Extending Multi-Document Summarization Evaluation to the Interactive Setting

Ori Shapira[1], **Ramakanth Pasunuru**[2], **Hadar Ronen**[3], **Mohit Bansal**[2], **Yael Amsterdamer**[1], **and Ido Dagan**[1]

[1]Bar-Ilan University [2]UNC Chapel Hill [3]Peres Academic Center
{obspp18, hadarg}@gmail.com
{ram, mbansal}@cs.unc.edu
{amstery, dagan}@cs.biu.ac.il

Abstract

Allowing users to interact with multi-document summarizers is a promising direction towards improving and customizing summary results. Different ideas for interactive summarization have been proposed in previous work but these solutions are highly divergent and incomparable. In this paper, we develop an end-to-end evaluation framework for interactive summarization, focusing on expansion-based interaction, which considers the accumulating information along a user session. Our framework includes a procedure of collecting real user sessions, as well as evaluation measures relying on summarization standards, but adapted to reflect interaction. All of our solutions and resources are available publicly as a benchmark, allowing comparison of future developments in interactive summarization, and spurring progress in its methodological evaluation. We demonstrate the use of our framework by evaluating and comparing baseline implementations that we developed for this purpose, which will serve as part of our benchmark. Our extensive experimentation and analysis motivate the proposed evaluation framework design and support its viability.

1 Introduction

Large bodies of texts on a topic oftentimes contain extensive information that is challenging for a potential reader to handle. Traditionally, information seeking tasks, like search, question-answering (QA) and multi-document summarization (MDS), are single-round input-output processes that can serve the information seeker only to a limited extent. This calls for an interactive setting where a user can guide the information gathering process. For search and QA, this type of research has been gaining momentum recently in areas such as exploratory search (Marchionini, 2006) and conversational QA (Reddy et al., 2019).

For MDS, where interaction would allow a user to affect summary content, only sporadic works have been seen over the years (e.g., Leuski et al., 2003; Lin et al., 2010; Yan et al., 2011; Baumel et al., 2014; Christensen et al., 2014; Shapira et al., 2017; Handler and O'Connor, 2017). A key gap in the development and adoption of *interactive summarization* (denoted here INTSUMM) solutions is the lack of evaluation methodologies and benchmarks for meaningful comparison of systems, similarly to those for static (non-interactive) summarization (e.g., NIST, 2014). The previous works on interactive or customizable summarization of multi-document sets are distinct, with proprietary evaluations that do not admit comparison. Furthermore, the evaluation processes are often not scalable and replicable, or do not give a comprehensive enough assessment.

In this paper we develop an end-to-end evaluation framework for INTSUMM systems. The framework starts with real user session collection on a system, via a concrete process of controlled crowdsourcing that we designed for this task. The sessions are then measured to produce absolute scores for the system, allowing for robust system comparison. Our framework supports a general notion of *expansion-based* interactive summarization, where the textual summary gradually expands in response to user interaction. Figure 1 presents an INTSUMM system that we implemented to illustrate this notion (§5.1). To ensure our evaluation framework is sound, we developed the framework in multiple cycles accompanied by user studies and extensive crowdsourcing experimentation. Our main contributions are as follows.

(1) *Evaluation measures.* We propose a set of automatic and manual evaluation measures for INTSUMM systems, which build upon a combination of established notions in static summarization and interactive systems, and enable utilizing available multi-document summarization (MDS)

Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, pages 657–677
June 6–11, 2021.

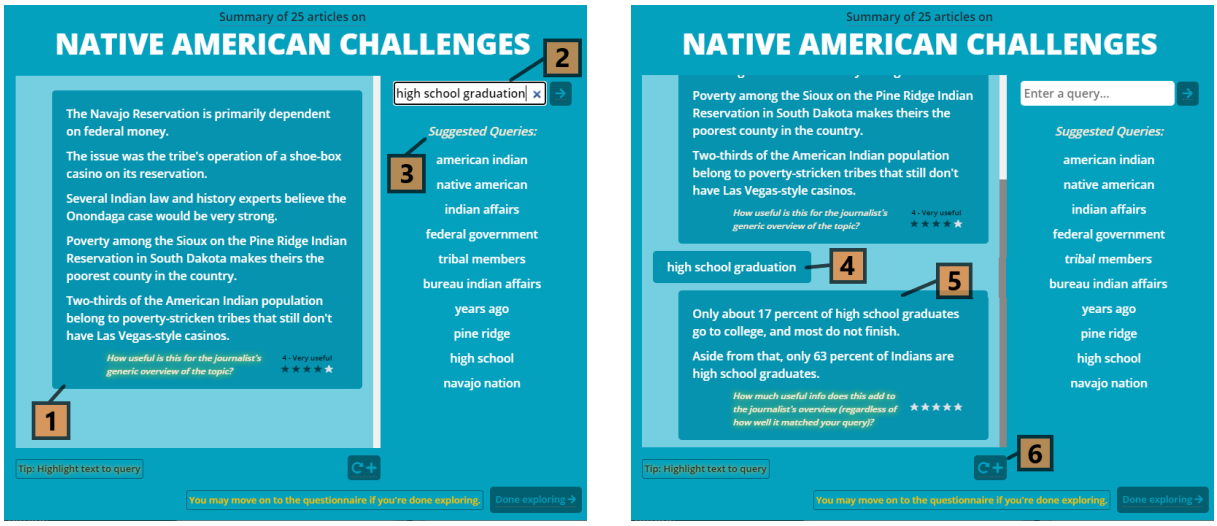

Figure 1: Our INTSUMM web application, implemented for testing our evaluation framework. The left screenshot shows the automatically generated initial summary for 25 articles on "Native American Challenges" (from the DUC 2006 MDS dataset), and a follow-up expansion query that the user might enter. The right screenshot shows the response to the query. [1] Initial summary; [2] box for entering free-text queries (or highlighted spans from the summary pane); [3] list of clickable system suggested queries; [4] last user query; [5] system response to the last query; [6] button to expand further on the last query. Subsequent queries and responses are continuously appended at the bottom of the summary pane, allowing exploration of the documents' content.

datasets. Our measures are aggregated over multiple interactive sessions and document sets to obtain an overall system evaluation. In contrast to static summarization, our measures apply to the steps along the interaction to reflect the progress of information acquirement rather than just its final result. This is done by converting an interactive session to a sequence of incrementally growing static summaries, and measuring the accumulating information gain with a recall metric. See §3.

