Facts Summarization at the TREC 2023: IIT(BHU) in CrisisFACTs Track

Amit Yadav and Sukomal Pal

Information Retrieval (IR) Lab - Department of Computer Science
Indian Institute of Technology(BHU) Varanasi Uttar Pradesh, India
221005

Abstract

The CrisisFACTS Track tackles the challenges of gathering crucial facts from diverse disaster-related events through multi-stream fact-finding. This paper presents our innovative method for summarizing crisis events in the TREC 2023 CrisisFACTS track. Our approach involves a two-step summarization process utilizing retrieval and ranking techniques. Initially, a sparse retrieval framework treats content from various online streams as a document corpus. It uses term matching to retrieve relevant contents, termed "facts", based on specific event day queries. Subsequently, pre-trained models assess the semantic similarity between query-fact and fact-fact pairs. These similarities are used to score and rank the facts, forming the basis for extracting daily event summaries. Relevant data are first retrieved using the IR technique from pyTerrier and then re-ranked. Top-k (k=32) posts are finally used to create summaries. Our model is not able to create good summaries for the event on a specific day. But We are confident that this approach holds potential for yielding promising results with "BM25 + DFReeKLIM" model, especially for labels with limited resources.

Keywords: Crisis-events, Facts Summarization, Indexing, Retrieval.

1 Introduction

In recent years, summarization has become an increasingly significant area of study. Especially during crises, generating concise, informative, and non-redundant summaries of events is crucial. These summaries are vital in keeping local communities and stakeholders promptly informed, enabling them to respond effectively. While numerous summarization approaches have been proposed in recent works, most assume that all input texts for summarization models are relevant to the event.

Some methods filter out irrelevant content using classification techniques, necessitating labelled data for training the classifiers. Moreover, previous studies have primarily focused on specific characteristics of individual data sources like Twitter or news articles rather than considering multi-stream data for the summarization process.

The inaugural CrisisFACTS[1] Track at the 2023 Text Retrieval Conference (TREC-2023)[2] is focused on delivering timely and pertinent answers to questions that are crucial for crisis response managers' daily reports. The objective is to present these answers in a single ranked list every day, with a strong emphasis on covering all available facts related to the crisis for that specific day. The system design discussed in this paper considers three essential factors: fact coverage, relevance, and recency; each plays a significant role in shaping the system.

2 Related Work

Dusart et al.[1] outlines two primary methodologies employing neural language models (NLMs). In the initial approach, a pre-trained NLM is utilized to encode individual items (text snippets) along with their token frequencies. Subsequently, a greedy algorithm is employed to select the most significant and non-repetitive items. The second approach involves using a pre-trained NLM to encode each item and then determining the stream's representation as the average of all item representations.

Nguyen and Rudra[2] introduces a two-part method for multi-stream summarization, employing sparse retrieval and semantic ranking techniques. In the retrieval phase, a term matching approach is utilized. The ranking phase employs various pre-trained language models to assess semantic similarity between pairs of query-fact (based on query) or fact-fact (based on graph), enabling the scoring and ranking of facts.

Pereira et al.[3] presents a two-step technique for summarizing crisis events from multiple data streams, employing GPT-3 and monoT5 models. Initially, relevant documents are retrieved utilizing BM25 and query expansion methods. Subsequently, summaries are generated through one-shot query-based multi-document summarization and chain-of-thought reasoning. The method is assessed on the TREC CrisisFACTS dataset, demonstrating strong performance in both automated and manual evaluations, albeit with notable redundancy. Moreover, it is characterized as a few-shot method, requiring no annotated data and facilitating swift application.

Saroj and Pal[5] examines existing research concerning the utilization of social media during disaster scenarios, with a particular emphasis on the technical aspects of extracting, processing, and disseminating information. It delves into three primary areas: the impact of emergencies on social media, techniques for gathering and processing information from social platforms, and the influence of social media on governmental and non-governmental policies and actions. Additionally, it outlines the hurdles and advantages associated with utilizing social media data for disaster response and management while proposing potential avenues for future research.

