Crisis2Sum: An Exploratory Study on Disaster Summarization from Multiple Streams

Philipp Seeberger
Technische Hochschule Nürnberg
philipp.seeberger@th-nuernberg.de

Korbinian Riedhammer
Technische Hochschule Nürnberg
korbinian.riedhammer@th-nuernberg.de

ABSTRACT

Automatic summarization of natural and human-made disaster events is an important area to increase situational awareness for human response organizations and disaster management. However, the incorporation of multiple data sources poses a challenge to current summarization systems, as the typically large document collections exceed the input limits of neural models. Additionally, Large Language Models (LLM) often omit key information present at different positions in long context inputs. Furthermore, disaster reporting requires fine-grained information content and therefore relaxes the restriction to high compression rates, resulting into rather long summaries. In this work, we study different extractive and LLM-based abstractive baselines and highlight shortcomings in present approaches. Our experimental results on the CrisisFACTS datasets show that LLM-based approaches tend to fail in generating long informative summaries. Taking these limitations into account, we propose a disaster summarization framework and introduce query-focused extensions, which demonstrate advantages and superior performance over the baseline methods.

Keywords

Crisis Informatics, Disaster Summarization, Large Language Models, Integer Linear Programming

INTRODUCTION

Disaster management is a critical process responsible for various key phases (e.g., preparedness, response, and recovery) during mass-emergency incidents, where human decisions heavily rely on a comprehensive understanding of the specific event. Situational awareness aims to provide an ideal understanding of what is happening during an event with numerous actors and sub-events (Vieweg et al., 2010). However, both the lack of information and the overload due to today's information ecosystem can paralyze disaster management and humanitarian response organizations (Imran et al., 2014; McCreadie & Buntain, 2023). It is therefore crucial to automatically localize valuable information and condense it into summary reports that can effectively contribute to situational awareness. Emerging heterogeneous information sources such as social media and microblogging platforms rapidly disseminate crucial details about ongoing events and offer new opportunities to complement traditional approaches (Imran et al., 2014). However, the high-velocity nature of content generation and the properties of different source domains and events pose challenges to current disaster reporting approaches (Kaufhold, 2021; Seeberger & Riedhammer, 2022b).

Unlike most other summarization tasks, we identify several key challenges for disaster reporting: (1) The lack of annotated datasets limits the application of supervised methods and makes it difficult to train sophisticated end-to-end models. (2) Compared with classical query-focused summarization, the information need about an event is complex and requires considering up to dozens of queries simultaneously, depending on the disaster type (McCreadie & Buntain, 2023). For example, it is necessary to capture many aspects such as the burned area, missing persons, contamination level, and evacuation orders during a wildfire event. (3) The multi-stream setting leads to large collections of noisy documents that cause issues regarding the length limitations of recent transformer-based models (Lewis et al., 2020). (4) The target summaries of most summarization tasks are rather short (e.g., up to 250 tokens) and the generation of long summaries (e.g., reports) remains challenging for abstractive methods (Chang et al., 2023; Kryscinski et al., 2022). Recent advances in pre-trained and instruction-following LLMs (Touvron

et al., 2023) as well as distant supervision (Y. Xu & Lapata, 2020) offer opportunities to address these issues; however, they are relatively unexplored in the disaster summarization domain.

In this work, we explore several LLM-based summarization frameworks for generating disaster reports, aiming to highlight new developments across disasters in the form of event timelines (McCreadie & Buntain, 2023). Due to the lack of appropriate training data, we rely on zero-shot and distant supervision methods while focusing on recent open-source LLM developments.

Contributions

(1) Disaster Summarization Framework To address the identified challenges, we design a disaster summarization framework that unifies and extends best practices from different related summarization domains. Automatic evaluation results show improvements over baselines but also highlight shortcomings related to summary generation lengths.

(2) Query-Focused Selection (QILP) We introduce query-focused extensions to an ILP-based extractive summarizer and report absolute improvements of up to 7.65 points for ROUGE-2 w.r.t. NIST reference summaries. In addition, we suggest to use QILP for subsequent selection of abstractive event nuggets and show improvements for the majority of evaluation metrics.

(3) Evaluation & Ablation Study We compare our method with several introduced extractive and abstractive baselines and underscore limitations regarding reference summaries. These are mainly connected to extractive and abstractive biases introduced due to the NIST reference summary creation process. Furthermore, we provide insights into the impact of document sources as well as retrieval pipelines.

RELATED WORK

Query-Focused Summarization

Query-focused summarization aims to generate a summary of given documents that satifies the information need of a specific query. This query often represents keywords, short titles, or long narratives that may cover multiple aspects. Due to the limited availability of query-specific datasets, initial work in this area focused on unsupervised extractive approaches (Litvak & Vanetik, 2017; Wan, 2008; Wan et al., 2007) and later utilize cross-task knowledge from question answering (QA) datasets (Egonmwan et al., 2019; Laskar et al., 2020; Su et al., 2020, 2021; Y. Xu & Lapata, 2020). Another line of research focuses on the utilization of generic summarization tasks by employing pseudo-queries (Vig et al., 2022; Y. Xu & Lapata, 2021) or latent query modeling (Y. Xu & Lapata, 2022). According to Vig et al., most abstractive summarization models follow an extract-then-abstract pipeline and can be categorized into two-step approaches, consisting of an extractor model that scores the source documents and an abstractor model that consumes the documents up to the input length limit (Vig et al., 2022). In contrast, end-to-end approaches aim to address shortcomings due to the performance of the extractor and achieve superior performance for several benchmarks (Vig et al., 2022). However, end-to-end models require appropriate training data and mechanisms that can handle long input texts. Moreover, most approaches focus on tasks with only one or a small number of queries, which is different from the task of summarizing disasters. Our work builds on two-stage models to cope with the lack of annotations as well as noisy and large data streams that are common in disaster events.

Long Document Summarization

The long document summarization field primarily addresses the input length limitations of neural abstractive models. Recent work can be broadly categorized into sparse attention, extract-then-abstract, divide-and-conquer, and hierarchical models (Mao et al., 2022). Sparse attention models such as Longformer (Beltagy et al., 2020), BigBird (Zaheer et al., 2020), and Reformer (Kitaev et al., 2020) utilize different attention mechanisms to reduce the quadratic memory cost of full attention. Several hierarchical models have been proposed for long input processing including models such as HAT-Bart (Rohde et al., 2021) and HMNet (Zhu et al., 2020) which take hierarchical structures into account. Extract-then-abstract methods are similar to the previously mentioned query-focused two-step approach. Most of these two components are trained separately (Bajaj et al., 2021; J. Xu & Durrett, 2019; Y. Zhang et al., 2021), but some work attempts to bridge the two stages by reinforcement learning (Bae et al., 2019; Chen & Bansal, 2018) or joint training (Mao et al., 2022). Divide-and-conquer models break long text inputs into multiple parts, which are summarized separately and then combined into a final summary (Gidiotis & Tsoumakas, 2020; Grail et al., 2021; Y. Zhang et al., 2022). Most recent work perform zero-shot LLM approaches and follow

hierarchical merging and incremental updating strategies (Chang et al., 2023). Hierarchical merging divides the input into several chunks and combines multiple levels of summaries until a desired length (Wu et al., 2021). In contrast, incremental updating aims to overcome the missing context of hierarchical merging by iteratively updating an existing summary with new input chunks (Adams et al., 2023). In our work, we explore divide-and-conquer approaches since they can be easily applied in a zero-shot setting. As previously mentioned, we also rely on the retrieve-then-abstract model. This enables the processing of large and noisy document collections.

Disaster-Related Summarization

Disaster-related methods mostly cover the summarization of social media posts or web news for mass-emergency events. Initial work proposed models for summarizing Twitter posts, which often consist of a classification and subsequent summarization step. The classification step involves the pre-filtering of noise and assignment into predefined categories, which are then summarized into specific or general summaries (P. K. Garg et al., 2023; Rudra et al., 2015, 2018, 2019). Rudra et al. introduced a pipeline consisting of a classifier followed by an extractive summarizer formulated as Integer Linear Programming (ILP) that aims to maximize content words and sub-events coverage w.r.t. specific categories (Rudra et al., 2018). Nguyen and Rudra employs a rationale-motivated classifier to extract rationales that are further used for ILP-based summarization (Nguyen & Rudra, 2022). For temporal summarization, other work focuses on saliency prediction and mechanisms that utilize similarity- or cluster-based methods for summarization (Dusart et al., 2021, 2023; Kedzie et al., 2015). Apart from temporal summarization studies such as TREC TS (Aslam et al., 2015), real-time summarization, with a focus on mobile device notifications and daily email digests from Twitter streams, has gained attention (Sequiera et al., 2018). Most recent work is centered around multi-source datasets, such as CrisisFACTS (McCreadie & Buntain, 2023), and the use of reinforcement learning (Cambrin et al., 2024) or pre-trained LLMs for query-focused disaster reporting (Pereira et al., 2023; Seeberger & Riedhammer, 2022a, 2023). Pereira et al. builds on QA-motivated methods and uses chain-of-thought (CoT) to extract relevant facts from retrieved stream items (Pereira et al., 2023) while FloodBrain (Colvert et al., 2023) aims to generate flood-related reports from web-based content using retrieval-augmented generation. However, classification-based pipelines are limited to predefined categories while summary aspects differ across events. Therefore, we focus on distant supervision for the extractive component. Since summarization on concept level has shown promising results in different domains (Ernst et al., 2022; Riedhammer et al., 2010), we use QA-motivated CoT for fact extraction.

METHOD

In the following, we define our framework for disaster summarization and propose components to overcome the challenges of noisy and large document collections. We give an overview of our approach in Figure 1. In the next section, we formulate our task and present the two-stage pipeline. Then, we detail the Extractor and Abstractor components. Lastly, we introduce extensions to an ILP-based selection model adapted to the query-focused case.

