CrisisFACTS 2023 – Overview Paper

Cody Buntain
University of Maryland
cbuntain@umd.edu

Benjamin D. Horne
University of Tennessee
bhorne6@utk.edu

Amanda Hughes
Brigham Young University
amanda.hughes@byu.edu

Muhammad Imran
Qatar Computing Research Institute
mimran@hbku.edu.qa

Richard McCreadie
University of Glasgow
Richard.Mccreadie@glasgow.ac.uk

Hemant Purohit
George Mason University
hpurohit@gmu.edu

1 Abstract

This paper describes the second and final edition of CrisisFACTS, run for TREC 2023. In this edition, we transitioned from a two-phases of manual assessment (fact identification followed by fact matching) to a single-phase approach where facts are manually identified from analysis of the output of the pooled systems and that output is matched to facts as a single step. We also introduced fact quality ratings, allowing us to distinguish between Useful, Poor, Redundant and Lagged (out-of-date) facts. We experimented with replacing the manual matching of participant outputs to facts with automatic matching techniques (both exact and semantic matching). And we added 7 new crisis events. For evaluation, we compared results from standard similarity-based summarization techniques to manual assessments and, while we show some similarity in rankings across methods, we point to paths for improving similarity-based summarization, as these methods are likely to be increasingly needed in the face of generative models.

2 Introduction

One of the core tasks that an emergency response agency needs to complete each day that can be significantly enhanced with online information is producing after-action reports. These are incident summaries written for response personnel, government officials and media agencies. They succinctly provide an overview of the event and major developments since the last report. The TREC Crisis Facts and Cross-Stream Temporal Summarization (CrisisFACTS) initiative aims to tackle the fully automated production of these reports/summaries using information from heterogeneous online content streams. These summaries will support attention-allocation for disaster-response personnel by highlighting new developments in the event. To this end, the primary use-case for CrisisFACTS is the creation of incident status summaries, in the form of *event timelines*, i.e. itemized lists of what important happened (Allan et al. 2001).

2023 marks the second (and final) year that CrisisFACTs will run. For a detailed overview of the first year of the track and lessons learned we recommend reading the extended overview paper published at ISCRAM 2023 (McCreadie and Buntain 2023). In summary, we were happy with the new multi-stream dataset and collection methodology, the manual assessment worked reasonably and the initial round of participant systems were able to produce meaningful timelines, although there was significant scope for improvement in system recall. On the other hand, there were concerns regarding the reusability of the corpus due to the need for manual matching between stream items and facts, as well as concerns regarding participant systems returning information that was not in our initial fact list (and hence those systems miss out on credit they should have been awarded). We aim to rectify both these issues in this second edition.

The primary changes to the second edition of the task are as follows:

- We transitioned from a two-phases of manual assessment (fact identification followed by fact matching) to a single-phase approach where facts are manually identified from analysis of the output of the pooled systems and that output is matched to facts as a single step
- Fact quality rating was introduced, allowing us to distinguish between Useful, Poor, Redundant and Lagged (out-of-date) facts.
- We experimented with replacing the manual matching of participant outputs to facts with automatic matching techniques (both exact and semantic matching).
- 7 new crisis events were added, but these new 2023 events do not have ICS-209 forms for use as an alternative ground truth.

The second edition of CrisisFACTS attracted increased participation in comparison to the first edition with 11 participating teams submitting 27 runs. Indeed, it was not due to lack of interest in the task that CrisisFACTS is concluding, but rather due to unavailability of an experienced team to continue to run it in 2024.

2.1 CrisisFACTS Task Formulation

The core objective in CrisisFACTS is to facilitate stakeholders' need for concise descriptions of new developments in crises, gathered from multiple online data streams. Figure 1 outlines the CrisisFACTS task formulation to support this objective, wherein a stakeholder wants to see daily summaries of important, new developments in an ongoing emergency event. By producing these summaries at regular intervals, e.g., when a new staff shift comes on duty, this task provides attention support by communicating what new events most require stakeholder attention. This process is, in effect, an automatic version of after-action reporting that emergency response personnel perform at the end of each day during an on-going emergency. Currently, such summarization efforts are performed by hand, and CrisisFACTS intends to produce such summaries automatically, using information extracted from online news and social media.

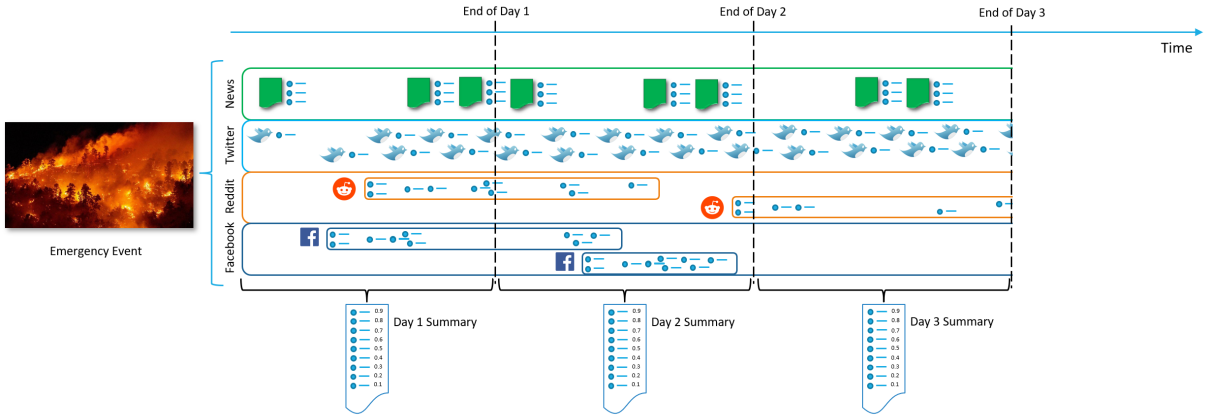

Figure 1. The Core CrisisFACTS Task Formulation. During an emergency event, disaster management personnel need information about new and important developments, which may be scattered across multiple data streams. The CrisisFACTS task is to summarize important information across these streams and across each day of the event, that stakeholders may then use for planning, resource allocation, etc.

