blvd. Expect delays, try to seek an alternate route if possible
May 29 10:29	crash is on west side of US 92 at Indian Lake Rd. and per @FHPOrlando road is closed at LPGA https://t.co/LCeSorvj2o
May 29 11:02	** MAJOR FATAL CRASH ** WB International Speedway Blvd x Indian Lake Rd - fatality on scene - WB Int Speedway CLOSED #Volusia #Monday https://t.co/5Wklu88sDi

Figure 5: Sample noisy timeline of a traffic event.

Timeline

July 28 22:49	#BREAKING Earlier reported overflow Los Flores de Catia As a result, flood waters have entered the buildings #Flood #Venezuela #Caracas #BreakingNews https://t.co/2FMIWrZb71 https://t.co/LpW1BP39fx
July 28 22:56	Guaire River overflowed at corner of Puerto Escondido, which caused a perimeter wall to collapse #Flood #Venezuela #Caracas #Lluvias #Breakingnews https://t.co/MEqXOSzFt8 https://t.co/gRvxdUdXOE
July 28 23:00	Internet users posted videos of the flow near the Ciudad Tamanaco Shopping Center & on the Francisco Fajardo Highway near Macaracuay. Other areas of the city were flooded after the rains that fell #Flood #Venezuela #Caracas #Lluvias https://t.co/uqNddfk6TC https://t.co/g7z9UBbG9P
July 28 23:03	In the city it has been raining heavily from approximately 3:00 in the afternoon. Different media & #Twitter users have also reported floods & difficulties to travel the roads,highways & streets due to the heavy rainfall #Flood #Venezuela #Caracas #Lluvias https://t.co/BVcjbBWbyY https://t.co/ovzP6MLgin
July 28 23:08	The floods are mostly affecting the west of the city. People point out in social networks that in Quinta Crespo, in the Sucre parish, in Plaza #Venezuela & in Catia the roads remain flooded. #Flood #Caracas #Lluvias https://t.co/uOjCbQcZF9 https://t.co/H05zISnRqn

Figure 6: Sample noisy timeline of a storm event.

Timeline

August 10 17:50	Cambridge: Pleasant and Allston St, command striking a Working Fire. Fire caused by a transformer explosion, Exterior pole and tree on fire, companies working on stopping the fire from extending to a 3 story large wood frame apartment house.
August 10 17:54	Command reporting multiple wires on fire all the way down Allston St, command requesting power cut to the block https://t.co/zJyDCQ1s1Q
August 10 19:00	Allston St electrical fire in #cambridgema from construction truck knocking over pole. Looks to be under control. Thanks, @cambridgefd https://t.co/bQ2gfoBxKp
August 10 19:37	The fire caught the tree (foreground) on fire, as well as the wires for 3 blocks up to Pleasant and Allston. This resulted in another transformer explosion further up the block, I believe. (I didn't see this one.) 2/
August 10 21:19	Ladder 17 a Structure Fire. 67 Acorn Street, Boston. Nearest Box 1373 CHESTNUT & WALNUT STS. 17:19 https://t.co/8xBLJ7m8Kd

Figure 7: Sample noisy timeline of a fire event.

The Timeline Generation Annotation

1 Introduction 2 Tutorial 3 Annotate 4 Submit

Task Introduction

This task involves creating event timelines. An event timeline is a set of **related**, **not repetitive**, and **informative** tweets describing an evolving crisis domain event, such as a spreading wildfire. Later tweets in the timeline provide more information about the event and how it is evolving over time. For example, a wildfire timeline might include tweets about the fire burning at a small scale, the fire spreading, the efforts to contain the fire, and a final tweet about the fire being extinguished.

During this task, you will create five event timelines and write brief summaries for them. For each timeline, **you will review a stream of tweets to determine which tweets belong in your event timeline (and which should not be included)**. Tweets do not belong if they are not relevant to the event, are repetitive (or duplicates) of prior tweets in the timeline, or do not add any useful information about the event. After you've reviewed all tweets and chosen to include or not include them in your timeline, **you will write a short summary describing the event and its evolution.**

→

Figure 8: The first page of our annotation interface for the Timeline Extraction & Summarization task, which contains the task introduction and a brief instruction.