Retrieve-then-Summarize

In disaster summarization, the input consists of $\mathcal{D} = \{d_1, \ldots, d_N\}$ as a set of event-related documents and $Q = \{q_1, \ldots, q_M\}$ as a set of queries with N and M as the number of documents and queries, respectively. Each query $q \in Q$ typically consists of a short natural text or a list of keywords and covers a topic or aspect (e.g., wildfire containment). The set of queries Q aims to fulfill the user's information need that is composed of various of these event aspects. Since the initial corpus is often large and still contains irrelevant documents, we employ an Extractor to retrieve and reduce the set of documents to query-related clusters $C = \{C^{(q_1)}, \ldots, C^{(q_M)}\}$ with $C^{(q_i)} = \{d_1^{(q_i)}, \ldots, d_k^{(q_i)}\}$ consisting of the top-k candidate documents ranked by relevance score. Then, optionally, each cluster $C^{(q_i)}$ is further refined by a reranking model which operates over all retrieved document-query pairs to estimate semantic matching scores. The Abstractor generates an output summary S (e.g., list of event nuggets) from the cluster pool $\bigcup_{i=1}^{M} C^{(q_i)}$. Following the view of event timeline summarization, each summary S_t is created w.r.t. a time period $p_t \in \{p_1, \ldots, p_T\}$ with T as the number of time periods and $\mathcal{D}_t \subseteq \mathcal{D}$ as the subset of documents created within the time period p_t. For the sake of clarity, however, we omit the time index t, consider each time period as an independent summarization task and leave redundancy issues for future work.

Extractor

How to model long texts and large document collections is an ongoing research area and various techniques have been proposed to overcome length and GPU memory limitations of neural models by independently processing chunks. It is common practice to use preprocessing functions such as truncation, chunking, and content selection

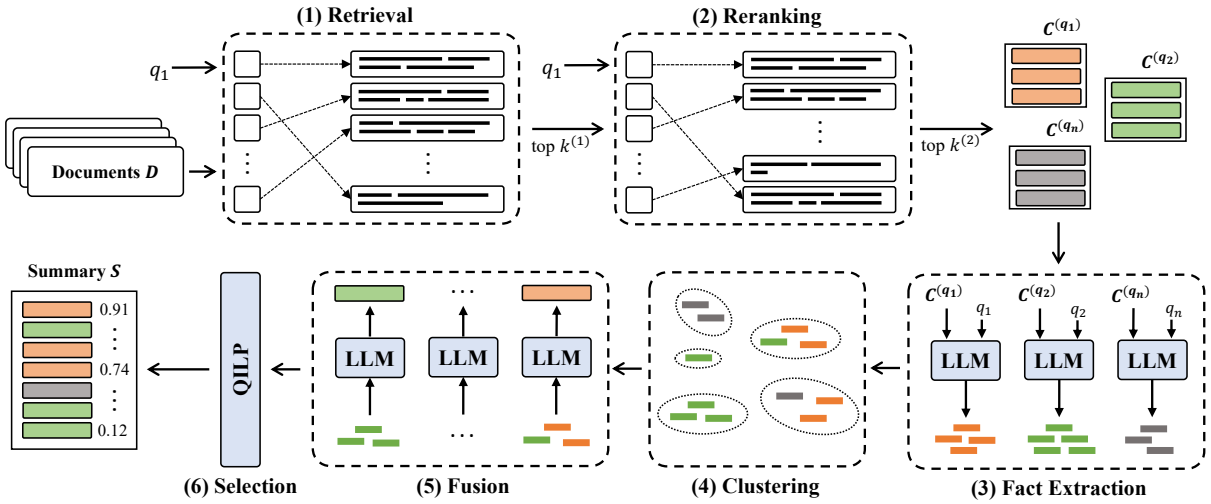

Figure 1. Overview of our framework for disaster summarization. The Extractor consists of stages (1) and (2) while the Abstractor is composed of stages (3), (4), and (5). Stages (1) and (2) retrieve and rerank all documents and extract query-related clusters. (3) generates the query-related facts and (4) clusters the facts into topic groups. Subsequently, (5) fuses the topic clusters to event nuggets. Optionally, (6) selects the event nuggets to create the final summary.

Reranked Documents

- lilac fire: WB SR-76 remains closed at I-15. EB SR-76 is closed at E Vista Way. Power remains out in some areas. **(score: 0.95)**
- Lilac Fire Update: Carlsbad Shelter Closed; Fire Containment Now at 20 Percent- San Diego County Emergency **(score: 0.67)**

⋮

- Does anyone know if the 76 is open or if it'll be opening soon? lilac fire **(score: 0.32)**
- More sobering was news that the number of homes destroyed by the fire had risen to 182, and 23 damaged **(score: 0.13)**

Fact Extraction

Input

{Reranked Documents}

What roads are blocked or closed?

- State Route 76 is closed in both directions from Interstate 15 to East Vista Way
- Route 306 is still impacted by the lilac fire
- State Route 76 remains closed due to damage
- Road closures on the 76 from the lilac fire

Clustering & Fusion

Topic SR-76

- State Route 76 is closed in both directions from Interstate 15 to East Vista Way
- EB SR-76 is closed at E Vista Way
- State Route 76 remains closed due to damage

Event nugget: State Route 76 is currently closed in both directions between Interstate 15 and East Vista Way due to damages.

Figure 2. Example for the event nugget creation process w.r.t. the query *What roads are blocked or closed?* and Lilac Wildfire event.

(Dong et al., 2023) while salient content selection can be achieved with retrieval-based models. In our work, we employ content selection methods since text documents from web news, social media, or microblogging platforms are typically short, often irrelevant, and occur in large volumes.

Based on the assumption that salient information occupies only a small portion of long documents (Ding et al., 2020), we employ a two-stage retrieval pipeline as our Extractor in order to reduce the set of documents $\mathcal{D}$ to the salient sets of query-related clusters C. As initial step, we retrieve the top-$k^{(1)}$ candidates for each query $q \in Q$ using an efficient sparse retrieval model. Then, we process the concatenation of each query-document pair and obtain a scalar score $s_{ij} = E(q_i, d_j)$, where $E(\cdot)$ represents a neural reranker model and s_{ij} the document relevance w.r.t. a query q_i. Then, the top-$k^{(2)}$ candidates are selected by descending order of the scores, resulting into a reranked set C'. In Figure 2, we can see that relevant documents achieve higher relevance scores for the query *What roads are blocked or closed?*.

However, the event-related documents may cover different aspects of the event and therefore can obtain multiple high relevance scores for different queries. If we require a single saliency score (e.g., for extractive summarization), we address this issue by a fusion function $f(\cdot)$ to obtain a final saliency score s_j for a document d_j. There exist several rank- and score-based fusion techniques, but in preliminary experiments, we found CombSUM (Fox & Shaw, 1994) to be an effective fusion method. CombSUM calculates the final saliency score for a document as $s_j = \sum_i s_{ij}$ by summing up the relevance scores for each query. As depicted in Figure 1, the Extractor consists of the main stages (1) Retrieval and (2) Reranking.

Abstractor

In contrast to recent long document processing approaches, we follow a QA-motivated[1] method to enable *event nugget*-based summarization. We first extract all query-relevant facts for the documents and then generate event nuggets employing a clustering-based summarization approach (Ernst et al., 2022). Inspired by (Pereira et al., 2023), we use an instruction-following LLM to generate our facts. Therefore, we utilize the CoT technique and prompt the model to provide a list of facts which serve as evidence for answering the query q based on the provided documents $\{d_1, \ldots, d_j\}$ with $d_i \in C^{(q)}$. The exact prompt is shown in Table 10. For example, we prompt the LLM with the query *What roads are blocked or closed?* and obtain the facts only relevant to this query (Figure 2).

In the next step, all extracted facts are clustered into semantically similar groups. This further prevents redundancy by avoiding multiple facts covering the same piece of information (e.g., *State Route 76 closed*). For clustering, we compute the cosine similarity for each pair of facts using sentence embeddings from a transformer model and then apply affinity propagation. We opt for affinity propagation since the approach does not require specifying the number of clusters and has demonstrated state-of-the-art performance in the disaster summarization domain (Kedzie et al., 2015). Lastly, the resulting documents of the topic clusters are merged with an additional summarization step using the identical LLM (Table 5). Simply selecting the cluster exemplars has shown insufficient results as this approach would discard similar sub-events with different entities such as locations.

For event nugget importance, we recompute the relevance scores for each event nugget and utilize the saliency scores for ranking. That is, each generated event nugget is fed into the neural reranker model, and the resulting scores are used for saliency. The final summary is produced by appending the ranked event nuggets until the desired summary length or by using an optional extractive selection model that we detail in the next section.

Query-Focused Selection

As extractive selection model, we extend previous concept-based methods and model the extractive summarization as an ILP problem (Gillick & Favre, 2009; Riedhammer et al., 2010; Rudra et al., 2018; Seeberger & Riedhammer, 2022a). Concepts typically represent units such as bi-grams, entities, and key-phrases. Our objective is to maximize the coverage of these units while keeping redundancy low. In the rest of this paper, we refer to our extensions as Query-Focused Integer Linear Programming (QILP). The introduced extensions aim to incorporate document saliency and query coverage on the summary level. Note that the extractive summarization can be applied on both the documents directly and generated event nuggets. In particular, we maximize the following objective:

$$\underset{d}{\text{argmax}} \quad \alpha \sum_i w_i c_i + \beta \sum_j s_j d_j - \gamma \sum_p \rho_p (1 - z_p) \tag{1}$$

$$\text{subject to} \quad \forall j \sum_j d_j \leq L \tag{2}$$

$$\forall i \sum_j d_j o_{ij} \geq c_i \tag{3}$$

$$\forall i, j \quad d_j o_{ij} \leq c_i \tag{4}$$

$$\forall_i c_i \in \{0, 1\} \quad \forall_j d_j \in \{0, 1\} \quad \forall_k z_k \in \{0, 1\} \tag{5}$$

Concept Coverage and Document Saliency The first and second term in Equation (1) correspond to the concepts coverage and documents saliency, respectively. Formally, let c_i denote the presence of concept i in the summary, d_j denote the presence of document j in the summary, w_i denote the weight of concept i, and s_j denote the saliency of document j. The final score of the concepts coverage and documents saliency is expressed as the sum of positive weights w_i of concepts and scores s_j of documents covered in the summary. If not otherwise mentioned, we use bi-grams as concepts and frequency as weights.