To this end, the CrisisFACTS data challenge provides participants with multiple streams of crisis-relevant data, currently including Twitter, Reddit, Facebook, and online news sources, broken down by day. Participant systems consume these daily multi-platform streams and produce a series of daily summaries for a given crisis event. These streams consist of approximately sentence-length items (i.e., stream items), which participants use to construct a short timeline of important information, filtering out and compressing content where appropriate. Systems then rank their resulting list of stream items for each <event,day> by item "importance", where the top 'k' scored items form the final event-day summary.

3 Terminology

For reference we define a common set of terminology for use in the remainder of this report:

- **Event**: A real-world disaster that was sufficiently serious to be reported about online.
- **User Profile**: An itemised list of general and event-type-specific queries representing a responder's information needs.
- **Summary Requests**: A request for a timeline summary for a particular period of time. For CrisisFACTs, this is always during an event and corresponds to a one day period. These are the equivalent to a topic in more traditional IR tasks. For clarity, for the remainder of this paper we refer to summary requests as 'event-day pairs'.
- **Content Streams**: Streams of text content collected about an event (although not all of the content is relevant). The stream types are Twitter, Reddit, Facebook and News.
- **Stream Item**: A roughly sentence-length snippet of text with a timestamp from the content streams. These are the input that participant systems use to build their summaries/timelines. Each stream item has a unique 'stream ID'.
- **Fact (Ground truth)**: A snippet of text that conveys an important piece of information about the event from the perspective of the User Profile. These are manually extracted from participant system output.
- **Timeline Item**: A snippet of text with a timestamp conveying information about the event at a point in time, produced by a participant system. Participants return a list of timeline items for each Summary Request/Event-Day Pair. Participant systems assign an 'importance score' to each timeline item, indicating which items they think are the most important to be included in the timeline if it was fixed length. These importance scores are used for pooling during assessment.
- **Extractive Run**: The timeline output for a participant system for each test event, where the timeline items for each event-day pair are extracted verbatim from the content streams. Each timeline item in this case is also a stream item.
- **Abstractive Run**: The timeline output for a participant system for each test event, where the timeline items for each event-day pair are generated by the participant system using one or more stream items.

4 Participant Approaches

For this 2023 edition, CrisisFACTS has received 27 runs from 11 participant teams. Of these runs, the majority were automatic (23 of 27 versus 4 manual), extractive (17 of 27 versus 8 abstractive and 2 "both"), and used all four data streams (21 using Twitter, Facebook, Reddit and news, with 5 using all but Facebook, and one using only Twitter). Further, only 5 runs actively use prior results from TREC-IS. Of the abstractive runs submitted, these runs are split between using GPT-3.5/4 and LLaMA models.

5 Two Approaches to Run Assessment

While a key motivation for CrisisFACTS's evaluation is to accurately measure the precision and recall of responder-relevant information in participant timelines. To this end, we include two types of evaluation: **Fact-Based, Manual Assessment**; a daily, fact-based summarization that relies on manual assessment of individual facts composing a single day's summary, and **Whole-Event Summary Assessment**; a wholistic summarization evaluation as bridge between the TREC-IS initiative and more traditional summarization evaluation based on gold-standard summaries. This second evaluation carries over from CrisisFACTS 2022 as well and provides continuity across the years.

5.1 Fact-Based, Manual Assessment

Manual assessment on sets of facts across CrisisFACTS runs requires a method for pooling runs, which is complicated by the potential use of abstractive text generation. That is, two runs may produce a very similar but not identical fact list because the underlying language model produces slightly different output. Consequently, de-duplicating these sets of facts cannot be done simply through unique stream IDs from the CrisisFACTS stream, as the abstractive text generation process may use several stream IDs as evidence for a single fact. Hence, we first construct a method to de-duplicate and pool fact lists across runs for each event-day pair. This pooling goes to a depth k, ranked by run-reported priority, to extract the top-most important facts from that run's event-day output. Following this pooling, we then create a single document for each event-day pair, composed of all pooled facts, which we send to NIST assessors for assessment. Assessment results provide fact-level labels for all pooled facts, which we then use as the reference set for evaluating comprehensiveness (recall) and redundancy (precision) for each run.

5.1.1 Pooling and De-Duplicating Submitted Runs

Our assessment assumes that we can construct a single summary for each event-day pair from the union of facts for that event-day pair across all submitted runs. This assumption requires a method for identifying potentially duplicate facts in this unified set to limit redundant assessment effort. A reasoned approach to de-duplication is especially critical in CrisisFACTS given the potential for abstractive models and recent advances in large language models, so we cannot rely on simply de-duplicating summaries by stream ID (e.g., Twitter ID, Reddit ID, news article URL, etc.). Additionally, a method for de-duplicating these sets may increase the long-term re-usability of the resulting CrisisFACTS test collection, as a de-duplication method will allow new runs to match at least some subset of found facts against the test collection even if their new fact is not an exact duplicate of a fact in the test collection.

To this end, for a single event-day pair, we are given a collection of submitted runs $r_{rid} \in R$, where each run contains a set of facts $f_{fid} \in r_{rid}$. We take only the top $k = 32$ facts from a given run, ordered by the reported 'importance' score. We can take the union of all facts from this event-day across all runs to obtain a unified collection of facts $f_{fid,rid} \in F$. To identify potential duplicates in F, we examine all pairs $p \in F \times F$ and assess similarity between these two facts via the BERTscore measure. If $BERTscore(p1, p2) \geq 0.91$, we tag this pair as a potential duplicate. This threshold comes from empirical assessment on CrisisFACTS 2022 facts.

The prior step provides potential pairwise duplicates. To collapse this set of duplicate pairs into a parsimonious set of non-duplicate facts, we use a greedy approach where we rank facts F by the number of duplicates in which they appear. We treat the similarity matrix induced on $F \times F$ as an adjacency matrix where $A_{i,j} = 1$ if $BERTscore(pi, pj) \geq 0.91$ else 0, we rank individual facts by their total degree.

Starting with the highest-degree fact (i.e., the fact with the highest coverage of duplicates), we collapse all neighboring facts into a single meta-fact, keeping the fact text from the highest-degree fact, and removing all neighboring facts from the graph. We then calculate the "importance" of this meta-fact as the median importance score of the associated collapsed facts. We then proceed to the next highest-rank fact that has not been removed from the graph. This greedy approach prioritizes facts that are broadly similar to many other facts to maximize coverage of submitted facts from the runs, such that assessors' annotations will cover many submissions.