How to Create a Timeline

The following steps describe how you will review tweets to create timelines and then summarize them. You will walk through a sample timeline of tweets. Pay attention to the decisions made for each tweet and the explanations for those decisions

Step 1:

To start, review the first (earliest) tweet of the timeline to learn about the event:

22 June 2020, 12:25:33 Tweet 1 of 11

A fire started in Sashta county at 2 am this morning.

This tweet appears to be about a wildfire event happening in Sashta county in california.

Step 2:

Now review the remaining tweets in chronological order to determine if they belong in the timeline:

- First, read the tweet and determine whether it belongs to your timeline. *A tweet belongs to the timeline if it adds new information about the main event and its consequences.*
- Second, if the tweet does not belong to the timeline, tell us why:
 - It is **not informative** (for example, it is a personal opinion or a generic tweet): The tweet *doesn't talk about crisis events*, or it is just a *personal comment with no valuable event update*.
 - It is **repetitive** to prior tweets in the timeline: *All* the information provided in the tweet has already been observed by prior tweets and there is not any small new information about the event in this tweet.
 - It is **not relevant** to the current event timeline: The tweet is about a crisis event but is not about the events of this timeline.
 - Other (open-ended response)
- Click "save" and continue to the next tweet. *You can change your response for each tweet at any time before submitting the timeline by clicking "edit".*

22 June 2020, 04:10:25 Tweet 2 of 11

Fire has started in sashta county from this morning.

Does this tweet belong to the timeline (does it add additional information)? No Yes

Select the reasons this tweet does not belong in the timeline Repetitive Information to ^

This tweet does not belong to the timeline because it provides repetitive information to the prior tweets.

Figure 9: Step 1 - 2 of the annotation instructions in our annotation interface for the Timeline Extraction & Summarization task

22 June 2020, 19:23:10 Tweet 11 of 11

Evacuation in order for J city in Sashta and traffic in I-9 due to the wildfire.

Does this tweet belong to the timeline (does it add additional information)? No Yes

This tweet belongs to the timeline, because it adds new information (evacuation and traffic). These are consequences of the wildfire which are considered very important and relevant to the main story.

Step 3:

After reviewing all tweets, the tweets in green will make up your timeline. Now you will write a short summary describing the event timeline and its evolution. For the summary, imagine you are a news reporter who has been asked to summarize the event for a breaking news report. You should write a short summary (between 1 to 5 sentences) to describe **what** happened, **where** it happened, any reasons for or major consequences of the event, and the final status. You do not need to include every single detail of the event in your summary.

You could write the following summary to describe the timeline above: "A wildfire in Sashta county started at 2 am of June 22nd and spread north while 2 groups of firefighters battled the fire. It grows into a 20-acre fire and airforce is deployed to help control the situation. The fire causes evacuation in J city and traffic on I-9"

Attention: To continue this task, please copy and paste the above summary into the following textbox.

Summarize the timeline:

This is just a sample summary, please **don't** copy-paste this to next timeline summaries.

Figure 10: Step 3 of the annotation instructions in our annotation interface for the Timeline Extraction & Summarization task

Step 4:

When you are finished with your summary, you will click "next" to go on to the next timeline. You will complete five timelines as part of this task.

Final Hints:

- If one of your timelines has a lot of tweets, **Don't Worry!!** the other timelines will be very short.
- Check the tweet timestamp, because if a tweet is much later than prior tweets, it might not be related to the timeline.
- Pay attention to mentions of important people, infrastructure, locations, and authorities involved, as these can help you better determine whether the new tweet is related to the prior ones in the timeline. Tweets about the same event and location are likely to belong to the same timeline (unless they are repetitive)
- If a tweet seems out of order (e.g., event described by the tweet should have happened sooner logically), we still consider it relevant to the timeline. For instance, a tweet about (Fire alarm in apartment X) might come after the tweet about (Fire took down apartment X).
- If a tweet is about a consequence of the prior events, or a sub-event related to the main event, we consider it relevant to the timeline.
- Don't copy-paste one of the tweets as the summary! instead, write the summary in your own language given what you have learned about the events in the timeline. **Don't** start the summaries with sentences like `This event is about`, `the timeline was`.
- Images and external links are not considered as new information. Also, more hashtags is not considered as new information.
- If you are unsure whether something belongs to the timeline or not, please use the **Other** option and then add the reason of your uncertainty in the input box that pops up.
- Be aware of the domain, for instance there is difference between a tweet saying 1st alarm and another one saying 2nd alarm for a fire event and that is considered new information. The same happens if there is an update about the evacuation orders regarding expansion, extending, removal, continuing. Specifically numbers and location are very important and we want to include any new information regarding those. For the traffic events, be careful about the time interval between tweets and the extension of traffic delays and blockage. If traffic grows that is a new information.
- If you are unsure about your decision, please select the Other option and explain your reasons.