Consistency Constraints Constraints ensure that if a document is selected, all concepts are also selected and if a concept is selected, at least one document that contains it is selected. Additionally, we ensure that the expected summary length do not exceed the defined length L. Here, the variable o_{ij} denotes the occurrence of concept i in document j. In detail, Equation (2) ensures the length constraint and Equation (3) ensures that if a concept i is selected, then there is at least one summary document covering it. Equation (4) ensures that if a document j is selected to be included in the summary, then all concepts in that document are also included in the objective function.

[1]We follow a typical Question-Answering task and extract the answer (facts) for a question (query).

Query Coverage We assume that only a subset of queries are salient depending on the event and day. Thus, we aim to select a characteristic set of documents which covers the salient queries. We already know that each document j is associated with a relevance score s_{ij} w.r.t. to a query q_i. Based upon this, we create a m-dimensional target vector $\tau(\mathcal{D}')$ and summary vector $\pi(S)$ that represents the fraction of documents in $\mathcal{D}'$ and S that cover the queries Q. Here, the set $\mathcal{D}'$ denotes the generated event nuggets for the abstractive and documents for the extractive case. Specifically, $\tau(\cdot, i)$ and $\pi(\cdot, i)$ denote the fraction of documents that cover query q_i. We compute the target and summary vectors as the normalized mean of relevance scores (Lappas et al., 2012). The third term in Equation (1) aims to match the target vector τ by imposing penalities on certain assignments with soft constraints. The variable z_p serves as an indicator if the constraint p is satisfied and ρ_p acts as penality score for constraint violation:

$$\forall p \in Q, \quad z_p = \begin{cases} 1 & \text{if the query contribution } \pi(S, p) \geq \tau(\mathcal{D}', p) \text{ or} \\ 0 & \text{otherwise.} \end{cases} \tag{6}$$

In this way, we reward the model for selecting documents which cover salient queries. However, we want to avoid penalization for ignoring queries with low saliency. Therefore, we apply an adaptive truncation threshold θ and only consider the $k^{tr} = \max_k \sum_i \tau(\mathcal{D}', i) \leq \theta$ queries. For instance, the queries q_1 with *What area has the wildfire burned?* and q_2 with *Where are firefighters needed?* are more salient topics w.r.t. a wildfire event than the query q_3 with *What curfews are in place?*. In this case, we assume the target vector with the corresponding queries q_1, q_2, q_3 is $\tau(\mathcal{D}') = (0.8, 0.6, 0.05)$ and the summary vector should match this target vector. We balance the linear combination of terms in Equation (1) with the hyper-parameters α, β, and γ, which can be any scalar values.

EXPERIMENTAL SETUP

Dataset

For our experiments, we use the CrisisFACTS datasets from 2022 and 2023 (McCreadie & Buntain, 2023). Both datasets include overall 18 disaster events and additional metadata (e.g., search keywords, queries, event types, and source types). Each event-day pair consists of multi-stream data and is extracted from the online sources Twitter, Facebook, Reddit, and Webnews. All documents are provided as stream items. That is, the documents are already segmented into short chunks which we use for our experiments. Along with the muli-stream data, the authors provide ground truth summaries extracted from Wikipedia, ICS-209 forms (Denis et al., 2020), and NIST annotated event nuggets. In Table 4, we provide further details about the number of stream items, days, and event types. More details on the dataset and summary construction process can be found in (McCreadie & Buntain, 2023).

Evaluation

In compliance with (McCreadie & Buntain, 2023), we evaluate the system summaries with ROUGE-1/2/L[2] and BERTScore F1-scores (T. Zhang et al., 2020). System summaries for each event-day pair are constructed by the concatenation of all event nuggets (ordered by importance score) up to a threshold of 32 items. For fair comparison, all event nuggets must not exceed 200 characters and each final summary is bounded to $32 * 200 = 6400$ characters. However, we found that 83% (1691 tokens on average) of the NIST reference summaries exceed the token limit 512 of the authors choosen BERTScore model DeBERTa[3]. In other words, only a fraction of the most important event nuggets is evaluated from a semantic perspective. For completeness, we evaluate and report with BART[4] instead but emphasize that the metrics are still flawed by truncation. This model has a context size of 1024 and doubles the event nugget coverage. Other models with larger context size have been reported with no distinguishing power (Bertsch et al., 2023).

Model Details

For the Extractor component, we use BM25 (Robertson & Zaragoza, 2009) with default settings of the *PyTerrier* (Macdonald & Tonellotto, 2020) library and extend it with Bo1 (Amati & Van Rijsbergen, 2002) query expansion to increase the recall. We set the number of feedback terms and documents as 3 and 20, respectively. As reranker, we employ MonoT5[5] (Nogueira et al., 2020) trained on MS MARCO and select the top-$k^{(1)}$=250 and $k^{(2)}$=50

[2]We compute all ROUGE scores with the *TorchMetrics* library and use the *PorterStemmer* as preprocessing step.
[3]https://huggingface.co/microsoft/deberta-xlarge-mnli
[4]https://huggingface.co/facebook/bart-large-mnli
[5]https://huggingface.co/castorini/monot5-large-msmarco-10k

candidates for each event-day-query triple. If not otherwise specified, all models and baselines use the same candidate documents as well as relevance and saliency scores.

Regarding the ABSTRACTOR component, we choose the *Transformers* (Wolf et al., 2020) library and use MISTRAL-7b[6] (Jiang et al., 2023) with greedy decoding as our instruction-following LLM. To increase the context, we select the left and right documents of each candidate document up to a limit of 500 characters. Due to computational constraints, we limit the number of input tokens for all considered methods to 4096. In terms of clustering, we utilize a *Sentence Transformers* model[7] to compute the sentence embeddings and cosine similarity between fact-pairs. We provide further prompt details in the Appendix Prompt Details.

For QILP, we use bi-grams and frequency as concepts and weights, respectively (Gillick & Favre, 2009; Riedhammer et al., 2010). All scores are normalized and the hyper-parameters are set to $\alpha = 1$, $\beta = 1$, $\gamma = 5$, $\rho = 1$, and $\theta = 0.7$ which we selected based on the TREC TS dataset (Aslam et al., 2015). We refer to our final models as follows: QILP[8] denotes the extractive approach that consists of the EXTRACTOR and QILP summarizer. LLMNUG-CONCAT consists of the EXTRACTOR and ABSTRACTOR based on simple fact concatenation up to the event nugget character limit. That is, we skip the clustering and fusion steps in Figure 1 and use multiple facts to present one event nugget. LLMNUG-CONCAT w/ QILP extends the previous model with QILP for final selection. LLMNUG-CLUSTERFUSE employs the proposed fact extraction and cluster fusion while LLMNUG-CLUSTERFUSE w/ QILP represents the extension with QILP selection.

Baseline Systems

We aim to establish an upper bound for the comparison of our models as well as extractive and abstractive baselines. Here, we only focus on NIST-based reference summaries since ICS forms are rather long and do not provide any splits for event-day pairs. We generate extractive oracles with a greedy search and denote them as ORACLE. Specifically, we start with an empty set and iteratively select documents such that the concatenation maximizes the average of ROUGE-1/2 scores given the reference summary. For feasiblity reasons, we limit the search space to the documents retrieved by our EXTRACTOR component. Furthermore, we include a LLaMA-based[9] (Touvron et al., 2023) submission that achieved strong results in the CrisisFACTS 2023 challenge and denote it as $\text{LLAMA}_{\text{TREC}}$. $\text{LLAMA}_{\text{TREC}}$ similarly prompts query-relevant facts with a QA-motivated approach and concatenates the resulting facts to event nuggets (Seeberger & Riedhammer, 2023). In the following, we introduce the extractive and abstractive baselines.

Extractive We compare with several heuristics and extractive baselines from previous query-focused summarization work. More specifically: RANDOM uses random selection, LEAD selects the top document from each query-related cluster, and GREEDY selects the overall top ranked documents (McCreadie & Buntain, 2023). As extractive models, we include the query-focused LEXRANK (Y. Xu & Lapata, 2020), a variant of SUBMOD (Dasgupta et al., 2013) composed of a linear combination of facility-location, document relevance, dispersion, and cluster contribution and lastly SCC (Rudra et al., 2018) as an ILP-based summarization framework with content word and sub-event coverage maximization. Furthermore, we conduct an ablation study for the different components of our introduced extractive model QILP.

Abstractive We perform experiments with different truncation-based and long document summarization versions while using the identical LLM with zero-shot prompting. For truncated baselines, we include SUM as vanilla summarization, SUMCOT with the two-step CoT summarization approach (Wang et al., 2023), and SELECT+SUM as combination of QILP (L=100) with SUM (Dong et al., 2023). We truncate all inputs w.r.t. the maximal input size of 4096 and ensure that the truncation considers document boundaries. For long document summarization, we include incremental updating and hierarchical merging (Chang et al., 2023). For INCREMENTAL updating, we iterate through each chunk of documents while continuously updating a global summary with only new salient information. For HIERARCHICAL merging, we generate summaries for each input chunk and recursively merge the summaries until we have reached one final summary. We restrict these models to the top 500 documents emitted by the EXTRACTOR component.

[6]https://huggingface.co/mistralai/Mistral-7B-Instruct-v0.2
[7]https://huggingface.co/sentence-transformers/all-mpnet-base-v2
[8]We use the GLPK 5.0 solver.
[9]https://huggingface.co/meta-llama/Llama-2-13b-chat-hf

Table 1. Rouge-1/2/L and BERTScore F_1-score (x100) results on NIST reference summaries for CrisisFACTS 2022 (NIST-22) and 2023 (NIST-23). $|W|$ represents the average word count of the system summaries. Bold numbers indicate the best performance whereas underlined numbers denote the best performance for the extractive and abstractive baseline systems.