Following this de-duplication, we rank resulting collapsed meta-facts by their aggregated importance score (again, calculated as the median importance score across individual facts). We then concatenate this ranked list into a single document representing the summary for this event-day pair, maintaining a map of the spans of text and their associated meta-facts.

5.1.2 NIST Assessment

For the fact-based NIST assessment, the main assessment task is span annotation, where we present NIST assessors with a single document for each event-day pair, ordered by event-day, and ask them to label spans according to the following:

- Useful Fact – Spans of text with this label represent facts that are clearly useful for providing information when filling out the ICS209 Incident Status Summary form.
- Poor Fact – Spans of text with this label represent facts that could potentially be useful but are confusing, lack a critical detail, are poorly formed, or otherwise difficult to parse.

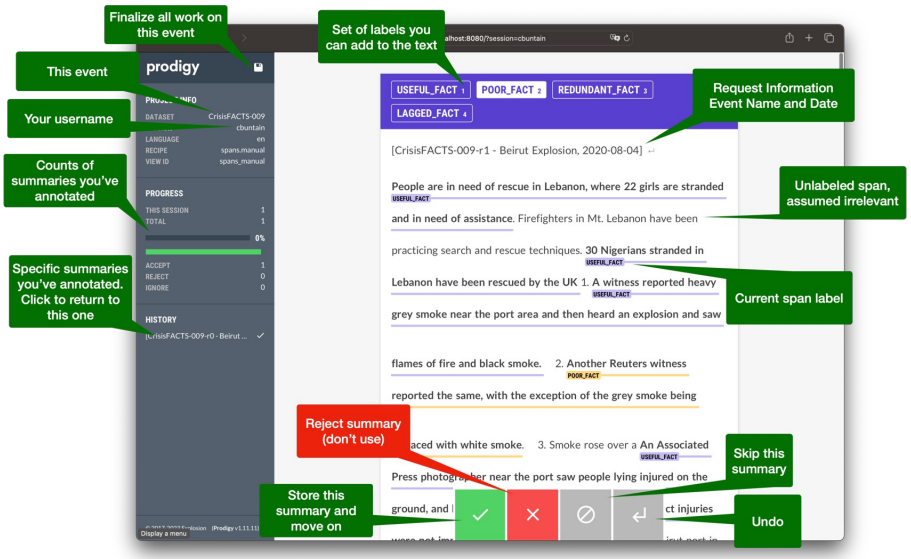

Figure 2. Prodi.gy Evaluation Interface for Fact-Span Labeling.

- Redundant Fact – Spans of text with this label represent facts that are already covered by an earlier span of text in the current summary.
- Lagged Fact – Spans of text with this label represent facts that were already covered in a previous day's summary.

5.1.2.1 Assessment Interface Given a summary constructed for a particular event-day pair, e.g., CrisisFACTS-009-r1, assessors use the Prodi.gy annotation tool (see `https://prodi.gy` for documentation) to annotate these spans. Each assessor is assigned a set of CrisisFACT events, and each event has a set of associated days (e.g., the Beirut explosion, CrisisFACTS-009, has 7 days associated with it). Each of these days will be presented as a summary for the assessor to annotate according to the above fact set.

For a given summary (i.e., one page of text to annotate in the Prodigy interface), the assessor should review the summary and annotate spans of text according to the facts listed above. Unannotated spans are assumed to be irrelevant, so the assessor need not ensure every word is tagged with a label. A label can be removed simply by clicking the span label under the text. After the assessor has identified all facts needing annotation in the current summary, they should click the green checkmark button to store this summary.

5.1.2.2 Propagating Span Labels to Facts Every event is assessed by at least two NIST assessors. We compute the intersection for all labeled spans with the facts that comprise the summary to attach labels to our pooled, de-duplicated meta-facts. To compute a single label for a fact from multiple assessors, we take the maximum score for that fact. E.g., if assessor 1 labels a fact as "poor" but assessor 2 labels that same fact as "useful", we assign the fact a "useful" label. Likewise, if a single fact has multiple spans, we take the maximum label for that span as the fact's label. For example, in Figure 3, the single fact has multiple spans, as an assessor has labeled the first phrase as a "lagged fact" since it refers to events in the previous day, but the second phrase is tagged as a "useful fact"; entire fact then receives the maximum score for all its associated spans, so it is labeled as a "useful fact". The assessment interface used to produce these span labels is shown in Figure 2. These labels can be found in the 'fact_list' field of the 'final-annotated-facts-results.json' file.

5.1.3 Results from Manual Assessment

For CrisisFACTS 2023, we have engaged six NIST assessors for the above span-labeling task. Each assessor has assessed at least six separate CrisisFACTS events, with the event-to-assessor assignments shown in Table 1. Note that the NIST assessors re-assessed the events from CrisisFACTS 2022, as how we performed manual evaluation

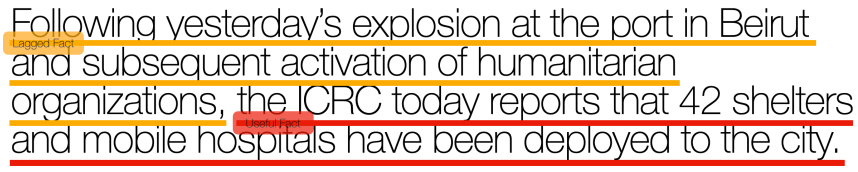

Figure 3. Example Fact with Multiple Spans.

Table 1. Assessors Assigned to Each Event. All but the final two events have two assessors assigned to them, and each assessor has at least six events they have assessed.

CrisisFACTS Event	Assessors
CrisisFACTS-001	assr-01, assr-03
CrisisFACTS-002	assr-01, assr-05
CrisisFACTS-003	assr-05, assr-06
CrisisFACTS-004	assr-02, assr-06
CrisisFACTS-005	assr-02, assr-04
CrisisFACTS-006	assr-04, assr-06
CrisisFACTS-007	assr-05, assr-06
CrisisFACTS-008	assr-01, assr-05
CrisisFACTS-009	assr-01, assr-02
CrisisFACTS-010	assr-02, assr-04
CrisisFACTS-011	assr-03, assr-04
CrisisFACTS-012	assr-03, assr-05
CrisisFACTS-013	assr-05, assr-06
CrisisFACTS-014	assr-02, assr-06
CrisisFACTS-015	assr-02, assr-04
CrisisFACTS-016	assr-03, assr-04
CrisisFACTS-017	assr-01, assr-02, assr-03, assr-06
CrisisFACTS-018	assr-01, assr-02, assr-03, assr-05, assr-06

changed between editions. To enable this we asked the 2023 edition participant systems to provide summaries for both 2022 and 2023 events.

