

# Query Expansion for Crisis Events

Jack Cheverton, Sharon Gower Small & Ting Liu

Siena College Institute for Artificial Intelligence  
515 Loudon Road  
Loudonville, NY 12211  
jt18chev, ssmall, tliu  
@siena.edu

## Abstract

This paper discusses our work and participation in the Text Retrieval Conference (TREC) CrisisFacts Track (CFT) of 2023. Social media systems can be a valuable source of information for emergency responders during a crisis event if harnessed properly. The task of extracting relevant information as a crisis event is unfolding is a unique information retrieval task, such that it is attempting to detect posts relative to a specific event that is ongoing and evolving in real time. The CFT is in its second year of fostering research in this area. The CFT team has supplied multi-stream datasets from several disasters, covering Twitter, Reddit, Facebook, and online news sources (from the NELA News Collection<sup>1</sup>). We will report on our query expansion work that we implement to participate in the CFT.

## 1. Introduction

The Incident Streams Track (Buntain et al., 2020), first run in 2018, is a program in the Text Retrieval Conference (TREC) (Voorhees 2007). TREC is a program co-sponsored by the National Institute of Standards and Technology (NIST) and the U.S. Department of Defense and it focuses on supporting research in information retrieval and extraction, and to increase availability of appropriate evaluation techniques. The CFT (McCreadie & Buntain 2022) evolved from the Incident Streams Track and was run for its second consecutive year in 2023.

Public Information Officers are tasked with monitoring social media streams in order to identify any requests for help. There are currently no satisfactory tools to aid them in this process and it becomes mostly manual. Given that it is quite obvious that information may not be provided to incident commanders in a timely fashion.

The CFT is in its second year of fostering research in this area. The CFT team has supplied multi-stream datasets from several disasters, covering Twitter, Reddit, Facebook, and online news sources (from the NELA News Collection). We had a team of two undergraduate researchers work for 6 weeks to generate explore ideas that we believed could potentially boost performance for this type of task. This paper discusses our work and participation in the TREC CrisisFacts Track of 2023.

---

<sup>1</sup> A Dataset of U.S. Local News Articles

## 2. CrisisFacts Track Literature Review

Many techniques to create a system that can effectively and efficiently label information types and priority levels have been tried. The Terroir team at the University of Glasgow (Hepburn et al., 2020) used text-based features, examining what distinguishes tweets between priority levels, as well as numerical features including the number of hashtags and the presence of URLs and other media. The team trained on Balanced Random Forest (BRF) and Easy Ensemble (EE) models and found that BRF models performed higher than EE. While the BRF model did well in identifying information types, it did not score high when predicting priority levels. A team at University College Dublin (Wang and Lillis, 2020) worked with multi-task transfer learning, fine-tuning transformer encoder-based models like BERT and sequence-to-sequence transformers like T5. They had two scenarios, one where they used an encoder model and one where they used a sequence-to-sequence model. For each of the scenarios, they trained two prediction models, one for predicting post category, and one for predicting post priority. Their work outperformed other runs in information type classification and in predicting priority levels.

One of our hypotheses was that solutions needed to be event specific to take system up a level in their performance. A system would then just need to identify the event type from the tweet and then apply the appropriate event specific solution in order to predict the information type and priority level. Work has been done in first story detection of events through Twitter and detection of single events. Wang and Goutte work with clustering temporal profiles of hashtags to then input into multivariate change point detection algorithms to find changes in events in Twitter streams (Wang and Goutte., 2020). Their method outperforms others in that it identifies up to 40% of subevents in the datasets tested. A team at University of Edinburgh (Wurzer et al., 2020) used k-term hashing for first story detection of events that operates  $O(1)$  per tweet. Rather than comparing a tweet with each that came before it, it can be compared with just one model that combines all previous tweets to greatly increase efficiency. Studies from Radboud University explore the use of estimating future events based on tweet text (Hürriyetoğlu et al., 2014). They parsed tweets for keywords and their variations and predicted the time of the event. Their systems typically had a margin of error of less than ten hours.

## 3. Track Overview

The overarching goal of CFT is to support emergency response services' efforts to harness the information in social media to respond better to social crisis situations. Participants were provided with multiple streams of event relevant tweets for a given crisis event as disaster-day pairs. The events developed for the 2023 run along with the data streams provided for each event can be seen in Table 1.