		ROUGE-1		ROUGE-2		ROUGE-L		BERTScore	
	$\|W\|$	NIST-22	NIST-23	NIST-22	NIST-23	NIST-22	NIST-23	NIST-22	NIST-23
Extractive Baselines									
RANDOM	586	33.00	28.19	9.40	8.29	12.56	11.37	58.99	54.96
GREEDY	654	46.42	33.66	24.56	14.11	18.06	13.72	65.06	58.04
LEAD	618	42.94	33.23	19.44	12.71	16.59	13.67	64.43	58.40
LEXRANK	663	45.69	34.35	22.09	13.20	17.40	13.25	64.90	58.42
SUBMOD	656	43.28	33.61	20.09	12.57	16.65	12.75	64.33	58.21
SCC	867	43.89	36.69	16.14	11.18	15.42	12.79	62.83	57.18
Abstractive Baselines									
SUM	357	33.14	26.30	12.32	8.68	14.04	11.71	58.40	56.06
SUMCOT	382	33.80	26.82	12.80	8.39	14.25	11.79	58.26	55.61
SELECT+SUM	355	32.89	26.19	12.42	8.52	14.18	11.82	58.48	55.82
INCREMENTAL	631	39.12	35.24	13.39	11.86	14.72	13.31	59.61	58.95
HIERARCHICAL	555	39.63	30.70	13.12	9.21	15.09	12.31	59.15	56.45
Proposed Approaches									
QILP	710	**47.87**	38.00	**26.07**	**17.49**	**18.21**	14.40	**66.18**	60.52
LLMNUG-CONCAT	717	44.55	38.65	18.59	16.03	16.46	**15.34**	63.87	**62.01**
W/ QILP	759	46.77	**40.14**	19.79	16.48	16.74	15.09	63.82	62.00
LLMNUG-CLUSTERFUSE	742	43.99	37.66	15.11	12.05	15.19	13.41	61.89	60.34
W/ QILP	755	44.31	37.95	15.36	12.25	15.34	13.41	62.47	60.48
$\text{LLAMA}_{\text{TREC}}$	707	44.95	41.43	18.75	25.10	16.82	25.74	63.02	66.01
ORACLE	760	70.37	68.11	63.09	53.54	27.96	29.53	75.46	70.11

MAIN RESULTS

In Table 1, we report the results for NIST reference summaries including the average word count for each generated summary. A general trend is that all proposed methods outperform the extractive and abstractive baseline counterparts. Interestingly, the GREEDY heuristic is competitive to our approaches and even surpasses most of the extractive and abstractive baselines. This finding is in line with the results of (McCreadie & Buntain, 2023). The LEAD model also shows decent results but the RANDOM selection achieved relatively low performance across all metrics which demonstrates the importance of document saliency (i.e., query-relevance) obtained by the EXTRACTOR.

Proposed Models The extractive summaries of our proposed approach QILP achieve better results in comparison to all extractive and abstractive baselines, especially for ROUGE-2 (26.07 and 17.49). Surprisingly, the simple concatenation of generated facts outperforms the computational more expensive clustering-based summarization approach. This highlights that disaster reports are rather focused on informativeness than coherence and fluency. The application of extractive selection also improves the LLM-based models for the majority of reported metrics. Critically, we find that most of the best performing models are extractive for NIST-22 and abstractive for NIST-23. This suggests that the reference summaries are potentially biased towards extractive and abstractive systems which could be caused by the pooling-based annotation process. For example, $\text{LLAMA}_{\text{TREC}}$ was part of the pooling-based evaluation of CrisisFACTS 2023 and therefore shows the best results for NIST-23. This finding also supports the assumption of biased reference summaries. Regarding the extractive ORACLE, we still can see large gaps in terms of ROUGE-1/2/L and BERTScore. These results depict the necessity for further developments in the disaster summarization domain.

ICS-209 and Wikipedia We present the evaluation results for ICS-209 and Wikipedia summaries in Table 2. Overall, we find that the system ranks differ from the NIST reference summaries. Shorter and more general summaries achieve the best results for Wikipedia while longer and more detailed summaries have the best coverage for ICS-209 forms. This is reasonable for this kind of summary formats, where Wikipedia articles provide high-level summaries for a public audience and ICS-209 forms represent informative reports written by emergency-response personnel (McCreadie & Buntain, 2023). This suggests the selection of summarization systems tailored to the

Table 2. Rouge-1/2/L and BERTScore F_1-score (x100) results on ICS-209 forms (ICS-22) and Wikipedia (WIKI) reference summaries. ICS summaries are only available for CrisisFACTS 2022. Bold numbers indicate the best performance. Due to context size limitations, we discard the BERTScore for ICS-22.

	ROUGE-1		ROUGE-2		ROUGE-L		BERTScore	
	ICS-22	WIKI	ICS-22	WIKI	ICS-22	WIKI	ICS-22	WIKI
GREEDY	27.98	8.85	6.24	2.52	9.48	4.58	-	48.86
SCC	28.34	7.79	5.42	2.51	9.21	3.91	-	49.32
INCREMENTAL	28.34	9.85	6.33	3.37	9.87	5.19	-	53.32
HIERARCHICAL	28.06	**10.34**	6.31	**3.61**	10.41	**5.57**	-	**53.71**
QILP	28.22	8.61	6.24	2.90	9.43	4.44	-	49.87
LLMNUG-CONCAT	29.60	8.26	6.92	2.85	10.23	4.48	-	51.87
w/ QILP	30.38	7.96	**7.09**	2.71	10.30	4.29	-	51.35
LLMNUG-CLUSTERFUSE	30.69	7.99	7.08	2.57	**10.61**	4.37	-	51.50
w/ QILP	**30.74**	8.06	7.06	2.74	10.57	4.40	-	51.87

Table 3. Rouge-1/2/L and BERTScore F_1-score (x100) results for the ablation study. We removed or substituted the components of our model and report the absolute increase (↑) or decrease (↓) w.r.t. NIST reference summaries.

	ROUGE-1		ROUGE-2		ROUGE-L		BERTScore	
	NIST-22	NIST-23	NIST-22	NIST-23	NIST-22	NIST-23	NIST-22	NIST-23
QILP	47.87	38.00	26.07	17.49	18.21	14.40	66.18	60.52
w/o Q-COV	↓0.14	↓0.14	↓0.17	↓0.09	↓0.03	↑0.03	↓0.15	↓0.14
w/o SAL	↓5,74	↓0,75	↓7,54	↓1,76	↓2,57	↓0,25	↓2,06	↓0,83
w/o SAL+Q-COV	↓5.98	↓1.42	↓7.65	↓2.21	↓2.37	↓0.59	↓2.12	↓1.08
w/ ENTITIES	↓0,06	↓2,67	↓0,65	↓2,23	↑0,17	↓0,52	↓0,19	↓1,26
w/o RERANKING	↓11,81	↓16,74	↓10,13	↓11,11	↓3,10	↓4,81	↓6,80	↓8,42

end-user and information need. However, we emphasize that all evaluated summarization models are used in the unsupervised and zero-shot setting. Appropriate datasets and model fine-tuning can further improve the results but data collections are costly to obtain in the disaster summarization domain.

Are we gaming the benchmark? If we consider summary lengths, one can easily spot the correlating ROUGE-1/2/L and BERTScore scores. Here, the question arises if simply producing long summaries can be seen as *gaming the benchmark*. In fact, we measure a moderate Spearman's rank correlation of 0.43 (0.48) for ROUGE-2 (BERTScore) and note that length bias in neural summarization (Guo & Vosoughi, 2023; Sun et al., 2019) as well as long document summarization (Chang et al., 2023) evaluation is an active research area. However, we argue that producing long and detailed summaries is a desirable feature in the field of disaster reporting. Therefore, we see our event nugget-based summarization as an extension to obtain more informative summaries, but would also like to emphasize the length limitations of our abstractive baselines. In Appendix Examples, we show excerpts for summaries about the *Kincade Wildfire 2019* event.

FURTHER ANALYSIS

In addition to the main experiments, we further assess the impact of different setups and components. Therefore, we analyse the influence of retrieval pipelines, document sources, and provide an ablation study for the QILP model components.

Influence of Retrieval Figure 3 presents the ROUGE-2 scores for different retrieval pipelines. We conducted experiments for the sparse retrieval models DFREEKLIM (Amati et al., 2011) and BM25, sparse neural search model SPLADE[10] (Formal et al., 2022), and rerankers ASNQ[11] (S. Garg et al., 2020) and MONOT5 with BM25 as the first-stage retriever. The retrieval experiments highlight two important points. First, the usage of reranking

[10]https://huggingface.co/naver/splade-cocondenser-ensembledistil
[11]TANDA RoBERTa-Large ASNQ-Wiki-QA

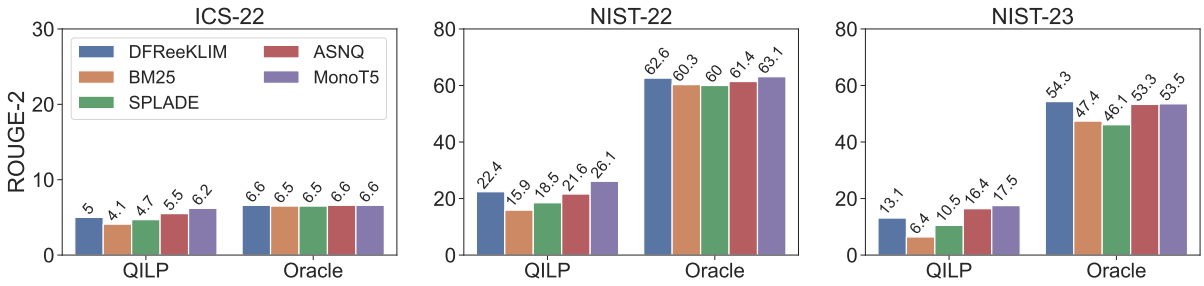

Figure 3. Rouge-2 F_1-score (x100) results for the different retrieval pipelines. We report the performance for QILP and the extractive Oracle. The rerankers ASNQ and MonoT5 use BM25 for first-stage retrieval.