When you are ready, click the button below to start the task.



Figure 11: Step 4 and final hits of the annotation instructions in our annotation interface for the Timeline Extraction & Summarization task

← To Instructions

Timeline 1 out of 5

Step 1: Review the first (earliest) tweet of the timeline to learn about the event.

2021-08-25 01:21:26+00:00 Tweet 1 of 54

@officialhambly Oof. The smoke has finally mostly cleared out for us from the Dixie, monument and McFarland fires. Stay safe

Step 2: Review the remaining tweets in chronological order to determine if they belong in the timeline.

2021-08-25 02:06:13+00:00 Tweet 2 of 54

Watch Live: Fire officials provide latest on Dixie Fire | Evacuations, road closures, updates <https://t.co/vCFh9NrUv7>

Does this tweet belong to the timeline (does it add additional information)? No Yes

Save

2021-08-25 04:10:10+00:00 Tweet 3 of 54

@buriedbybooks Ugh this. My Facebook feed has been completely silent about the Dixie Fire minus my close friend who lives in Westwood. When the Caldor Fire started, suddenly it was OMG NOOOOOO NOT TAHOE!!!!!! So sorry on behalf of urban Californians who don't get it.

Does this tweet belong to the timeline (does it add additional information)? No Yes

Save

Figure 12: Step 1 - 2 of the task page in the annotation interface for the Timeline Extraction & Summarization task

2021-08-26 06:00:17+00:00 Tweet 53 of 54

Lifting spirits with delicious, fresh-cooked meals, Guy Fieri spent the day taking care of firefighters battling the Dixie Fire in NorCal. <https://t.co/P7cOueSy3p>

Does this tweet belong to the timeline (does it add additional information)? No Yes

Save

2021-08-26 07:11:24+00:00 Tweet 54 of 54

Smoke should stay east of the Bay Area on Thursday but as dry north winds develop by Friday, smoke from the Dixie and Monument fires will likely drift southward over the Bay Area on Friday. #FireWeather #CAwx <https://t.co/taJfnyBJVU>

Does this tweet belong to the timeline (does it add additional information)? No Yes

Save

Step 3: Summarize the timeline

Next Timeline

[← To Instructions](#)

Figure 13: Step 3 of the task page in the annotation interface for the Timeline Extraction & Summarization task

The Timeline Generation Annotation

1 Introduction
 2 Tutorial
 3 Annotate
 4 Submit

[←](#)

Thank you for completing the tasks. We will invite you for the next experiment given your performance.

Are you willing to do more timeline creation task? Yes No

To submit your responses please click on the following button!

Submit

Figure 14: Completion page in the annotation interface for the Timeline Extraction & Summarization task

Task Instructions

There will be many similar HITs for you to perform if you do well at this task. Please follow the instructions carefully for each HIT; we will be reviewing your HITs periodically and if we note any unusual responses, you might not see any additional tasks from us.

During this task, you will be reading a timeline that consists of a sequence of tweets describing an event and eight different summaries for the event in the timeline. You will rate the quality of each of the eight summaries by four axes: *coherence*, *accuracy*, *coverage*, and *overall quality*.

The rubrics below give specific guidance on how each axis should be rated. Please read the rubrics carefully before continuing to the task.

Coherence

For each summary, answer the question "how coherent is the summary on its own?" (on a scale from 1 to 5). A summary is *coherent* if, when read by itself, it's easy to understand and free of English errors. A summary is not coherent if it's difficult to understand what the summary is trying to say. Generally, it's more important that the summary is understandable than it being free of grammar errors.