Turning to assessor labels, Table 2 shows label frequency and totals for each assessor. This table also shows that, in general, the 'USEFUL_FACT' label is the most frequent span label (with the exception of assessor 3). Per-assessor frequencies are not directly comparable, however, as frequencies are tied to the number of days and information quality across CrisisFACTS events.

Breaking down assessor labels by individual summary request-id (i.e., by day), as shown in Table 3, some patterns emerge. In particular, we see that, generally, earlier days have more facts associated with them, with a higher frequency of useful, poor, and redundant facts early in the event. We confirm this finding with a linear regression model, regressing the log-transformed number of useful facts on the day index plus a constant, finding a significant effect ($B = -0.2996$ for the day, $SE = 0.089$, $p < 0.01$). Lagged facts (facts that should have been reported on an earlier day) appear to increase, peak, and the decrease as the event progresses–these peaks tend to appear in days after the peak in useful facts–though this effect is not statistically significant in our linear model.

Table 3. Fact Labels By Request ID.

Request ID	USEFUL_FACT	POOR_FACT	REDUNDANT_FACT	LAGGED_FACT
CrisisFACTS-009-r0	11	3	0	0
CrisisFACTS-009-r1	103	0	137	4
CrisisFACTS-009-r2	94	4	48	154
CrisisFACTS-009-r3	74	6	11	71
CrisisFACTS-009-r4	77	1	16	68
CrisisFACTS-009-r5	63	5	10	67
CrisisFACTS-009-r6	54	4	9	67

CrisisFACTS-010-r0	5	0	1	0
CrisisFACTS-010-r1	115	16	178	3
CrisisFACTS-010-r2	55	10	55	97
CrisisFACTS-010-r3	27	4	15	100
CrisisFACTS-010-r4	23	5	54	59
CrisisFACTS-010-r5	4	0	1	27
CrisisFACTS-011-r0	18	19	10	0
CrisisFACTS-011-r1	28	33	18	11
CrisisFACTS-011-r2	57	49	22	36
CrisisFACTS-011-r3	16	26	8	15
CrisisFACTS-011-r4	1	1	5	0
CrisisFACTS-012-r0	3	21	0	0
CrisisFACTS-012-r1	168	48	9	0
CrisisFACTS-012-r2	116	59	26	40
CrisisFACTS-012-r3	70	57	23	42
CrisisFACTS-012-r4	56	31	11	28
CrisisFACTS-012-r5	0	17	0	0
CrisisFACTS-012-r6	0	24	0	0
CrisisFACTS-013-r0	118	121	38	0
CrisisFACTS-013-r1	91	81	47	19
CrisisFACTS-013-r2	71	74	77	19
CrisisFACTS-013-r3	96	56	55	26
CrisisFACTS-013-r4	84	34	40	7
CrisisFACTS-013-r5	5	22	3	0
CrisisFACTS-013-r6	0	17	0	0
CrisisFACTS-014-r0	175	81	73	0
CrisisFACTS-014-r1	118	60	58	33
CrisisFACTS-014-r2	120	33	14	74
CrisisFACTS-014-r3	145	52	40	26
CrisisFACTS-014-r4	105	77	7	33
CrisisFACTS-014-r5	124	65	25	36
CrisisFACTS-014-r6	118	41	6	43
CrisisFACTS-015-r0	102	8	42	13
CrisisFACTS-015-r1	115	14	71	19
CrisisFACTS-015-r2	151	6	64	23
CrisisFACTS-015-r3	102	3	31	49
CrisisFACTS-015-r4	76	2	11	22
CrisisFACTS-015-r5	23	1	20	13
CrisisFACTS-015-r6	20	4	26	2
CrisisFACTS-016-r0	108	29	42	47
CrisisFACTS-016-r1	126	43	40	90
CrisisFACTS-016-r2	75	22	38	96
CrisisFACTS-016-r3	62	13	14	59
CrisisFACTS-016-r4	2	0	8	4
CrisisFACTS-017-r0	94	25	11	1
CrisisFACTS-017-r1	236	63	107	3
CrisisFACTS-017-r2	251	89	76	99
CrisisFACTS-017-r3	71	29	29	103
CrisisFACTS-017-r4	44	42	22	13
CrisisFACTS-017-r5	33	27	13	23
CrisisFACTS-018-r0	109	116	49	0
CrisisFACTS-018-r1	507	209	218	2
CrisisFACTS-018-r2	351	163	92	94
CrisisFACTS-018-r3	193	97	29	60
CrisisFACTS-018-r4	118	67	28	43
CrisisFACTS-018-r5	23	24	47	5

5.1.3.1 A Note on Assessor Labeling Throughput A surprising finding from this year's assessment is that assessors are much faster at the span-labeling task than we had originally anticipated. Our early expectations were that this task would be cognitively intensive, especially with respect to the lagged-fact label, which requires assessors to keep in mind the prior days' summaries. In reality, however, assessors have been much faster at this task, generally able to evaluate an entire event's worth of summaries (around seven documents, one for each day, per event) in a few days, as opposed to the week we had initially budgeted.

5.1.4 Calculating Redundancy and Comprehensiveness

We can now construct a bipartite graph of run-specific facts connected to CrisisFACTS meta-facts, where each meta-fact has an associated score based on its assessor label–or zero if no label was provided, which suggests irrelevance.

Table 2. Assessor Statistics. Assessors generally have identified most facts as being useful, with poor facts being the second most frequent, and lagged/redundant facts being relatively infrequent.