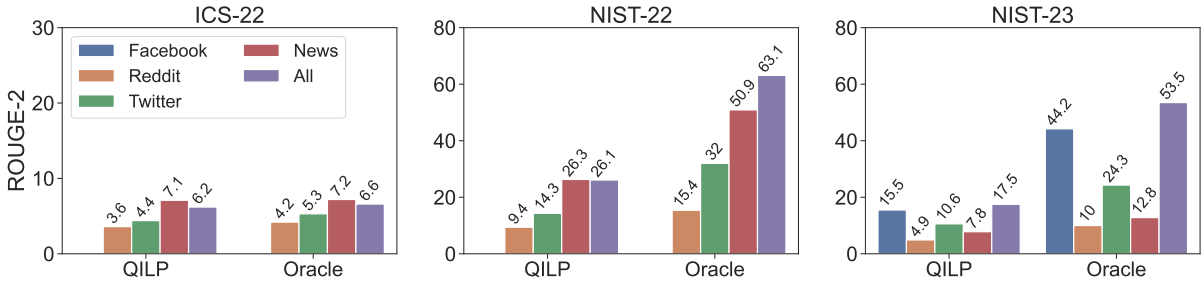

Figure 4. Rouge-2 F_1-score (x100) results for the different available document sources. We report the performance for QILP and the extractive Oracle. The label *All* denotes the mixture of all document sources.

significantly improves the results over the first-stage retrieval and highlights the importance of appropriate document scoring w.r.t. the downstream task. Second, dedicated efficient sparse retrieval models like DFReeKLIM can outperform off-the-shelf models such as BM25 by large margins but still lack behind the two-stage approaches. However, the DFReeKLIM results can be strongly biased due to the pooling-based process as the organizers use two basic DFReeKLIM variants as baseline systems.

Influence of Source Figure 4 presents the ROUGE-2 scores by restricting the input documents to the source types. Note that we only had access to Facebook posts for CrisisFACTS 2023 which explains the missing scores for ICS-22 and NIST-22 summaries. First, we can see that posts from Reddit and Twitter are mostly outperformed by the sources Facebook, News, and All for both QILP and Oracle. We assume that News already summarizes important facts about an event and Facebook often refers to published news articles or include posts written by news accounts (McCreadie & Buntain, 2023). In contrast, Twitter posts contain much noise and Reddit posts are only available in smaller scale. The drop of News for NIST-23 can be explained by both the low number of available stream items and the possible bias towards Facebook posts. This highlights the benefit of multiple document sources as long as the availability of traditional sources is limited.

Influence of Components Table 3 presents the results of our component ablation for the best performing QILP model. As already mentioned in previous sections, removing the reranking component yields the largest drop for all metrics. Exchanging the Bi-gram concepts with Entity concepts leads to an further decrease in performance that mirrors the effectiveness of simple Bi-grams for content coverage. For our saliency (Sal) and query coverage (Q-Cov) extensions, we found that removing saliency leads to a higher performance drop while the largest decrease occurs if we remove both components. In other words, the saliency and query coverage terms can be seen as complementary and not just as substitute. However, we only consider retrieval relevance scores as document saliency and would expect a performance increase by incorporating more sophisticated saliency predictors (Kedzie et al., 2015).

Efficiency Practical utilization of information systems in emergency and disaster response necessitates the deployment of efficient real-time systems. Therefore, we analyze the computation time of the proposed components concerning the event *Lilac Wildfire 2017*. All experiments are conducted with AMD EPYC 7662 CPUs and one Nvidia A100 GPU. On average, the stages of Retrieval and Reranking require 1.2 and 67.4 seconds for each

event-day pair and provide the input for the subsequent summarization components. While the extractive approach QILP only takes 0.8 seconds, the abstractive LLMNug-Concat and LLMNug-ClusterFuse approaches require 673.2 and 839.7 seconds, respectively. This highlights the computational overhead of the incorporated LLMs, which may be insufficient for summarization updates in shorter intervals. Using proprietary LLMs via API access can enhance performance and avoid the need for dedicated infrastructure. However, this also introduces latency issues and may raise data privacy concerns.

CONCLUSION

In this work, we studied the task of disaster summarization from multiple streams. We establish several unsupervised and zero-shot baselines and assess the performance with CrisisFACTS, a state-of-the-art dataset for disaster summarization. Current summarization systems reveal shortcomings related to large document collections and face challenges in generating long disaster reports. To address these issues, we introduce a retrieve-then-summarize approach and extend previous research from the query-focused and long document summarization domain. Experimental results show that our approach outperforms baselines, but we also conclude that simple extractive methods achieve competitive or better results. In addition, we examine the influence of different components and point out shortcomings related to summary length that make fair evaluation difficult. Depending on the dataset, we find that current reference summaries favor either extractive or abstractive techniques. This suggests that further efforts are needed to develop datasets and automatic evaluation metrics which take bias issues into account. Interesting future directions include the automatic evaluation of disaster reports, comprehensive human evaluation, and benchmarking of larger (proprietary) LLMs.

REFERENCES

Adams, G., Fabbri, A., Ladhak, F., Lehman, E., & Elhadad, N. (2023, December). From Sparse to Dense: GPT-4 Summarization with Chain of Density Prompting. In Y. Dong, W. Xiao, L. Wang, F. Liu, & G. Carenini (Eds.), *Proceedings of the 4th New Frontiers in Summarization Workshop* (pp. 68–74). Association for Computational Linguistics. https://doi.org/10.18653/v1/2023.newsum-1.7

Amati, G., Amodeo, G., Bianchi, M., Celi, A., Nicola, C., Flammini, M., Gaibisso, C., Gambosi, G., & Marcone, G. (2011). Fub, iasi-cnr, univaq at trec 2011 microblog track. *The Twentieth Text REtrieval Conference (TREC 2011) Proceedings*.

Amati, G., & Van Rijsbergen, C. J. (2002). Probabilistic models of information retrieval based on measuring the divergence from randomness. *ACM Transactions on Information Systems*, *20*(4), 357–389. https://doi.org/10.1145/582415.582416

Aslam, J., Diaz, F., Ekstrand-Abueg, M., McCreadie, R., Pavlu, V., & Sakai, T. (2015). Trec 2015 temporal summarization track overview. *The Twenty-Fourth Text REtrieval Conference, TREC 2015, November*.

Bae, S., Kim, T., Kim, J., & Lee, S.-g. (2019, November). Summary Level Training of Sentence Rewriting for Abstractive Summarization. In L. Wang, J. C. K. Cheung, G. Carenini, & F. Liu (Eds.), *Proceedings of the 2nd Workshop on New Frontiers in Summarization* (pp. 10–20). Association for Computational Linguistics. https://doi.org/10.18653/v1/D19-5402

Bajaj, A., Dangati, P., Krishna, K., Ashok Kumar, P., Uppaal, R., Windsor, B., Brenner, E., Dotterrer, D., Das, R., & McCallum, A. (2021, August). Long Document Summarization in a Low Resource Setting using Pretrained Language Models. In J. Kabbara, H. Lin, A. Paullada, & J. Vamvas (Eds.), *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing: Student Research Workshop* (pp. 71–80). Association for Computational Linguistics. https://doi.org/10.18653/v1/2021.acl-srw.7

Beltagy, I., Peters, M. E., & Cohan, A. (2020). Longformer: The Long-Document Transformer. *arXiv:2004.05150*.

Bertsch, A., Alon, U., Neubig, G., & Gormley, M. R. (2023). Unlimiformer: Long-range transformers with unlimited length input. *37th Conference on Neural Information Processing Systems*.

Cambrin, D. R., Cagliero, L., & Garza, P. (2024). Dqnc2s: Dqn-based cross-stream crisis event summarizer.

Chang, Y., Lo, K., Goyal, T., & Iyyer, M. (2023, October). BooookScore: A systematic exploration of book-length summarization in the era of LLMs.

Chen, Y.-C., & Bansal, M. (2018, July). Fast Abstractive Summarization with Reinforce-Selected Sentence Rewriting. In I. Gurevych & Y. Miyao (Eds.), *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)* (pp. 675–686). Association for Computational Linguistics. https://doi.org/10.18653/v1/P18-1063

Colvert, G., Silverberg, L., Darm, P., & Kasmanoff, N. (2023). Floodbrain: Flood disaster reporting by web-based retrieval augmented generation with an llm. *6th Workshop on Artificial Intelligence for Humanitarian Assistance and Disaster Response*.

Dasgupta, A., Kumar, R., & Ravi, S. (2013, August). Summarization Through Submodularity and Dispersion. In H. Schuetze, P. Fung, & M. Poesio (Eds.), *Proceedings of the 51st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)* (pp. 1014–1022). Association for Computational Linguistics.

Denis, L. S., Mietkiewicz, N., Short, K., Buckland, M., & Balch, J. (2020). ICS-209-PLUS - An all-hazards dataset mined from the US National Incident Management System 1999-2014. https://doi.org/10.6084/m9.figshare.8048252.v14

Ding, M., Zhou, C., Yang, H., & Tang, J. (2020). Cogltx: Applying bert to long texts. In H. Larochelle, M. Ranzato, R. Hadsell, M. Balcan, & H. Lin (Eds.), *Advances in neural information processing systems* (pp. 12792–12804, Vol. 33). Curran Associates, Inc.

Dong, Z., Tang, T., Li, L., & Zhao, W. X. (2023, February). A Survey on Long Text Modeling with Transformers.