Rubric:

- Score of 1: The summary is impossible to understand.
- Score of 2: The summary has many mistakes or confusing phrasing.
- Score of 3: The summary has some mistakes or confusing phrasing that make it hard to understand.
- Score of 4: The summary has only one or two mistakes or confusing phrasing.
- Score of 5: The summary is perfectly clear.

Accuracy

For each summary, answer the question "does the factual information in the summary accurately match the information in the timeline?" (on a scale of 1 to 5). A summary is *accurate* if it doesn't say things that aren't in the timeline, it doesn't mix up information, and generally is not misleading.

Rubric:

- Score of 1: The summary is completely wrong, made up, or exactly contradicts what is written in the timeline.
- Score of 2: The summary says many things not mentioned in or contradicting the timeline.
- Score of 3: The summary says at least one substantial thing that is not mentioned in the timeline, or that contradicts something in the timeline.
- Score of 4: The summary says anything at all that is not mentioned in the timeline or contradicts something in the timeline.
- Score of 5: The summary has no incorrect statements or misleading implications.

Figure 15: Instruction page of the Summarization Quality Estimation Task part 1.

Coverage

For each summary, answer the question "how well does the summary cover the important information in the timeline?" (on a scale of 1 to 5). A summary has good *coverage* if it mentions the main information from the timeline that's important to understand the event described in the timeline. A summary has poor coverage if someone reading only the summary would be missing several important pieces of information about the event in the timeline.

Rubric:

- Score of 1: The summary contains no information relevant to the timeline.
- Score of 2: The summary is missing many important pieces of information required to understand the timeline.
- Score of 3: The summary is missing at least one crucial piece of information required to understand the timeline.
- Score of 4: The summary is missing any information (no matter how small) required to understand the timeline.
- Score of 5: The summary covers all of the important information required to understand the timeline.

Overall quality

For each summary, answer the question "how good is the summary overall at representing the timeline?" (on a scale of 1 to 5). This encompasses all of the above axes, as well as the information included in the summary and if it has helped you understand the event. If it's hard to find ways to make the summary better, give the summary a high score. If there are lots of different ways the summary can be made better, give the summary a low score.

Rubric:

- Score of 1: The summary is terrible.
- Score of 2: The summary is a pretty bad representation of the timeline and needs significant improvement.
- Score of 3: The summary is an okay representation of the timeline, but could be significantly improved.
- Score of 4: The summary is a pretty good representation of the timeline, but it's not perfect.
- Score of 5: The summary is an excellent representation of the timeline.

Example

Now you will review an example timeline and two associated summaries.

- For each of the summaries, we have provided ratings on the four axes: coherence, accuracy, coverage, and overall quality with explanations for why those ratings were chosen.
- Please review the summaries and their ratings carefully, so you understand how to rate the summaries during the task.
- If you have any questions about how to rate the summaries, please consult the rubric (above).

Figure 16: Instruction page of the Summarization Quality Estimation Task part 2.

Example timeline

2021-06-28 13:58:14
STREET FLOOD ADVISORY now for Northern Miami-Dade issued until noon. About 2 inches of rain have fallen in the area. Slow-moving rain and storms around. Avoid driving through flooded streets. @wsvn @7weather #flwx <https://t.co/hBa2Tfzsl0>

2021-06-28 14:03:58
Flood Advisory for parts of Central and Southern Miami-Dade please drive safe! <https://t.co/khCkpDlIXe>

2021-06-28 14:11:23
MFL continues Flood Advisory for Miami-Dade [FL] till Jun 28, 11:15 AM EDT <https://t.co/lln0oiMTHx> <https://t.co/zGfJyPyohA>

2021-06-28 14:14:29
June 28th @ 10:15am - Heavy rainfall continues to fall across portions of central Miami-Dade county. Localized flooding is ongoing as 2-3+ inches of rain have fallen in a relatively short period of time. Anyone seeing any flooding right now? #flwx <https://t.co/id7hnG4pYn>

2021-06-28 14:31:01
FLASH FLOOD WARNING issued for Central Miami-Dade until 1:30 pm. 2 to 3 inches of rain have already fallen. More rain to come located over Biscayne Bay. Getting reports of ongoing flooding in the area. Please don't drive through flooded roadways. @wsvn @7weather #flwx <https://t.co/BDafrJb7Xp>

2021-06-28 19:09:03
A storm over NW Miami-Dade has the potential for gusty winds to 45 mph and small hail. It's traveling NW at 20 mph & may impact the SW suburbs of Borward County thru 3:30 pm <https://t.co/KEUkneh09>

How many tweets are there in the timeline?