Assessor	LAGGED_FACT	POOR_FACT	REDUNDANT_FACT	USEFUL_FACT	Total
assr-01	246	251	370	913	1,780
assr-02	1,149	119	826	2,237	4,331
assr-03	822	813	425	755	2,815
assr-04	521	150	600	1,020	2,291
assr-05	82	1,293	592	3,017	4,984
assr-06	205	1,821	394	1,985	4,405
Total	3,025	4,447	3,207	9,927	20,606

For a run's given set of produced facts, we identify the set of adjacent meta-facts. From this set of adjacent meta-facts, we compute redundancy as follows:

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

We then compute comprehensiveness as follows:

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

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

Below, Table 4 provides the average redundancy and comprehensiveness scores for each event-day pair. We provide participant-run-level scores, averaged over the CrisisFACTS 2023 events (event 9-18) in Appendix A. Examining the table suggests redundancy might decrease over time (i.e., as days increase), whereas comprehensiveness appears to increase. Checking these relationships with a standard linear regression model that regresses these metrics on the day show, however, that only comprehensiveness has a significant relationship with the event-day, though the coefficient is small ($B = 0.0034$, $p < 0.001$).

Table 4. Event-Day Score Pairs, Averaged Across the Top Two Submitted Runs From Each Team.

Request ID	Redundancy	Comprehensiveness
CrisisFACTS-009-r0	0.162573	0.164773
CrisisFACTS-009-r1	0.460921	0.064574
CrisisFACTS-009-r2	0.251970	0.060357
CrisisFACTS-009-r3	0.220838	0.065657
CrisisFACTS-009-r4	0.267949	0.064815
CrisisFACTS-009-r5	0.223467	0.064103
CrisisFACTS-009-r6	0.167576	0.065217
CrisisFACTS-010-r0	0.121555	0.092593
CrisisFACTS-010-r1	0.427219	0.068182
CrisisFACTS-010-r2	0.223781	0.069254
CrisisFACTS-010-r3	0.116292	0.066667
CrisisFACTS-010-r4	0.178829	0.079665
CrisisFACTS-010-r5	0.017096	0.109375
CrisisFACTS-011-r0	0.088787	0.078704
CrisisFACTS-011-r1	0.145241	0.112903
CrisisFACTS-011-r2	0.189606	0.070000
CrisisFACTS-011-r3	0.068650	0.070988
CrisisFACTS-011-r4	0.013252	0.072917

CrisisFACTS-012-r0	0.005556	0.055556
CrisisFACTS-012-r1	0.490060	0.063549
CrisisFACTS-012-r2	0.380978	0.069172
CrisisFACTS-012-r3	0.277686	0.076797
CrisisFACTS-012-r4	0.238883	0.090278
CrisisFACTS-013-r0	0.509879	0.080038
CrisisFACTS-013-r1	0.443148	0.090000
CrisisFACTS-013-r2	0.367121	0.082755
CrisisFACTS-013-r3	0.405597	0.078836
CrisisFACTS-013-r4	0.417357	0.098529
CrisisFACTS-013-r5	0.015175	0.066667
CrisisFACTS-014-r0	0.421514	0.062551
CrisisFACTS-014-r1	0.342515	0.059524
CrisisFACTS-014-r2	0.345299	0.064626
CrisisFACTS-014-r3	0.405224	0.064426
CrisisFACTS-014-r4	0.249774	0.056927
CrisisFACTS-014-r5	0.367892	0.063492
CrisisFACTS-014-r6	0.359437	0.062428
CrisisFACTS-015-r0	0.303756	0.072090
CrisisFACTS-015-r1	0.363617	0.064426
CrisisFACTS-015-r2	0.416933	0.070962
CrisisFACTS-015-r3	0.202574	0.066840
CrisisFACTS-015-r4	0.175434	0.072562
CrisisFACTS-015-r5	0.121978	0.078078
CrisisFACTS-015-r6	0.119229	0.076023
CrisisFACTS-016-r0	0.258071	0.061869
CrisisFACTS-016-r1	0.323112	0.065865
CrisisFACTS-016-r2	0.276365	0.073413
CrisisFACTS-016-r3	0.250741	0.087302
CrisisFACTS-016-r4	0.090610	0.131944
CrisisFACTS-017-r0	0.355785	0.070370
CrisisFACTS-017-r1	0.506471	0.066993
CrisisFACTS-017-r2	0.533680	0.066449
CrisisFACTS-017-r3	0.259162	0.076797
CrisisFACTS-017-r4	0.157092	0.067183
CrisisFACTS-017-r5	0.248327	0.140203
CrisisFACTS-018-r0	0.460000	0.089849
CrisisFACTS-018-r1	0.719831	0.063333
CrisisFACTS-018-r2	0.557259	0.065562
CrisisFACTS-018-r3	0.391224	0.072440
CrisisFACTS-018-r4	0.421484	0.096094
CrisisFACTS-018-r5	0.289336	0.151773

5.1.5 Limitations and Potential Problems

For extractive runs, this approach will collapse facts from the same stream IDs, as the $BERTscore()$ between two facts with the same stream ID should be 1.0 and will be treated as a duplicate. This approach may be biased toward covering extractive runs, however, as a stream ID that is returned by many extractive runs will necessarily have a high degree in the similarity network (degree $\geq$ the number of extractive runs returning this fact - 1). Facts from abstractive runs may not have the same degree. We rank meta-facts by the median importance of their related facts to try to offset this possibility, as abstractive methods can still produce facts with high importance; we could alternately rank meta-facts by the number of associated individual facts, but that approach would over-preference facts from extractive runs.

This approach to evaluation is also limited in our ability to assess redundant facts. Since the assessment interface only allows us to mark spans as redundant, we do not have a mechanism to match the redundant fact to the progenitor fact. Hence, if two facts are redundant but fall below the BERTscore threshold, they will be both included in the summary. Since we will not know which prior fact the redundant fact is duplicating, it will be difficult to better tune the BERTscore threshold. That said, a redundant fact could still receive some credit during our final evaluation, and we can decide this credit after examining how frequent the "Redundant Fact" label is following assessment.

Separately, this approach to assessment is limited in recall as relevant facts that are not returned by any run have no path for inclusion in these summaries. Consequently, these aggregate summaries may be incomplete, which fundamentally limits reusability of the resulting test collections. The multi-stream nature of CrisisFACTS will hopefully limit this possibility, as a relevant fact that is simultaneously absent from Facebook, Twitter, Reddit, and news articles is less likely than a fact that is only absent from Twitter.

5.2 Whole-Event Summary Assessment

Above, our evaluation aggregates participant runs into daily collections of facts that assessors tag as useful, and we compare participants to this aggregation of facts. Alternatively, we can compare this aggregation of facts across all event-days with a single summary of the event, in a more common document-to-document summary comparison, exactly as done in CrisisFACTS 2022. To assess these' event-level summaries, we require "gold standard" summaries against which we can compare. We use two sources for these summaries: Wikipedia, wherein the pages for each crisis event has an associated, manually created summary, and the aggregated set of `USEFUL_FACT` facts from each NIST-assessed day.