Dusart, A., Pinel-Sauvagnat, K., & Hubert, G. (2021). ISSumSet: A tweet summarization dataset hidden in a TREC track. *Proceedings of the 36th Annual ACM Symposium on Applied Computing*, 665–671. https://doi.org/10.1145/3412841.3441946

Dusart, A., Pinel-Sauvagnat, K., & Hubert, G. (2023). TSSuBERT: How to Sum Up Multiple Years of Reading in a Few Tweets. *ACM Transactions on Information Systems*, *41*(4), 1–33. https://doi.org/10.1145/3581786

Egonmwan, E., Castelli, V., & Sultan, M. A. (2019, November). Cross-Task Knowledge Transfer for Query-Based Text Summarization. In A. Fisch, A. Talmor, R. Jia, M. Seo, E. Choi, & D. Chen (Eds.), *Proceedings of the 2nd Workshop on Machine Reading for Question Answering* (pp. 72–77). Association for Computational Linguistics. https://doi.org/10.18653/v1/D19-5810

Ernst, O., Caciularu, A., Shapira, O., Pasunuru, R., Bansal, M., Goldberger, J., & Dagan, I. (2022, July). Proposition-level clustering for multi-document summarization. In M. Carpuat, M.-C. de Marneffe, & I. V. Meza Ruiz (Eds.), *Proceedings of the 2022 conference of the north american chapter of the association for computational linguistics: Human language technologies* (pp. 1765–1779). Association for Computational Linguistics. https://doi.org/10.18653/v1/2022.naacl-main.128

Formal, T., Lassance, C., Piwowarski, B., & Clinchant, S. (2022). From distillation to hard negative sampling: Making sparse neural ir models more effective. *Proceedings of the 45th International ACM SIGIR Conference on Research and Development in Information Retrieval*, 2353–2359. https://doi.org/10.1145/3477495.3531857

Fox, E. A., & Shaw, J. A. (1994). Combination of multiple searches. *Proceedings of TREC-2*.

Garg, P. K., Chakraborty, R., & Dandapat, S. K. (2023). OntoDSumm: Ontology-Based Tweet Summarization for Disaster Events. *IEEE Transactions on Computational Social Systems*, 1–16. https://doi.org/10.1109/TCSS.2023.3266025

Garg, S., Vu, T., & Moschitti, A. (2020). Tanda: Transfer and adapt pre-trained transformer models for answer sentence selection. *Proceedings of the AAAI Conference on Artificial Intelligence*, *34*(05), 7780–7788. https://doi.org/10.1609/aaai.v34i05.6282

Gidiotis, A., & Tsoumakas, G. (2020). A Divide-and-Conquer Approach to the Summarization of Long Documents. *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, *28*, 3029–3040. https://doi.org/10.1109/TASLP.2020.3037401

Gillick, D., & Favre, B. (2009). A Scalable Global Model for Summarization. *Proceedings of the Workshop on Integer Linear Programming for Natural Language Processing*, 10–18.

Grail, Q., Perez, J., & Gaussier, E. (2021, April). Globalizing BERT-based Transformer Architectures for Long Document Summarization. In P. Merlo, J. Tiedemann, & R. Tsarfaty (Eds.), *Proceedings of the 16th Conference of the European Chapter of the Association for Computational Linguistics: Main Volume* (pp. 1792–1810). Association for Computational Linguistics. https://doi.org/10.18653/v1/2021.eacl-main.154

Guo, X., & Vosoughi, S. (2023, December). Length does matter: Summary length can bias summarization metrics. In H. Bouamor, J. Pino, & K. Bali (Eds.), *Proceedings of the 2023 conference on empirical methods in natural language processing* (pp. 15869–15879). Association for Computational Linguistics. https://doi.org/10.18653/v1/2023.emnlp-main.984

Imran, M., Castillo, C., Lucas, J., Meier, P., & Vieweg, S. (2014). Aidr: Artificial intelligence for disaster response. *Proceedings of the 23rd International Conference on World Wide Web*, 159–162. https://doi.org/10.1145/2567948.2577034

Jiang, A. Q., Sablayrolles, A., Mensch, A., Bamford, C., Chaplot, D. S., de las Casas, D., Bressand, F., Lengyel, G., Lample, G., Saulnier, L., Lavaud, L. R., Lachaux, M.-A., Stock, P., Scao, T. L., Lavril, T., Wang, T., Lacroix, T., & Sayed, W. E. (2023). Mistral 7b.

Kaufhold, M.-A. (2021). *Information Refinement Technologies for Crisis Informatics: User Expectations and Design Principles for Social Media and Mobile Apps*. Springer Fachmedien Wiesbaden. https://doi.org/10.1007/978-3-658-33341-6

Kedzie, C., McKeown, K., & Diaz, F. (2015, July). Predicting Salient Updates for Disaster Summarization. In C. Zong & M. Strube (Eds.), *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)* (pp. 1608–1617). Association for Computational Linguistics. https://doi.org/10.3115/v1/P15-1155

Kitaev, N., Kaiser, L., & Levskaya, A. (2020). Reformer: The Efficient Transformer. *International Conference on Learning Representations*.

Kryscinski, W., Rajani, N., Agarwal, D., Xiong, C., & Radev, D. (2022, December). BOOKSUM: A collection of datasets for long-form narrative summarization. In Y. Goldberg, Z. Kozareva, & Y. Zhang (Eds.), *Findings of the association for computational linguistics: Emnlp 2022* (pp. 6536–6558). Association for Computational Linguistics. https://doi.org/10.18653/v1/2022.findings-emnlp.488

Lappas, T., Crovella, M., & Terzi, E. (2012). Selecting a characteristic set of reviews. *Proceedings of the 18th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, 832–840. https://doi.org/10.1145/2339530.2339663

Laskar, M. T. R., Hoque, E., & Huang, J. X. (2020, December). WSL-DS: Weakly Supervised Learning with Distant Supervision for Query Focused Multi-Document Abstractive Summarization. In D. Scott, N. Bel, & C. Zong (Eds.), *Proceedings of the 28th International Conference on Computational Linguistics* (pp. 5647–5654). International Committee on Computational Linguistics. https://doi.org/10.18653/v1/2020.coling-main.495

Lewis, M., Liu, Y., Goyal, N., Ghazvininejad, M., Mohamed, A., Levy, O., Stoyanov, V., & Zettlemoyer, L. (2020, July). BART: Denoising sequence-to-sequence pre-training for natural language generation, translation, and comprehension. In D. Jurafsky, J. Chai, N. Schluter, & J. Tetreault (Eds.), *Proceedings of the 58th annual meeting of the association for computational linguistics* (pp. 7871–7880). Association for Computational Linguistics. https://doi.org/10.18653/v1/2020.acl-main.703

Litvak, M., & Vanetik, N. (2017, April). Query-based summarization using MDL principle. In G. Giannakopoulos, E. Lloret, J. M. Conroy, J. Steinberger, M. Litvak, P. Rankel, & B. Favre (Eds.), *Proceedings of the MultiLing 2017 Workshop on Summarization and Summary Evaluation Across Source Types and Genres* (pp. 22–31). Association for Computational Linguistics. https://doi.org/10.18653/v1/W17-1004

Macdonald, C., & Tonellotto, N. (2020). Declarative Experimentation in Information Retrieval Using PyTerrier. *Proceedings of the 2020 ACM SIGIR on International Conference on Theory of Information Retrieval*, 161–168. https://doi.org/10.1145/3409256.3409829

Mao, Z., Wu, C. H., Ni, A., Zhang, Y., Zhang, R., Yu, T., Deb, B., Zhu, C., Awadallah, A., & Radev, D. (2022, May). DYLE: Dynamic Latent Extraction for Abstractive Long-Input Summarization. In S. Muresan, P. Nakov, & A. Villavicencio (Eds.), *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)* (pp. 1687–1698). Association for Computational Linguistics. https://doi.org/10.18653/v1/2022.acl-long.118

McCreadie, R., & Buntain, C. L. (2023). CrisisFACTS: Buidling and Evaluating Crisis Timelines. *Proceedings of the 20th International ISCRAM Conference*, 320–339. https://doi.org/10.59297/JVQZ9405

Nguyen, T. H., & Rudra, K. (2022). Towards an Interpretable Approach to Classify and Summarize Crisis Events from Microblogs. *Proceedings of the ACM Web Conference 2022*, 3641–3650. https://doi.org/10.1145/3485447.3512259

Nogueira, R., Jiang, Z., Pradeep, R., & Lin, J. (2020, November). Document ranking with a pretrained sequence-to-sequence model. In T. Cohn, Y. He, & Y. Liu (Eds.), *Findings of the association for computational linguistics: Emnlp 2020* (pp. 708–718). Association for Computational Linguistics. https://doi.org/10.18653/v1/2020.findings-emnlp.63

Pereira, J., Fidalgo, R., Lotufo, R., & Nogueira, R. (2023). Crisis Event Social Media Summarization with GPT-3 and Neural Reranking. *Proceedings of the 20th International ISCRAM Conference*, 371–384. https://doi.org/10.59297/JJYT4136

Riedhammer, K., Favre, B., & Hakkani-Tür, D. (2010). Long story short – Global unsupervised models for keyphrase based meeting summarization. *Speech Communication*, *52*(10), 801–815. https://doi.org/10.1016/j.specom.2010.06.002

Robertson, S., & Zaragoza, H. (2009). The probabilistic relevance framework: Bm25 and beyond. *Found. Trends Inf. Retr.*, *3*(4), 333–389. https://doi.org/10.1561/1500000019

Rohde, T., Wu, X., & Liu, Y. (2021, September). Hierarchical Learning for Generation with Long Source Sequences.