Less than 5 5 to 10 10 to 20 More than 20

Example summary #1

A street flood advisory has been issued for Northern Miami-Dade after 2-3 inches of rain has fallen in the area. The advisory will last until Jun 28, 11:15 AM. A storm is also going over northwest Miami-Dad and has potential for winds up to 45mph and small hail.

Ratings for example summary #1

The summary should be rated as follows:

How coherent is the summary on its own?
It is impossible to understand. 1 2 3 4 5 The summary is perfectly clear.
Explanation for rating: This summary is easy to understand and read with clear language and no grammatical errors and therefore coherent, so we rate it as 5 (of 5).

Figure 17: Instruction page of the Summarization Quality Estimation Task part 3.

Does the factual information in the summary accurately match the timeline?
The summary is completely wrong, made up, or exactly contradicts what is written in the timeline. 1 2 3 4 5 The summary has no incorrect statements or misleading implications.
Explanation for rating: This summary contains an incorrect statement by stating that the flash flood warning was issued for Northern, Central, and Southern Florida instead of Miami-Dade, so we rate it as 3 (of 5).

How well does the summary cover the important information in the timeline?
The summary contains no information relevant to the timeline. 1 2 3 4 5 The summary covers all of the important information required to understand the event in the timeline.
Explanation for rating: The summary doesn't mention that a flash flood warning was issued for Miami-Dade and it doesn't mention that the storm has potential winds and that it's traveling NW, which could impact suburbs of Borward County, so we rate it as 3 (of 5).

How good is the summary overall at representing the timeline?
It is terrible. 1 2 3 4 5 It is an excellent representation of the timeline.
Explanation for rating: This summary contains grammatical errors, inaccurate information, and it doesn't cover the important information about the event described in the timeline, so we rate it as 3 (of 5).

Test

Answer the following question to start the task. If you are unsure of the answer, review the rubrics above. The task section will appear when you've completed the test.

Which axis measures whether the summary contains all important information from the timeline? Enter your answer and click "Start task".

Figure 18: Instruction page of the Summarization Quality Estimation Task part 4.

Coverage Correct

Timeline

2021-09-18 00:00:46+00:00
warning: SAFETY ALERT warning: Please stay away from the areas of Happy Valley Rd, Sanctuary Ln, Inspiration Ter, and Callippe Golf Course for a fire. Emergency Personnel on-scene and evacuations are underway. Updates to follow. #PleasantonFire https://t.co/9kQNVcXia

2021-09-18 00:06:14+00:00
#BREAKING : SKY7 live over new fire in #Pleasanton along I-680. Evacuations ordered at Happy Valley Rd, Inspiration Terrace & Sleepy Head Lane. #BayArea #fire https://t.co/TalwvSPNn

2021-09-18 00:15:05+00:00
JUST IN: Crews quickly contained a brush fire burning near I-680 in #Pleasanton that forced lane closures and prompted evacuations Friday afternoon, according to officials https://t.co/9kQNVcXia

2021-09-18 01:20:01+00:00
ONGONG: Brush fire on Northbound I-680 South of Bernal Ave in Pleasanton. Right Lane Remains Blocked. https://t.co/1TRwLs14h

2021-09-18 01:35:42+00:00
UPDATE: Crews Contain Pleasanton Vegetation Fire Near Interstate 680; Evacuations Lifted https://t.co/1ax2sj2mgg

Summaries

Test

Answer the following question to start the rating task. The rating section will only appear when you've completed the test.

How many there tweets are in the timeline? Select your answer and click "Start Rating".

Less than 5
 5 to 10
 10 to 20
 More than 20
 [Start Rating](#)

[Submit](#)

Figure 19: Test page of the Summarization Quality Estimation Task.