5.2.1 Wikipedia Summaries

Every crisis event in the CrisisFACTS dataset has an associated Wikipedia entry, and each such page includes a manually developed summary. This "page summary" is available in the `extract` field for a given page in the Wikipedia API or from the `page.summary` field in the Python wrapper for the Wikipedia API. This field generally corresponds to the paragraphs in the Wikipedia page's zeroeth section, which appears above the page's table of contents (see the red box in Figure 4). As such, extracting event summaries from Wikipedia is simply a matter of using the Wikipedia API to collect the `extract` field from each page in the list of crisis-event Wikipedia entries.

Figure 4. Event Summary in Wikipedia Page. The red box captures the section of the Wikipedia article returned for `page.summary` in the Wikipedia API wrapper.

5.2.2 NIST-Assessor Summaries

In addition to comparing against Wikipedia summaries, we can create a whole-event summary using the aggregated set of `USEFUL_FACT` facts tagged by NIST assessors across all days of that event. This approach produces a single document summary of all useful, non-redundant, and non-lagged facts built from all participant systems. This aggregation provides the reference summary for each event.

For constructing a candidate summary from a participant system, we perform two aggregations: First, for each day, we again take the top-k most important facts from that day to create a daily summary. Second, we aggregate all the daily summaries into a single document for that event. As with the fact-based assessment, we set $k = 32$ here for every run.

5.2.3 Results of Summary-to-Summary Similarity

Our first question focuses on the quality of summaries produced by participant systems. Summarization is a well-researched task with multiple automated metrics for comparing summaries. The "Recall-Oriented Understudy for Gisting Evaluation", or ROUGE, metric is common in this space and operates by comparing n-grams of various lengths between the submitted and reference summaries. ROUGE's reliance on matching exact tokens, however, may be particularly problematic in the CrisisFACTS context, as social media content has substantial stylistic differences

from Wikipedia text, news articles, and professional writing. To address this concern, we also use BERTScore (Zhang et al. 2020) to assess summaries and via BERT-based contextual embeddings.

Having selected the two families of metrics and before actually assessing participant systems, we first compare our two target summaries: Wikipedia-based summaries, and summaries built from the NIST-constructed fact lists. Evaluating pairs from these references establishes valuable context for later comparisons between participant systems and these targets.

Table 5 shows the ROUGE-2 and BERTScore metrics, averaged across each crisis event, for each pair of gold-standard summaries. We also provide scores for individual runs, averaged across the events, in Table 9 in Appendix A.

Table 5. Comparing Pairs of Gold-Standard Summaries.

Target	bertscore fmeasure	rouge2 fmeasure
Wikipedia to NIST	0.553641	0.039266

Table 6. Mean ROUGE-2 Score By Event

Event	nist.f1	nist.precision	nist.recall	wiki.f1	wiki.precision	wiki.recall
CrisisFACTS-009	0.213318	0.239624	0.245875	0.033659	0.019623	0.242778
CrisisFACTS-010	0.149192	0.114315	0.286098	0.023204	0.012812	0.189320
CrisisFACTS-011	0.145257	0.100190	0.359034	–	–	–
CrisisFACTS-012	0.179170	0.179724	0.210144	–	–	–
CrisisFACTS-013	0.229716	0.224260	0.286118	0.019199	0.010242	0.243693
CrisisFACTS-014	0.234355	0.295634	0.220883	0.053600	0.032229	0.221145
CrisisFACTS-015	0.233751	0.233564	0.287291	0.020736	0.010894	0.328307
CrisisFACTS-016	0.182192	0.182008	0.208195	0.034347	0.019665	0.178250
CrisisFACTS-017	0.224246	0.266495	0.219846	0.018630	0.009963	0.233593
CrisisFACTS-018	0.228155	0.291395	0.211670	0.020362	0.010897	0.224767

Table 7. Mean BERTScores By Event

Event	nist.f1	nist.precision	nist.recall	wiki.f1	wiki.precision	wiki.recall
CrisisFACTS-009	0.589849	0.579551	0.601160	0.521315	0.497135	0.549008
CrisisFACTS-010	0.539738	0.537219	0.542816	0.507508	0.480131	0.538417
CrisisFACTS-011	0.584929	0.584297	0.585698	–	–	–
CrisisFACTS-012	0.541760	0.537036	0.546708	–	–	–
CrisisFACTS-013	0.608627	0.609467	0.608022	0.507653	0.474955	0.545453
CrisisFACTS-014	0.652125	0.658483	0.646350	0.497006	0.495349	0.498889
CrisisFACTS-015	0.591514	0.590219	0.593328	0.535723	0.497754	0.580440
CrisisFACTS-016	0.588953	0.587074	0.591479	0.498277	0.484401	0.513171
CrisisFACTS-017	0.576741	0.571791	0.581963	0.500104	0.452149	0.559761
CrisisFACTS-018	0.616166	0.619042	0.613623	0.518931	0.469754	0.580061

6 Comparing Assessment Methods

The manual, fact-based assessment described above is superior to the whole-event summarization evaluations in that we get better insight into missing content and performance across individuals day. That said, such an assessment is potentially limited in its reusability, as new runs–especially abstractive ones–are likely to produce new facts that were not present in the 2023 submission set. While this issue is lessened by 1) additional participation in CrisisFACTS and 2) our de-duplication approach, it remains a limitation in the reusability of the CrisisFACTS test collection.

More traditional document summarization approaches may be more amenable to such re-use, however, as they do not rely on explicit fact matching. As such, the whole-event summarization approach, while limited in its own ways, yields an additional avenue for evaluation. Ideally then, the rankings this summary-based method induces on the submitted runs should correlate strongly with those produced by the manual assessment. In Table 8, we show the correlations across evaluation methods and see that, indeed, the ranking produced by the BERTScore-based evaluation against NIST-produced summaries correlates strongly ($\tau > 0.6$) with that induced by the manual assessment.