2023 New Events						
eventID	Title	Type	Tweets	Reddit	News	Facebook
CrisisFACTS-009	Beirut Explosion, 2020	Accident	94,892	3,257	1,163	368,866
CrisisFACTS-010	Houston Explosion, 2020	Accident	58,370	5,704	2,175	6,281
CrisisFACTS-011	Rutherford TN Floods, 2020	Floods	11,019	475	268	9,116
CrisisFACTS-012	TN Derecho, 2020	Storm/Flood	49,247	1,496	15,425	13,521
CrisisFACTS-013	Edenville Dam Fail, 2020	Accident	16,527	2,339	961	8,358
CrisisFACTS-014	Hurricane Dorian, 2019	Hurricane	86,915	91,173	7,507	370,644
CrisisFACTS-015	Kincade Wildfire, 2019	Wildfire	91,548	10,174	339	35,011
CrisisFACTS-016	Easter Tornado Outbreak, 2020	Tornadoes	91,812	5,070	750	34,343
CrisisFACTS-017	Tornado Outbreak, 2020 Apr	Tornadoes	99,575	1,233	217	19,878
CrisisFACTS-018	Tornado Outbreak, 2020 March	Tornadoes	95,221	16,911	641	87,242

**Table 1: CFT 2023 Events and their Stream data**

Participants were tasked with extracting a non-redundant list of atomic facts and assigning an importance score to each fact indicating how critical the information is. TREC supplied us with a gold standard fact list from the 2022 track run. The events and the data sources for this training data can be seen in Table 2. The events in the training data included: Wildfire, Hurricane and Flood. Therefore we had no training data for the new Accident and Tornado events of 2023.

eventID	Title	Type	Tweets	Reddit	News	Facebook
CrisisFACTS-001	Lilac Wildfire 2017	Wildfire	41,346	1,738	2,494	5,437
CrisisFACTS-002	Cranston Wildfire 2018	Wildfire	22,974	231	1,967	5,386
CrisisFACTS-003	Holy Wildfire 2018	Wildfire	23,528	459	1,495	7,016
CrisisFACTS-004	Hurricane Florence 2018	Hurricane	41,187	120,776	18,323	196,281
CrisisFACTS-005	Maryland Flood 2018	Flood	33,584	2,006	2,008	4,148
CrisisFACTS-006	Saddleridge Wildfire 2019	Wildfire	31,969	244	2,267	3,869
CrisisFACTS-007	Hurricane Laura 2020	Hurricane	36,120	10,035	6,406	9,048
CrisisFACTS-008	Hurricane Sally 2020	Hurricane	40,695	11,825	15,112	48,492

**Table 2: CFT 2022 events that the gold standard fact-list was generated from**

For each event participants were provided with an Event Definition (Figure 1) and a set of User Profiles (can be thought of as queries). The Event Definition contains informational identifiers as well as the type of event, the url for the wikipedia coverage of the event and a natural language description of the event.

```
{
  "eventID": "CrisisFACTS-001",
  "trecisId": "TRECIS-CTIT-H-092",
  "dataset": "2017_12_07_lilac_wildfire.2017",
  "title": "Lilac Wildfire 2017",
  "type": "Wildfire",
  "url": "https://en.wikipedia.org/wiki/Lilac Fire",
  "description": "The Lilac Fire was a fire that burned
}
```

**Figure 1: An example Event Definition**

The User Profiles (Figure 2) for each event include its id, a set of indicative terms and the category of the query. The query set was developed by the track organizers by examining the information needs of the disaster summaries ICS-209<sup>2</sup> data collection.

<sup>2</sup> Actual disaster summaries from the US National Incident Management System between 1999-2000.

```
[{
  "queryID": "CrisisFACTS-General-q001",
  "indicativeTerms": "airport closed",
  "query": "Have airports closed",
  "trecisCategoryMapping": "Report-Factoid"
},
{
  "queryID": "CrisisFACTS-General-q002",
  "indicativeTerms": "rail closed",
  "query": "Have railways closed",
  "trecisCategoryMapping": "Report-Factoid"
},
{
  "queryID": "CrisisFACTS-General-q003",
  "indicativeTerms": "water supply",
  "query": "Have water supplies been contaminated",
  "trecisCategoryMapping": "Report-EmergingThreats"
},
...
{
  "queryID": "CrisisFACTS-Wildfire-q001",
  "indicativeTerms": "acres size",
  "query": "What area has the wildfire burned",
  "trecisCategoryMapping": "Report-Factoid"
},
{
  "queryID": "CrisisFACTS-Wildfire-q002",
  "indicativeTerms": "wind speed",
  "query": "Where are wind speeds expected to be high",
  "trecisCategoryMapping": "Report-Weather"
},
...
]
```

**Figure 2: Sample of User Profiles**

## 4. Our Approach

Given our short time frame for work we decided to focus just on the comprehensiveness metric by experimenting with query expansion. Comprehensiveness is defined by the track organizers as: "a summary's fact-recall with higher values being better." The highest comprehensiveness achieved on all of the 2022 runs, showed a system best of only 21.7%. Meaning the summary covered around 21% of all the facts for an event for any given day.