3 Approach

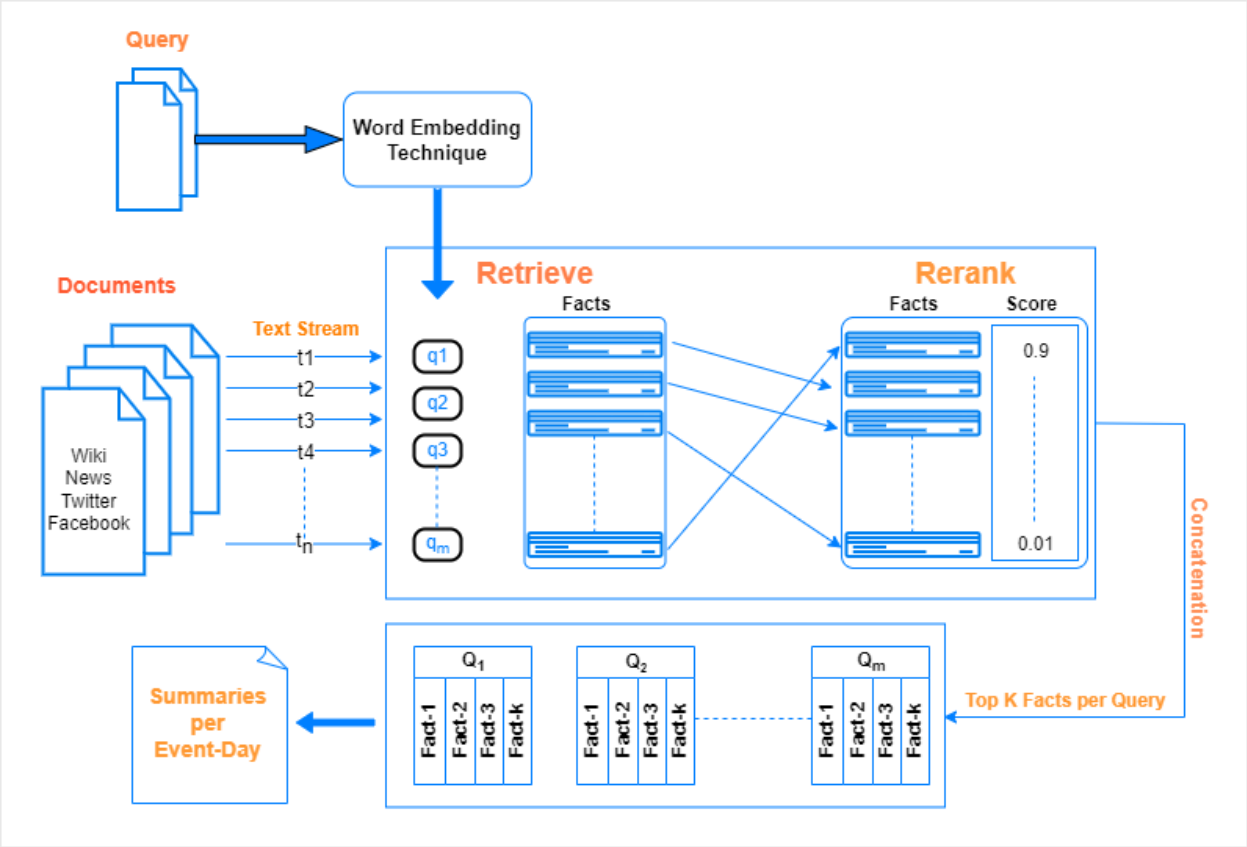
In this section, we outline our proposed method for extracting essential information from social media and web news to aid in crisis event management. The method comprises two main steps: retrieval and re-ranking. This approach falls under query-based multi-document summarization,

[1] https://crisisfacts.github.io/

[2] https://trec.nist.gov/

which condenses information from multiple documents based on user queries. The method assumes the existence of questions representing the user's specific information requirements. The following subsections provide a detailed explanation of the process involved. Given a collection of content text $T = \{t_1, t_2, .., t_n\}$ from various sources such as Twitter, Reddit, News, and Facebook, and a set of queries $Q = \{q_1, q_2, .., q_m\}$ outlining stakeholders' information needs for a particular event on a specific day, our objective is to provide a list of up to K most relevant content texts, referred to as "facts", along with their respective importance scores, forming our daily system summary.

Figure 1: Proposed Framework



3.1 Dataset

We are using a dataset from CrisisFACTS [3] and access to the track's datasets (i.e., data streams and queries) of 2022 TREC. We download datasets according to event-wise and create a single dataset file. This dataset has 5815363 entries and six columns:
"doc_id", "event", "text", "source", "source_type", "unix_timestamp".