Table 8. Kendall Tau Correlations Across Evaluation Metrics.

	rank.assessor	rank.nist.rouge	rank.wiki.rouge	rank.nist.bert	rank.wiki.bert
rank.assessor	1.000000	0.581699	0.215686	0.594771	0.620915
rank.nist.rouge	0.581699	1.000000	0.346405	0.647059	0.594771
rank.wiki.rouge	0.215686	0.346405	1.000000	0.437908	0.228758
rank.nist.bert	0.594771	0.647059	0.437908	1.000000	0.633987
rank.wiki.bert	0.620915	0.594771	0.228758	0.633987	1.000000

Given that the ranking between our fact-matching and BERTScore methods have a near-strong correlation, Figure 5 shows the pairs of performance metrics between these two metric types (redundancy compared to BERTScore-based precision and comprehensiveness compared to BERTScore-based recall). From this figure, we see that our redundancy metric has much wider variance than comprehensiveness, whereas all the BERTScore metrics have limited range. As most systems appear to score low in the comprehensiveness metric, one explanation for this finding is that the BERTScore threshold we use to de-duplicate facts is too high, leading to many systems producing their own unique sets of facts. More investigation is needed here. We also breakdown these results by summary type, as shown in Figure 6, where it is clear that abstractive systems dominate extractive ones across all three metrics.

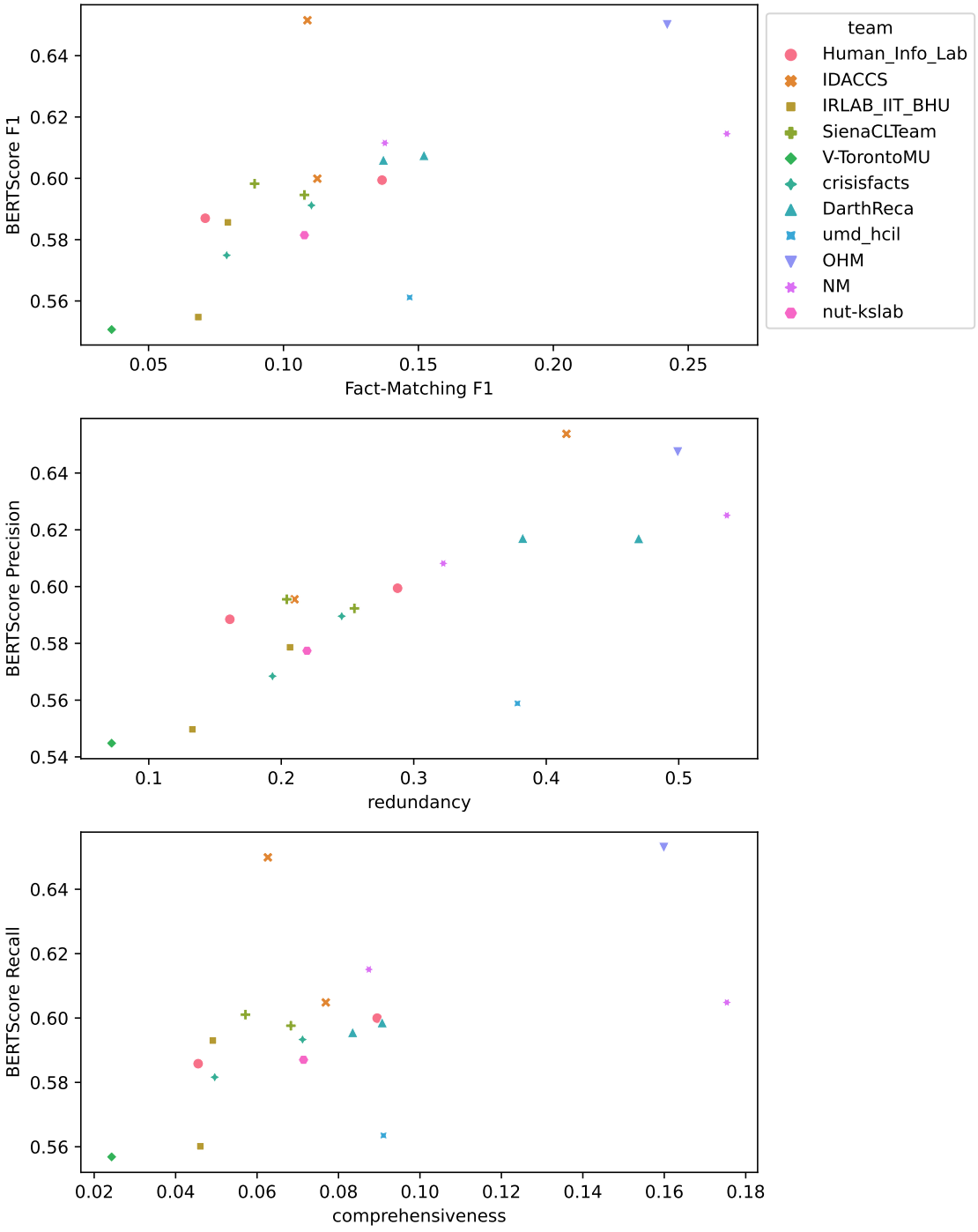

Figure 5. Comparing Fact-Matched and BERTScore Assessments per Team.