For each query we utilized the indicative terms. We used these terms during the query retrieval process to score the query against the facts. Our approach was to expand each query by adding indicative terms that could help fine tune the query retrieval process.

We began by going through each event and extracting text from the facts assigned to that event. The extraction process used six different methods. The first two were to extract n-grams from the facts. These first two methods searched for bigrams (2-grams) and trigrams (3-grams) respectively.

To do this, the facts were first separated by sentences and then filtered based on keywords relative to the event. All sentences which did not include this keyword were discarded. For events 1, 2, 3, 6, and 15, the word "fire" was used. For events 4, 7, 8, and

14, the word “hurricane” was used. For events 5 and 11, the word “flood” was used. For events 9 and 10, the word “explosion” was used. For event 12, the word “storm” was used. For event 13, the word “dam” was used. Finally, for events 16, 17, and 18, the word “tornado” was used. When checking if the keyword was included, it checked for substrings (such as searching for the word “fire” in “firefighter”) and ignored case sensitivity. When all sentences with the keyword were collected, all bigrams and trigrams from the sentences were extracted.

Methods three and four also extracted bigrams and trigrams, but excluded stop words. Method five and six extracted noun phrases from sentences that included the searched-for keywords. To do this extraction, the Python module NLTK was used to first tag each word. Using these tags, noun phrases were extracted by using regular expressions. Each method used a slightly different regular expression, with method five usually gathering less phrases than method six.

In addition to gathering this information from the facts, we also utilized each event’s assigned Wikipedia article. Each article’s summary (the top portion of a wikipedia article) was extracted, and then split into sentences. Sentences with the specified keyword were saved while all others were discarded, and the six different methods were applied to these sentences. It should be noted that events 11 and 12 had no wikipedia page assigned to them and instead had a news article instead. The text from the news article was manually copied into a text document, and the process continued using these documents.

When the extraction process was complete, all terms were placed into a text document. Each of the eighteen events had six documents associated with it for each of the different methods. For each of these event-method combinations, there are two different versions made: one that stores the text extracted from the facts and one for the wikipedia articles. At this point, there are twelve different types query expansion techniques. The first six are gathered from the six methods used on the facts and the last six are gathered from the wikipedia articles. Each different event has its own version of these twelve files.

Included in all of the gathered terms is information that is not needed for the purposes of query expansion. Good terms need to be sorted out from useless terms. We did this by running a query retrieval process which would grade two different sets of text passages by similarity. The first set of text, the index, would be a list of all of the original queries. The second set, the query set, would be made up of the expansion terms included in one of the twelve different types of text files. This would let us score the proposed expansion terms. Any term that was scored below the average of all terms was discarded.

In summary, for each of the eighteen different events, twelve groups of terms are created. Six of these groups are gathered from the facts while the other six are gathered from Wikipedia. The result is twelve different versions of the original queries with different indicative terms for each event.

During the development of this system, we were only able to run it on 2022 data. This means that there were only eight events available for training. Using this auto-grading script from the track organizers, all twenty-four runs were produced and then graded.

One method proved to be the best during the grading process: The trigram gathering method that did not remove stopwords. It performed the best consistently both when it was used on the facts and on the Wikipedia articles. With this knowledge in mind, four final runs were created and submitted to TREC on the 2023 data. The first was a baseline that changed none of the indicative terms and that used the first type of run produced from the

run creation script. The second used the trigram gathering method on the facts and also used the first type of run produced from the script. The third and fourth run used the trigram method on the Wikipedia articles.

## References

Hepburn, Alexander J. and Richard McCreadie, 2020. *University of Glasgow Terrier Team (uogTr) at the TREC 2020 Incident Streams Track*. University of Glasgow, UK.

Hürriyetoğlu, Ali, Nelleke Oostdijk, and Antal van den Bosch, 2014. *Estimating Time to Event from Tweets Using Temporal Expressions*. Radboud University Nijmegen, The Netherlands.

McCreadie, Richard and Cody Buntain. CrisisFACTS: Building and Evaluating Crisis Timelines. In Proceedings of The Thirty First Text REtrieval Conference, Gaithersburg, Maryland, November 2022.

Voorhees, Ellen M. 2007. Overview of TREC 2007. In Proceedings of The Sixteenth Text Retrieval Conference (TREC 2007).

Wang, Congcong and David Lillis, 2020. *Multi-task transfer learning for finding actionable information from crisis-related messages on social media*. University College Dublin.

Wurzer, Dominik, Victor Lavrenko, and Miles Osborne, 2015. *Twitter-scale New Event Detection via K-term Hashing*. University of Edinburgh.