[3] https://github.com/crisisfacts

doc_id	5815358
event	CrisisFACTS-001
text	Live updates: San Diego County fire is 92 percent contained
source	https://en.wikipedia.org/wiki/Egli_Trimi
source_type	News
unix_timestamp	1512604800.0

We used this data frame to retrieve the relevant data based on every query.

3.2 Retrieve and Re-rank

The first step, candidate document retrieval, focuses on identifying relevant documents based on user queries. This is crucial to reduce the computational workload in subsequent re-ranking stages. Various search functions can be used for this purpose. In our study, we utilize the BM25[3] and DFReeKLIM[4] algorithms.

We used BM25 and DFReeKLIM for their effectiveness in identifying and ranking relevant content. BM25 is computationally efficient, making it suitable for processing large volumes of text data commonly encountered in summarization tasks. DFReeKLIM assesses the divergence between the distribution of terms in the source document and the distribution of terms in the summary.

We used the PyTerrier[4] , an information retrieval framework, to extract and rank the retrieved documents for each query search. First, we are expanding the length of query vector by word embedding as mentioned in Figure: 1. Then, for each query, we are extracting the Facts data up to 400 in length for each Fact. We are creating a BatchRetriever (a pyTerrier retriever algorithm) for retrieving the metadata as [‘doc_id’, ‘text’].

The re-ranking stage prioritizes candidate documents’ relevance to the user’s queries. It takes candidate documents from diverse sources, such as web news and social media, as input and produces a ranked list of documents considered most relevant to the specific query. We used cosine similarity between document and query to rank the multiple document for a set of queries. We use TF-IDF algorithm to find the similarity between documents and queries.

In this study, we retrieve the documents based on each query for the specific event and day. For each event and each day, we are retrieving the top facts after re-ranking the retrieved document for each query. We used ‘text’ from each query request to retrieve the relevant data from the upcoming data stream of different data sources for a particular event day.

Top-K (K=32) facts based on ranking are then used to create the summaries according to each query for an event for a specific day. These created summaries are Extracted Summaries which are used to help stack-holders to information needs for a particular event on a specific day.

[4] https://pyterrier.readthedocs.io/en/latest/

4 Evaluation Metrics

- **Redundancy and Comprehensiveness**

  We are able to create a bipartite graph linking specific facts from runs to meta-facts from CrisisFACTS. Each meta-fact is assigned a score based on its assessor label, or zero if no label is provided indicating irrelevance. When examining the facts generated by a particular run, we determine the corresponding set of connected meta-facts.
  The **redundancy**[2] score quantifies the level of duplicate or overlapping information within a collection of extracted facts or summaries. A higher redundancy score signifies a greater degree of repetition or duplication, potentially diminishing the utility and effectiveness of the extracted content.
  The **comprehensiveness**[2] score evaluates the extent to which a set of extracted facts or summaries encompasses the full spectrum of pertinent information within a specific topic or dataset. A higher comprehensiveness score indicates that the extracted content is more exhaustive and inclusive, capturing all pertinent aspects or details of the subject or dataset. From this set of adjacent meta-facts, we compute redundancy and comprehensiveness as follows:

$$Redundancy = \frac{\sum \text{score of adjacent meta-facts}}{|\text{ all adjacent meta-facts }|} \tag{1}$$

$$Comprehensiveness = \frac{\sum \text{score of adjacent meta-facts}}{|\text{ all meta-facts with non-zero score }|} \tag{2}$$

  For assigning scores to meta-facts, we use the following mapping, though many such mappings are possible:

Label	Score
USEFUL FACT	1.0
REDUNDANT FACT	0.5
POOR FACT	0.0
LAGGED FACT	0.0

  - **Useful Fact:**Text segments labeled as such denote information deemed valuable for completing the ICS209 Incident Status Summary form effectively.
  - **Poor Fact:** Text segments labeled as such may hold potential utility but are marred by confusion, lack essential details, are poorly articulated, or are otherwise challenging to interpret.
  - **Redundant Fact:** Text segments labeled as such denote information that has already been presented earlier in the current summary.
  - **Lagged Fact:** Text segments labeled as such represent information that has been previously covered in a summary from a prior day.