Rudra, K., Ghosh, S., Ganguly, N., Goyal, P., & Ghosh, S. (2015). Extracting Situational Information from Microblogs during Disaster Events: A Classification-Summarization Approach. *Proceedings of the 24th ACM International on Conference on Information and Knowledge Management*, 583–592. https://doi.org/10.1145/2806416.2806485

Rudra, K., Goyal, P., Ganguly, N., Imran, M., & Mitra, P. (2019). Summarizing Situational Tweets in Crisis Scenarios: An Extractive-Abstractive Approach. *IEEE Transactions on Computational Social Systems*, *6*(5), 981–993. https://doi.org/10.1109/TCSS.2019.2937899

Rudra, K., Goyal, P., Ganguly, N., Mitra, P., & Imran, M. (2018). Identifying Sub-events and Summarizing Disaster-Related Information from Microblogs. *The 41st International ACM SIGIR Conference on Research & Development in Information Retrieval*, 265–274. https://doi.org/10.1145/3209978.3210030

Seeberger, P., & Riedhammer, K. (2022a). Combining deep neural reranking and unsupervised extraction for multi-query focused summarization. In I. Soboroff & A. Ellis (Eds.), *Proceedings of the thirty-first text retrieval conference, TREC 2022, online, november 15-19, 2022* (Vol. 500-338). National Institute of Standards; Technology (NIST).

Seeberger, P., & Riedhammer, K. (2022b). Enhancing crisis-related tweet classification with entity-masked language modeling and multi-task learning. *Proceedings of the Second Workshop on NLP for Positive Impact (NLP4PI)*, 70–78.

Seeberger, P., & Riedhammer, K. (2023). Multi-query focused disaster summarization via instruction-based prompting. In I. Soboroff (Ed.), *Proceedings of the thirty-second text retrieval conference, TREC 2023, gaithersburg, november 13-17, 2023*. National Institute of Standards; Technology (NIST).

Sequiera, R., Tan, L., & Lin, J. (2018). Overview of the trec 2018 real-time summarization track. *The Twenty-Seventh Text REtrieval Conference, TREC 2018, November*.

Su, D., Xu, Y., Yu, T., Siddique, F. B., Barezi, E., & Fung, P. (2020, December). CAiRE-COVID: A Question Answering and Query-focused Multi-Document Summarization System for COVID-19 Scholarly Information Management. In K. Verspoor, K. B. Cohen, M. Conway, B. de Bruijn, M. Dredze, R. Mihalcea, & B. Wallace (Eds.), *Proceedings of the 1st Workshop on NLP for COVID-19 (Part 2) at EMNLP 2020*. Association for Computational Linguistics. https://doi.org/10.18653/v1/2020.nlpcovid19-2.14

Su, D., Yu, T., & Fung, P. (2021, August). Improve Query Focused Abstractive Summarization by Incorporating Answer Relevance. In C. Zong, F. Xia, W. Li, & R. Navigli (Eds.), *Findings of the Association for Computational Linguistics: ACL-IJCNLP 2021* (pp. 3124–3131). Association for Computational Linguistics. https://doi.org/10.18653/v1/2021.findings-acl.275

Sun, S., Shapira, O., Dagan, I., & Nenkova, A. (2019, June). How to compare summarizers without target length? pitfalls, solutions and re-examination of the neural summarization literature. In A. Bosselut, A. Celikyilmaz, M. Ghazvininejad, S. Iyer, U. Khandelwal, H. Rashkin, & T. Wolf (Eds.), *Proceedings of the workshop on methods for optimizing and evaluating neural language generation* (pp. 21–29). Association for Computational Linguistics. https://doi.org/10.18653/v1/W19-2303

Touvron, H., Lavril, T., Izacard, G., Martinet, X., Lachaux, M.-A., Lacroix, T., Rozière, B., Goyal, N., Hambro, E., Azhar, F., Rodriguez, A., Joulin, A., Grave, E., & Lample, G. (2023). Llama: Open and efficient foundation language models.

Vieweg, S., Hughes, A. L., Starbird, K., & Palen, L. (2010). Microblogging during two natural hazards events: What twitter may contribute to situational awareness. *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, 1079–1088. https://doi.org/10.1145/1753326.1753486

Vig, J., Fabbri, A., Kryscinski, W., Wu, C.-S., & Liu, W. (2022, July). Exploring Neural Models for Query-Focused Summarization. In M. Carpuat, M.-C. de Marneffe, & I. V. Meza Ruiz (Eds.), *Findings of the Association for Computational Linguistics: NAACL 2022* (pp. 1455–1468). Association for Computational Linguistics. https://doi.org/10.18653/v1/2022.findings-naacl.109

Wan, X. (2008). Using only cross-document relationships for both generic and topic-focused multi-document summarizations. *Information Retrieval*, *11*(1), 25–49. https://doi.org/10.1007/s10791-007-9037-5

Wan, X., Yang, J., & Xiao, J. (2007). Manifold-ranking based topic-focused multi-document summarization. *Proceedings of the 20th International Joint Conference on Artifical Intelligence*, 2903–2908.

Wang, Y., Zhang, Z., & Wang, R. (2023, July). Element-aware summarization with large language models: Expert-aligned evaluation and chain-of-thought method. In A. Rogers, J. Boyd-Graber, & N. Okazaki (Eds.), *Proceedings of the 61st annual meeting of the association for computational linguistics (volume 1: Long papers)* (pp. 8640–8665). Association for Computational Linguistics. https://doi.org/10.18653/v1/2023.acl-long.482

Wolf, T., Debut, L., Sanh, V., Chaumond, J., Delangue, C., Moi, A., Cistac, P., Rault, T., Louf, R., Funtowicz, M., Davison, J., Shleifer, S., von Platen, P., Ma, C., Jernite, Y., Plu, J., Xu, C., Le Scao, T., Gugger, S., . . . Rush, A. (2020). Transformers: State-of-the-art natural language processing. *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing: System Demonstrations*, 38–45. https://doi.org/10.18653/v1/2020.emnlp-demos.6

Wu, J., Ouyang, L., Ziegler, D. M., Stiennon, N., Lowe, R., Leike, J., & Christiano, P. (2021, September). Recursively Summarizing Books with Human Feedback.

Xu, J., & Durrett, G. (2019, November). Neural Extractive Text Summarization with Syntactic Compression. In K. Inui, J. Jiang, V. Ng, & X. Wan (Eds.), *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)* (pp. 3292–3303). Association for Computational Linguistics. https://doi.org/10.18653/v1/D19-1324

Xu, Y., & Lapata, M. (2020, November). Coarse-to-Fine Query Focused Multi-Document Summarization. In B. Webber, T. Cohn, Y. He, & Y. Liu (Eds.), *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)* (pp. 3632–3645). Association for Computational Linguistics. https://doi.org/10.18653/v1/2020.emnlp-main.296

Xu, Y., & Lapata, M. (2021, August). Generating Query Focused Summaries from Query-Free Resources. In C. Zong, F. Xia, W. Li, & R. Navigli (Eds.), *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)* (pp. 6096–6109). Association for Computational Linguistics. https://doi.org/10.18653/v1/2021.acl-long.475

Xu, Y., & Lapata, M. (2022). Document Summarization with Latent Queries. *Transactions of the Association for Computational Linguistics*, *10*, 623–638. https://doi.org/10.1162/tacl_a_00480

Zaheer, M., Guruganesh, G., Dubey, K. A., Ainslie, J., Alberti, C., Ontanon, S., Pham, P., Ravula, A., Wang, Q., Yang, L., & Ahmed, A. (2020). Big Bird: Transformers for Longer Sequences. In H. Larochelle, M. Ranzato, R. Hadsell, M. F. Balcan, & H. Lin (Eds.), *Advances in Neural Information Processing Systems* (pp. 17283–17297, Vol. 33). Curran Associates, Inc.

Zhang, T., Kishore, V., Wu, F., Weinberger, K. Q., & Artzi, Y. (2020). Bertscore: Evaluating text generation with bert. *International Conference on Learning Representations*.

Zhang, Y., Ni, A., Mao, Z., Wu, C. H., Zhu, C., Deb, B., Awadallah, A., Radev, D., & Zhang, R. (2022, May). Summˆ N: A Multi-Stage Summarization Framework for Long Input Dialogues and Documents. In S. Muresan, P. Nakov, & A. Villavicencio (Eds.), *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)* (pp. 1592–1604). Association for Computational Linguistics. https://doi.org/10.18653/v1/2022.acl-long.112

Zhang, Y., Ni, A., Yu, T., Zhang, R., Zhu, C., Deb, B., Celikyilmaz, A., Awadallah, A. H., & Radev, D. (2021, November). An Exploratory Study on Long Dialogue Summarization: What Works and What's Next. In M.-F. Moens, X. Huang, L. Specia, & S. W.-t. Yih (Eds.), *Findings of the Association for Computational Linguistics: EMNLP 2021* (pp. 4426–4433). Association for Computational Linguistics. https://doi.org/10.18653/v1/2021.findings-emnlp.377

Zhu, C., Xu, R., Zeng, M., & Huang, X. (2020, November). A Hierarchical Network for Abstractive Meeting Summarization with Cross-Domain Pretraining. In T. Cohn, Y. He, & Y. Liu (Eds.), *Findings of the Association for Computational Linguistics: EMNLP 2020* (pp. 194–203). Association for Computational Linguistics. https://doi.org/10.18653/v1/2020.findings-emnlp.19

DATASET DETAILS

Table 4. Details about the CrisisFACTS datasets. Events with the ID's 001 to 008 and 009 to 018 correspond to CrisisFACTS 2022 and CrisisFACTS 2023, respectively. The columns Tweets, Reddit, News, and Facebook represent the number of available stream items.