Coverage Correct

Timeline

2021-09-18 00:00:46+00:00
warning: SAFETY ALERT warning: Please stay away from the areas of Happy Valley Rd, Sanctuary Ln, Inspiration Ter, and Callippe Golf Course for a fire. Emergency Personnel on-scene and evacuations are underway. Updates to follow. #PleasantonFire https://t.co/9kQNVcXia

2021-09-18 00:06:14+00:00
#BREAKING : SKY7 live over new fire in #Pleasanton along I-680. Evacuations ordered at Happy Valley Rd, Inspiration Terrace & Sleepy Head Lane. #BayArea #fire https://t.co/TalwvSPNn

2021-09-18 00:15:05+00:00
JUST IN: Crews quickly contained a brush fire burning near I-680 in #Pleasanton that forced lane closures and prompted evacuations Friday afternoon, according to officials https://t.co/9kQNVcXia

2021-09-18 01:20:01+00:00
ONGONG: Brush fire on Northbound I-680 South of Bernal Ave in Pleasanton. Right Lane Remains Blocked. https://t.co/1TRwLs14h

2021-09-18 01:35:42+00:00
UPDATE: Crews Contain Pleasanton Vegetation Fire Near Interstate 680; Evacuations Lifted https://t.co/1ax2sj2mgg

Summaries

Summary 1 / 8

Pleasanton Fire causing evacs at Happy Valley Rd, Inspiration Terrace & Sleepy Head Lane. The fire is burning near I-680, Right Lane Remains Blocked. Crews quickly contained the brush fire. Evacuations LIFTED.

How **coherent** is the summary on its own?
view rubric for [Coherence](#)

It is impossible to understand. The summary is perfectly clear.
1 2 3 4 5

Does the factual information in the summary accurately match the timeline?
view rubric for [Accuracy](#)

The summary is completely wrong, made up, or exactly contradicts what is written in the timeline. The summary has no incorrect statements or misleading implications.
1 2 3 4 5

How well does the summary **cover** the important information in the timeline?
view rubric for [Coverage](#)

The summary contains no information relevant to the timeline. The summary covers all of the important information required to understand the situation.
1 2 3 4 5

How good is the summary **overall** at representing the timeline?
view rubric for [Overall quality](#)

It is terrible. It is an excellent representation of the timeline.
1 2 3 4 5

[Submit](#)

Figure 20: Rating page of the Summarization Quality Estimation Task part 1.

Coverage Correct

Timeline

2021-09-18 00:00:46+00:00
warning: SAFETY ALERT warning: Please stay away from the areas of Happy Valley Rd, Sanctuary Ln, Inspiration Ter, and Callippe Golf Course for a fire. Emergency Personnel on-scene and evacuations are underway. Updates to follow. #PleasantonFire https://t.co/9kQNVcXia

2021-09-18 00:06:14+00:00
#BREAKING : SKY7 live over new fire in #Pleasanton along I-680. Evacuations ordered at Happy Valley Rd, Inspiration Terrace & Sleepy Head Lane. #BayArea #fire https://t.co/TalwvSPNn

2021-09-18 00:15:05+00:00
JUST IN: Crews quickly contained a brush fire burning near I-680 in #Pleasanton that forced lane closures and prompted evacuations Friday afternoon, according to officials https://t.co/9kQNVcXia

2021-09-18 01:20:01+00:00
ONGONG: Brush fire on Northbound I-680 South of Bernal Ave in Pleasanton. Right Lane Remains Blocked. https://t.co/1TRwLs14h

2021-09-18 01:35:42+00:00
UPDATE: Crews Contain Pleasanton Vegetation Fire Near Interstate 680; Evacuations Lifted https://t.co/1ax2sj2mgg

How **coherent** is the summary on its own?
view rubric for [Coherence](#)

It is impossible to understand. The summary is perfectly clear.
1 2 3 4 5

Does the factual information in the summary accurately match the timeline?
view rubric for [Accuracy](#)

The summary is completely wrong, made up, or exactly contradicts what is written in the timeline. The summary has no incorrect statements or misleading implications.
1 2 3 4 5

How well does the summary **cover** the important information in the timeline?
view rubric for [Coverage](#)

The summary contains no information relevant to the timeline. The summary covers all of the important information required to understand the situation.
1 2 3 4 5

How good is the summary **overall** at representing the timeline?
view rubric for [Overall quality](#)

It is terrible. It is an excellent representation of the timeline.
1 2 3 4 5

[Back](#) [Next](#)

[Submit](#)

Figure 21: Rating page of the Summarization Quality Estimation Task part 2.