  Figure 2 provides the average redundancy and comprehensiveness scores for each event-day pair:

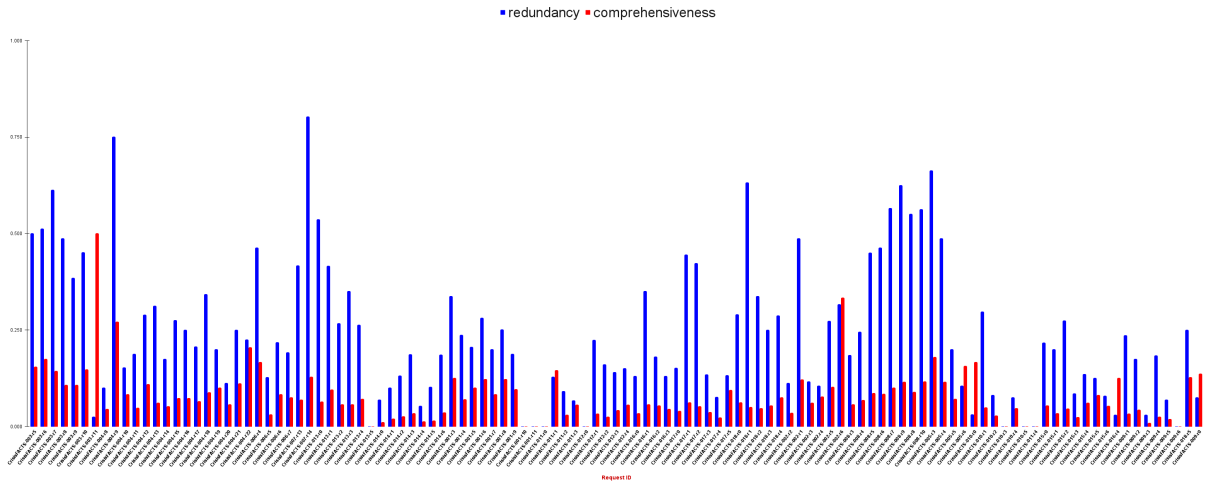


Figure 2: Redundancy and Comprehensiveness

- **BERTScore –** The BERTScore metric measures the similarity between a reference text and a generated text by comparing their embeddings using contextualized representations obtained from the BERT model. Unlike traditional evaluation metrics such as BLEU or ROUGE, which rely on exact matching of words or n-grams, BERTScore takes into account the semantic similarity and context of words in the text.

  Given a reference sentence

  $$x = \langle x_1, x_2, ..., x_k \rangle \tag{3}$$

  and a generated sentence

  $$\widehat{x} = \langle \widehat{x}_1, \widehat{x}_2, ..., \widehat{x}_k \rangle \tag{4}$$

  BERTScore compares sentences by aligning them to have the same length. This is done to ensure a fair evaluation where each token in the generated sentence is compared to a corresponding token in the reference sentence. Having sentences of similar lengths is crucial for meaningful comparison, as it prevents longer sentences from being unfairly penalized for potentially containing more information or context. By aligning sentence lengths, BERTScore facilitates a more accurate assessment of semantic similarity and quality, leveraging contextual embedding from BERT to capture nuanced meanings beyond surface-level text.. we use contextual embedding to represent the tokens, and compute matching using cosine similarity, optionally weighted with inverse document frequency scores.

  $$R_{BERT} = \frac{1}{|x|} \sum_{x_i \epsilon x} \underset{\widehat{x}_j \epsilon \widehat{x}}{max} x_i^T \widehat{x}_j \tag{5}$$

  $$P_{BERT} = \frac{1}{|\widehat{x}|} \sum_{\widehat{x}_j \epsilon \widehat{x}} \underset{x_i \epsilon x}{max} x_i^T \widehat{x}_j \tag{6}$$

  $$F_{BERT} = 2 \frac{P_{BERT} \,.\, R_{BERT}}{P_{BERT} \,+\, R_{BERT}} \tag{7}$$

- **ROUGE Score –** The ROUGE-2 score serves as a key metric for assessing the performance of text summarization systems. Derived from the acronym "Recall-Oriented Understudy for Gisting Evaluation". ROUGE-2 specifically evaluates the extent of overlap between bigrams, which are sequences of two adjacent words, within both the generated summary and the reference summary. This metric offers insight into how effectively the generated summary encapsulates the content and structure of the reference summary.

$$R_{ROUGE-2} = \frac{\text{Overlapping number of 2-grams}}{\text{Number of 2-grams in the reference sentence}} \tag{8}$$

$$P_{ROUGE-2} = \frac{\text{Overlapping number of 2-grams}}{\text{Number of 2-grams in the generated sentence}} \tag{9}$$

$$F_{ROUGE-2} = 2\,\frac{P_{ROUGE-2}\,.\,R_{ROUGE-2}}{P_{ROUGE-2} + R_{ROUGE-2}} \tag{10}$$

5 Results

Results are shown below (Table 1, 2) for different metrics like Rouge-2 and BERT for each event. Table 1 & 2 shows the BERT and ROUGE mean-score of the top two submissions respectively. These results are the mean of results generated from the model - **DFReeKLIM + BM25** and **BM25**.