ID	Event	Event Type	Days	Tweets	Reddit	News	Facebook
001	Lilac Wildfire 2017	Wildfire	9	41,346	1,738	2,494	5,437
002	Cranston Wildfire 2018	Wildfire	6	22,974	231	1,967	5,386
003	Holy Wildfire 2018	Wildfire	7	23,528	459	1,495	7,016
004	Hurricane Florence 2018	Hurricane	15	41,187	120,776	18,323	196,281
005	Maryland Flood 2018	Flood	4	33,584	2,006	2,008	4,148
006	Saddleridge Wildfire 2019	Wildfire	4	31,969	244	2,267	3,869
007	Hurricane Laura 2020	Hurricane	2	36,120	10,035	6,406	9,048
008	Hurricane Sally 2020	Hurricane	8	40,695	11,825	15,112	48,492
009	Beirut Explosion 2020	Accident	7	94,892	3,257	1,163	368,866
010	Houston Explosion 2020	Accident	7	58,370	5,704	2,175	6,281
011	Rutherford TN Floods 2020	Floods	5	11,019	475	268	9,116
012	TN Derecho 2020	Storm/Flood	7	49,247	1,496	15,425	13,521
013	Edenville Dam Fail 2020	Accident	7	16,527	2,339	961	8,358
014	Hurricane Dorian 2019	Hurricane	7	86,915	91,173	7,507	370,644
015	Kincade Wildfire 2019	Wildfire	7	91,548	10,174	339	35,011
016	Easter Tornado Outbreak 2020	Tornadoes	5	91,812	5,070	750	34,343
017	Tornado Outbreak April 2020	Tornadoes	6	99,575	1,233	217	19,878
018	Tornado Outbreak March 2020	Tornadoes	6	95,221	16,911	641	87,242

PROMPT DETAILS

We provide an overview of the prompts used in our work. For the LLMNug-based methods, we rely on one demonstration example as we otherwise noticed format errors that made it difficult to parse the facts. All other methods use zero-shot prompting without the use of any demonstration examples. We list all LLM-based methods and show the prompt dependencies in the corresponding brackets: Sum (Table 5), Select+Sum (Table 5), SumCoT (Table 6, Table 7), Incremental (Table 5, Table 8), Hierarchical (Table 5, Table 9), LLMNug-Concat (Table 10), and LLMNug-ClusterFuse (Table 10, Table 5).

Table 5. Generate initial summary

We are creating one detailed summary for disaster response officers. Summarize and rewrite the following documents into coherent and readable paragraphs. Do not deviate from the facts of these documents or add any new information. Include as many details as possible. The summary could include multiple paragraphs.

Documents: {documents}

Summary:

Table 6. Element extraction

Documents: {documents}

What are the important entities in this documents?
What events are happening in this documents?
What is the result of these events?

Please answer the above questions:

Table 7. Generate element aware summary

We are creating one detailed summary for disaster response officers. Summarize and rewrite the following documents into coherent and readable paragraphs. Do not deviate from the facts of these documents or add any new information. Include as many details as possible. The summary could include multiple paragraphs. Documents: {documents} {elements} Let's integrate the above information and summarize the documents:

Table 8. Iterative refinement

We are creating one detailed summary for disaster response officers and provide an existing summary up to a certain point: {previous_summary} New documents: {documents} Given the new documents, incorporate any new vital information into a new coherent and readable summary. Do not deviate from the facts of these documents or add any new information. Include as many details as possible. The summary could include multiple paragraphs. Summary:

Table 9. Hierarchical merging

We are creating one detailed summary for disaster response officers. Distill and rewrite the following previous summaries into consolidated paragraphs. Do not deviate from the facts of these summaries or add any new information. Include as many details as possible. The final summary could include multiple paragraphs. Previous summaries: {summaries} Summary:

Table 10. Fact extraction

You are a fact extractor for disaster response organizations. Use the documents to answer the question based on a list of extracted facts as evidence. Please follow the instructions for the facts: 1. The facts must be short. 2. The format of one fact is text-snippet (source document). 3. Provide the source documents for each fact with the format: (Doc-1, Doc-2, ..) 4. Include fact-relevant entities such as locations, numbers, dates, etc. 5. Only include facts which are focused on the question. 6. The list items must start with * bullet points. Do not use numberings. We provide you one example within " marks: '{demonstration}' Your task: Documents: {documents} Question: {query} Facts list:

EXAMPLES

Table 11. Summary of LLMNug-Concat w/ QILP with L=16 for request CrisisFACTS-015-r4

- Pharmacies, supermarkets, and restaurants are closed during blackouts in bedroom communities of San Francisco and Silicon Valley; Town Hall is closed except for essential services - - The main evacuation order encompasses a huge area of Sonoma County, including Santa Rosa - - Strong winds are forecasted to impact the Kincade Fire, Santa Rosa and much of the North Bay .; WILDFIRE SMOKE is impacting the San Francisco Bay Area and Los Angeles Area . - - The Kincade Fire has burned through 50,000 acres (20,200 hectares) of land; The Kincade fire spread across more than 75,000 acres (303 square kilometers) - - 4,548 firefighters are assigned to the fires; 4,500 firefighters have been working on containment lines and structure protection - - The Kincade Fire has burned 75,415 acres and is 15% contained - - The Wine & Food Affair event in Sonoma County may have been canceled; Classes at Chalon and Doheny have been canceled - - La Tortilla Factory's main office and production center is closed due to power outages and evacuations; Think Tank Photo offices are closed due to wildfires and evacuations - - At least eight homes were destroyed and six damaged in the LA-area blaze; Three people were killed and over 1,600 homes and other buildings were destroyed from north of Los Angeles to Malibu - - Four shelters for Kincade Fire evacuees are at capacity, while five are still accepting evacuees; Jaxon Keys' Winery & Distillery is an open shelter for large animals affected by the Kincade Fire - - A High Wind Warning is also in effect during the same period, with northeast winds 30 to 50 mph and gusts up to 60 to 70 mph, and isolated gusts up to 80 mph at higher elevations - - Volunteers are needed for Petaluma evacuation centers; Volunteers are needed for food service and general labor; Farm animals need volunteers to take them in - - Nearly 200,000 people have been evacuated from their homes throughout California; Over 100,000 structures are threatened - - Several state parks have closed due to the Kincade Fire and Public Safety Power Shutoffs; The registrar's office in Santa Rosa is closed because of Kincade Fire evacuations - - Red Flag No Parking restrictions are in effect citywide in Los Angeles due to the extreme fire danger - - An evacuation warning was announced for parts of Lake County; The entire town of Geyserville and the entire city of [redacted] are under mandatory evacuation orders -

Table 12. Summary of LLMNug-ClusterFuse w/ QILP with L=16 for request CrisisFACTS-015-r4

- The affected areas are located south of Mulholland, west of the 405 freeway, north of Sunset Boulevard, and west of Temescal Canyon. Entrance to Brentwood above Sunset Boulevard has been closed. - - Evacuation orders and warnings continue to apply to various cities and communities within the affected region. - - The towns of Geyserville and a redacted city, as well as certain communities and cities in the affected areas, are under mandatory evacuation orders due to ongoing wind events. - - Among them, around 4,500 firefighters have been actively engaged in setting up containment lines and safeguarding structures to prevent further damage. - - These evacuations have led to road closures on the 405 freeway, which is the nation's busiest highway. - - Nearly 1 million people are without power, and over 200,000 have been displaced from their homes. The Memorial bridge near Kennedy Ln was not impacted by the fires. - - Evacuations can pose challenges, especially for vulnerable communities that depend on childcare. Emergency response teams are encountering difficulties accessing some regions due to road closures. - - An evacuation warning has been announced for various areas in Lake County, including the northern part of Dry Creek Valley and Middletown, as well as the Twin Pines Casino area (Zone 4). - - Zone 4B, which includes areas south of Westside Road to Millcreek Road, has specific evacuation orders in effect. Some individuals have been unable to take their animals with them during evacuations. - - The registrar's office in Santa Rosa has closed due to evacuations. Recovery efforts are underway, with at least 124 structures confirmed destroyed in Sonoma County. - - Over 185,000 people have been evacuated from their homes in Sonoma County alone due to the Kincade Fire. Nearly 200,000 people have been displaced by the wildfires across California. - - Overflowing conditions have been reported in both emergency shelters and hotels, creating a critical need for additional assistance. - - Some evacuation orders have been lifted in other areas. Individuals in high-risk zones should be prepared to evacuate with short notice. - - Four of these shelters have reached full capacity, but fifteen continue to accept evacuees. Additionally, the Westwood Recreation Center serves as a shelter for those displaced by the Getty Fire. - - Sonoma County Fairgrounds and Napa Valley Expo are accommodating evacuees and small animals. Animal shelters are working diligently to provide care for pets displaced by the fires. - - Specifically mentioned are the need for volunteers at Petaluma evacuation centers, Sonoma County Animal Response Team (CART), Integrative Healers Action Network, and Kincade Fire Clinics. -

Table 13. Summary of QILP with L=16 for request CrisisFACTS-015-r4

- Some 180,000 people have been ordered to leave homes, with roads around Santa Rosa north of San Francisco packed with cars as people tried to flee. - - Evacuations have been lifted in a number of communities, and the Napa County emergency evacuation shelter at Napa Valley College closed today at noon. - - Over 200,000 people have been ordered to evacuate and more than 100,000 structures are threatened. . - - About 156,000 people were under mandatory evacuation orders. - - The following DOR offices remain closed due to the PSPS event: Novato, Ukiah and Lakeport. - - Nearly 200,000 people evacuated their homes in California this weekend. - - About 156,000 people were still under evacuation orders. - - Another million people have been told they could lose supplies. - - Evacuation Orders effecting 189,980 people. - - CWS Kits will be made available if requested, and we are checking in with our long-term recovery group partners in Los Angeles as to other ways that we can support them. - - The following road closures are in effect: Westside Road at Highway 101 offramp – no eastbound traffic on Mill Street Dry Creek Road at Highway 101 – no eastbound traffic on Dry Creek Road Chiquita Ro - - At least 10,000 people have been evacuated, gardeners and cleaners could be seen on their way to work. - - ”There now are 4,548 people, mainly firefighters, assigned to the fire and more were expected later in the day.” - - Areas North of Westside Road to Millcreek Road within Zone 4 have been reduced to an EVACUATION WARNING and are open for repopulation. - - Mandatory evacuations are in place for parts of the area. - - #KincadeFire VOLUNTEERS NEEDED #Petaluma People Services is coordinating volunteers for the Petaluma evacuation centers. -