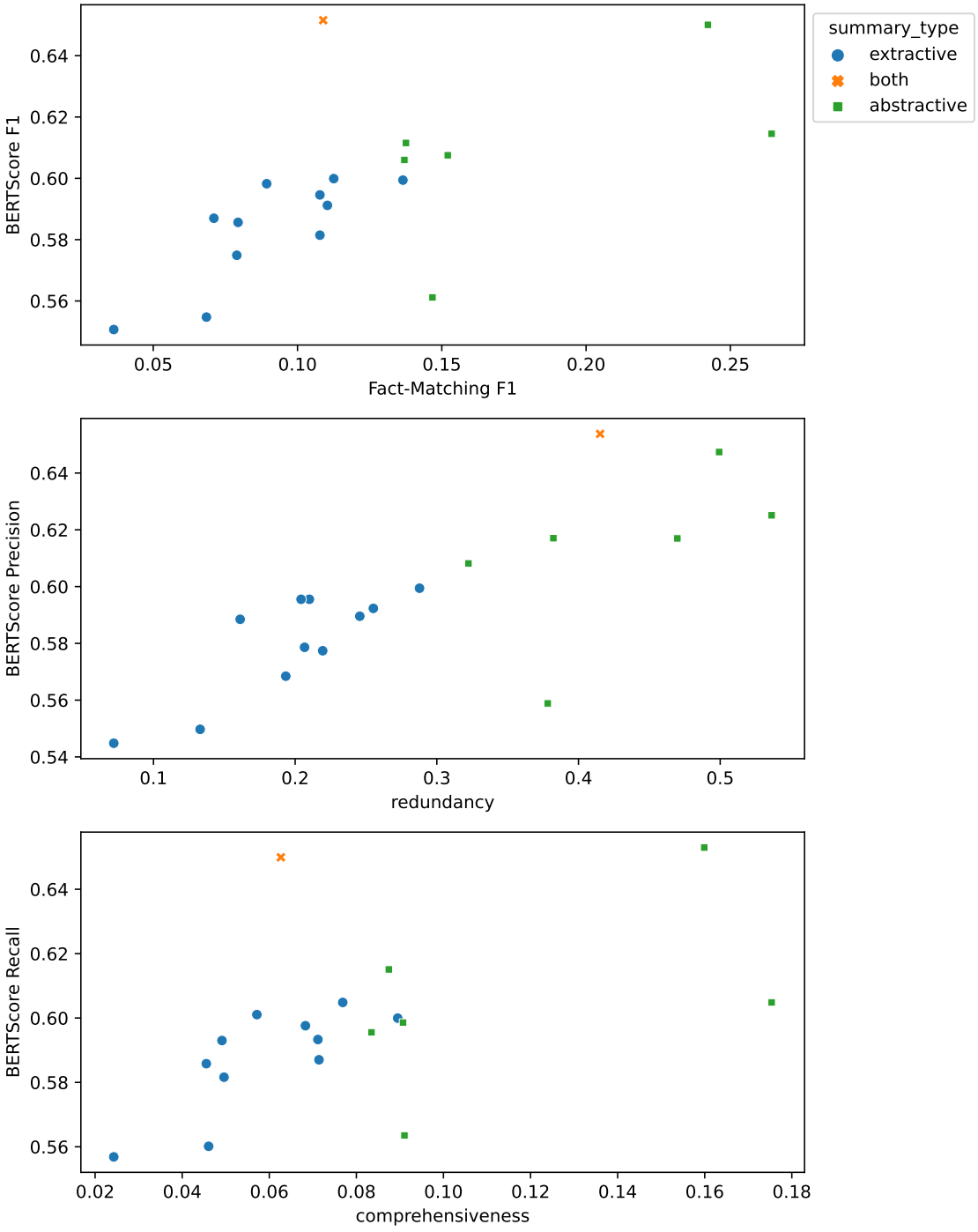

Figure 6. Comparing Fact-Matched and BERTScore Assessments by Summary Type.

7 References

Allan, J., Gupta, R., and Khandelwal, V. (2001). "Temporal summaries of new topics". In: *Proceedings of the 24th annual international ACM SIGIR conference on Research and development in information retrieval*, pp. 10–18.

McCreadie, R. and Buntain, C. (2023). "CrisisFACTS: Buidling and Evaluating Crisis Timelines". In: *Proceedings of the 20th International Conference on Information Systems for Crisis Response and Management (ISCRAM)*.

Zhang, T., Kishore, V., Wu, F., Weinberger, K. Q., and Artzi, Y. (2020). "BERTScore: Evaluating text generation with BERT". In: *ICLR*.

A Per-Run Scores

Table 9. Per-Run Average Scores, Using NIST-Labeled Facts and Wikipedia Summaries, Ordered by Fact-Matching F1.

		Fact-Matching NIST		BERTScore NIST			BERTScore Wikipedia		
Team	Redundancy	Comprehensiveness	F1	nist.f1	nist.precision	nist.recall	wiki.f1	wiki.precision	wiki.recall
NM:nm-gpt35	0.536325	0.175368	0.264311	0.614533	0.625103	0.604840	0.511478	0.484986	0.544210
OHM:llama_13b_chat	0.499265	0.159890	0.242211	0.650057	0.647403	0.652950	0.536347	0.509542	0.566918
DarthReca:drdqn-all	0.469766	0.090718	0.152069	0.607515	0.616971	0.598586	0.527017	0.501981	0.555754
umd_hcil:llama	0.378287	0.091042	0.146763	0.561154	0.558854	0.563516	0.535183	0.509500	0.564614
NM:gpt35-bm25	0.322320	0.087454	0.137579	0.611530	0.608162	0.615089	0.531794	0.503929	0.563953
DarthReca:drdqn-notopic	0.382261	0.083478	0.137031	0.605995	0.617046	0.595547	0.514654	0.489745	0.543785
Human_Info_Lab:FM-B	0.287780	0.089494	0.136530	0.599417	0.599429	0.599964	0.533170	0.510079	0.559507
IDACCS:occams_extract	0.210036	0.076873	0.112552	0.599904	0.595526	0.604842	0.519510	0.489406	0.554526
crisisfacts:baseline.v1	0.245629	0.071170	0.110363	0.591187	0.589564	0.593316	0.508443	0.475838	0.548294
IDACCS:occamsHybridGPT3.5	0.415243	0.062649	0.108872	0.651558	0.653789	0.649920	0.552321	0.528936	0.578634
SienaCLTeam:WikiTrigrams1	0.255198	0.068303	0.107763	0.594590	0.592328	0.597603	0.500634	0.471222	0.535462
nut-kslab:01	0.219438	0.071416	0.107761	0.581459	0.577393	0.587021	0.501040	0.462300	0.550485
SienaCLTeam:FactTrigrams1	0.204179	0.057154	0.089309	0.598239	0.595526	0.601060	0.514366	0.479710	0.556090
IRLAB_IIT_BHU:DFReeKLIM_1	0.206637	0.049137	0.079394	0.585638	0.578607	0.593001	0.495649	0.464279	0.533339
crisisfacts:baseline.v2	0.193387	0.049606	0.078959	0.574903	0.568440	0.581614	0.491146	0.460323	0.528307
Human_Info_Lab:FM-A	0.161194	0.045535	0.071011	0.586982	0.588476	0.585809	0.496209	0.473359	0.522602
IRLAB_IIT_BHU:BM25_1	0.132925	0.046075	0.068431	0.554739	0.549706	0.560143	0.482295	0.450480	0.520604
V-TorontoMU:USE_4	0.071966	0.024286	0.036317	0.550683	0.544823	0.556858	0.466081	0.432823	0.506116