To generate the BERTScore, we use model "microsoft/deberta-xlarge-mnli". This model is an iteration of the DeBERTa-XLARGE model tailored for the MultiNLI task. MultiNLI (Multi-Genre Natural Language Inference) is a large-scale dataset for natural language inference tasks, where the goal is to determine the relationship between two given sentences.

Table 1: Mean ROUGE-2 Score by Submission

Event	nist.f1	nist.precision	nist.recall	wiki.f1	wiki.precision	wiki.recall
CrisisFACTS-009	0.270	0.302	0.245	0.024	0.013	0.282
CrisisFACTS-010	0.147	0.122	0.201	0.015	0.008	0.199
CrisisFACTS-011	0.170	0.126	0.334	-	-	-
CrisisFACTS-012	0.176	0.204	0.155	-	-	-
CrisisFACTS-013	0.325	0.400	0.275	0.014	0.007	0.286
CrisisFACTS-014	0.244	0.427	0.171	0.046	0.027	0.246
CrisisFACTS-015	0.310	0.332	0.291	0.015	0.008	0.368
CrisisFACTS-016	0.195	0.253	0.163	0.026	0.014	0.183
CrisisFACTS-017	0.249	0.407	0.179	0.016	0.008	0.272
CrisisFACTS-018	0.228	0.445	0.155	0.014	0.007	0.244

Table 2: Mean BERT Score by Submission

Event	nist.f1	nist.precision	nist.recall	wiki.f1	wiki.precision	wiki.recall
CrisisFACTS-009	0.586	0.578	0.593	0.509	0.483	0.537
CrisisFACTS-010	0.524	0.514	0.536	0.476	0.440	0.518
CrisisFACTS-011	0.558	0.557	0.559	-	-	-
CrisisFACTS-012	0.503	0.505	0.501	-	-	-
CrisisFACTS-013	0.582	0.590	0.577	0.514	0.469	0.570
CrisisFACTS-014	0.551	0.547	0.554	0.479	0.512	0.483
CrisisFACTS-015	0.590	0.587	0.593	0.527	0.486	0.574
CrisisFACTS-016	0.556	0.555	0.557	0.485	0.471	0.501
CrisisFACTS-017	0.551	0.531	0.576	0.501	0.446	0.572
CrisisFACTS-018	0.591	0.589	0.593	0.499	0.450	0.561

Table 3 describes the Rouge-2 mean-scores of the top two participants of CrisisFACTs TREC 2023. These results are provided by organizers.

Table 3: Mean ROUGE-2 Score By TREC 2023

Event	nist.f1	nist.precision	nist.recall	wiki.f1	wiki.precision	wiki.recall
CrisisFACTS-009	0.213	0.239	0.245	0.033	0.019	0.242
CrisisFACTS-010	0.149	0.114	0.286	0.023	0.012	0.189
CrisisFACTS-011	0.145	0.100	0.359	-	-	-
CrisisFACTS-012	0.179	0.179	0.210	-	-	-
CrisisFACTS-013	0.229	0.224	0.286	0.019	0.010	0.243
CrisisFACTS-014	0.234	0.295	0.220	0.0536	0.032	0.221
CrisisFACTS-015	0.233	0.233	0.287	0.020	0.010	0.328
CrisisFACTS-016	0.182	0.182	0.208	0.034	0.019	0.178
CrisisFACTS-017	0.224	0.266	0.219	0.018	0.009	0.233
CrisisFACTS-018	0.228	0.291	0.211	0.020	0.010	0.224

Table 4: Rouge-2 Score by Submission

Run	nist.f1	nist.precision	nist.recall	wiki.f1	wiki.precision	wiki.recall
DFReeKLIM	0.231	0.298	0.217	0.016	0.008	0.208
BM25	0.180	0.284	0.158	0.017	0.009	0.163
DFReeKLIM + BM25	**0.232**	**0.302**	**0.217**	0.017	0.009	**0.208**
TF-IDF	0.148	0.276	0.116	**0.022**	**0.012**	0.166

Table 5: BERT Score by Submission

Run	nist.f1	nist.precision	nist.recall	wiki.f1	wiki.precision	wiki.recall
DFReeKLIM	0.558	0.544	**0.562**	0.394	0.369	**0.428**
BM25	0.541	0.534	0.549	0.393	0.370	0.422
DFReeKLIM + BM25	**0.559**	**0.554**	0.555	**0.396**	**0.370**	0.425
TF-IDF	0.538	0.531	0.545	0.382	0354	0.414

We implement four algorithms. Table 4 & 5 show the average scores achieved by them. We implement four algorithms. The bold values indicate the best scores among the all models implemented. We can see that the model ' DFReeKLIM + BM25 ' outperforms the other models.

6 Discussion

Comparison with other systems revealed that the advancements made in our ongoing work were not as prominently demonstrated in our notebook submission. Our algorithm is not able to create good summaries for the event on a specific day. It is because my model was not able to re-rank documents efficiently for a specific query with model "BM25" and "DFReeKLIM" individually. Because of this the most relevant document was not used to create summaries and we were not using query embeddings. Post-submissions, we implement two more models: "BM25 + DFReeKLIM and TF-IDF". We can compare the performance of all models in Table 4 & 5 and say that the model ' DFReeKLIM + BM25 ' outperforms the other models because we used query embedding. We are confident that this approach holds potential for yielding promising results, especially for labels with limited resources. However, further investigation is necessary to accurately assess the compatibility between task pairs and parameters before initiating training.

7 Conclusion

In this paper, we outlined our methodologies for the TREC CrisisFACTS 2023 track, which seeks to condense online data to assist emergency services during crisis events. The overall evaluation outcomes are promising. Further scrutiny will enable us to pinpoint the advantages and drawbacks of the proposed methodologies. We used BM25, DFReeKLIM and TF-IDF based retrieval model with different combinations to retrieve relevant data based on each query from the data stream coming from various sources like Twitter, Facebook, Reddit and News using Python PyTerrier. In future, we are going to modify our retrieving and re-ranking algorithm to get more efficient and accurate results by embedding techniques and different combinations of algorithms.

8 Acknowledgement

Second Author Sukomal Pal[5] received a fund from the Council of Science and Technology (UPCST[6]), Government of Uttar Pradesh, India and assistance from Mr. Akhilesh Yadav for this research.

[5] https://cse-iitbhu.github.io/irlab/spal.html
[6] http://www.cst.up.gov.in/

References

[1] Alexis Dusart, Gilles Hubert, and Karen Pinel-Sauvagnat. "IRIT-IRIS at TREC 2022: CrisisFACTS Track". In: NIST Special Publication 500-338 (2022). Ed. by Ian Soboroff and Angela Ellis. URL: https://trec.nist.gov/pubs/trec31/papers/IRIT%5C_IRIS.R.pdf.

[2] Thi Huyen Nguyen and Koustav Rudra. "L3S at the TREC 2022 CrisisFACTS Track". In: NIST Special Publication 500-338 (2022). Ed. by Ian Soboroff and Angela Ellis. URL: https://trec.nist.gov/pubs/trec31/papers/eXSum22.R.pdf.

[3] Jayr Alencar Pereira, Robson do Nascimento Fidalgo, and Roberto de Alencar Lotufo. "Using Neural Reranking and GPT-3 for Social Media Disaster Content Summarization". In: NIST Special Publication 500-338 (2022). Ed. by Ian Soboroff and Angela Ellis. URL: https://trec.nist.gov/pubs/trec31/papers/NM.unicamp.R.pdf.

[4] Nathaniel W. Rollings, Peter A. Rankel, and Douglas W. Oard. "Multi-Faceted Question Fusion in the TREC 2022 CrisisFACTS Track". In: *Proceedings of the Thirty-First Text REtrieval Conference, TREC 2022, online, November 15-19, 2022*. Ed. by Ian Soboroff and Angela Ellis. Vol. 500-338. NIST Special Publication. National Institute of Standards and Technology (NIST), 2022. URL: https://trec.nist.gov/pubs/trec31/papers/umcp.R.pdf.

[5] Anita Saroj and Sukomal Pal. "Use of social media in crisis management: A survey". In: *International Journal of Disaster Risk Reduction* 48 (2020), p. 101584. ISSN: 2212-4209. DOI: https://doi.org/10.1016/j.ijdrr.2020.101584. URL: https://www.sciencedirect.com/science/article/pii/S221242091931684X.