(2) *Crowdsourced session collection process.* Adequate INTSUMM system evaluation and comparison requires collecting realistic user sessions in a consistent manner, on which the measurements are conducted. Previous work mostly turn to in-house user-studies, which are less replicable, not scalable, and not always easily attainable. In contrast, standard crowdsourcing induces noise and overly tolerates subjective behavior, hindering replicability and comparability. We describe a *controlled crowdsourcing* procedure that overcomes the above obstacles, making the evaluation process reliable and much more accessible for researchers interested in pursuing INTSUMM research. See §4.

We demonstrate the use of our full evaluation framework on two INTSUMM systems that we implemented, which apply different algorithms but share a common user interface, with the DUC 2006 (Dang, 2006) MDS dataset. Analysis shows favorable results in terms of internal consistency between sessions, users, and different evaluation measures, indicating that our solutions may serve as a promising benchmark for future INTSUMM research. See §5. The evaluation procedures and systems are available publicly.[1]

2 Background

Traditional MDS has been researched extensively (e.g. Goldstein et al., 2000b; Radev et al., 2004; Haghighi and Vanderwende, 2009; Yin and Pei, 2015). It encompasses variants of *query-focused summarization* (Dang, 2005), orienting the output summary around a given query (e.g. Daumé III and Marcu, 2006; Zhao et al., 2009; Cao et al., 2016; Feigenblat et al., 2017; Baumel et al., 2018), and *incremental update summarization* (Dang and Owczarzak, 2008), generating a summary of a document set with the assumption of prior knowledge on an earlier set (e.g. Li et al., 2008; Wang and Li, 2010; McCreadie et al., 2014; Zopf et al., 2016). Evaluation approaches predominantly include automatic ROUGE (Lin, 2004) measurement, i.e. word overlap against reference summaries, and manual responsiveness (Dang, 2006) scores or pairwise comparison (Zopf, 2018) between summaries.

[1] https://github.com/OriShapira/InterExp

In the related QA task (Voorhees et al., 1999), a system extracts an answer for a targeted question. Similarly, in the interactive setting, a conversational QA (Reddy et al., 2019) system extracts answers to a series of interconnected questions with a clear informational goal. To check correctness in both cases, a system answer is simply compared to the true answer via text-comparison. On the contrary, in the exploratory style of INTSUMM, where the knowledge desired is less certain, evaluation must consider dynamically accumulating information.

Exploratory search (Marchionini, 2006; White and Roth, 2009) addresses the need for converting big data to knowledge via human-machine cooperation. For example, interactive information retrieval (Ingwersen, 1992) focuses on fine-tuning document retrieval interactively, and complex-interactive-QA (ciQA) (Kelly and Lin, 2007) involves interacting with a system to generate a passage that answers a complex question. Evaluation is a major challenge in dealing with these tasks (White et al., 2008; Palagi et al., 2017; Hendahewa and Shah, 2017). Firstly, real users must use the system being evaluated by completing a task-appropriate assignment, requiring large-scale user studies that highly increase the cost and complexity of evaluation. Furthermore, varying user behavior could mean distorted session comparison. Then, a system is measured on the basis of its final outputs, mostly disregarding the evolvement of the interactional session.

Among interactive summarization, in the query-chain focused summarization task (Baumel et al., 2014), a chain of queries yields a sequence of short summaries, each refraining from repeating content. The task's evaluation relies solely on pre-defined sequences of queries with a respective reference summary per iteration (laboriously prepared by experts) that disregards previous outputs by the system. Other interactive summarization systems, such as Christensen et al. (2014); Shapira et al. (2017), present a *preassembled* summary with several levels of detail, allowing a user to drill down to or expand on information of interest. These works do not evaluate in a manner that is comparable to others, and do not consider information variation due to interaction. They perform small-scale user-studies for preference between their system and static variants, or a single automatic assessment of the fully expanded final summary.

We address all of the above-mentioned evaluation issues, specifically targeting the INTSUMM task, where the interaction-induced outputs are purely textual summary snippets of the input document set.

3 INTSUMM Evaluation

An INTSUMM system is evaluated by measuring its performance on multiple sessions produced as a result of human operation. The input of a session, Σ, is a set of documents, D, on which to explore. A session comprises an automatically generated initial summary, σ_0, and a sequence of user-posed requests, q_i, and corresponding output responses, r_i. The responses can be viewed as *expansions* of σ_0. Consequently, the overall interactive summary resulting from Σ defines a sequence of incrementally expanding *snapshots* $[\sigma_0, \sigma_1, \ldots, \sigma_{|\Sigma|}]$ where $\sigma_i = \sigma_0 \cup \bigcup_{j=1}^{i} r_j$ is the union of accumulative (summarized) information presented to the user after i interactions. Each snapshot may thus be regarded as a static summary, allowing static summarization measures to be applied on it.

For compared INTSUMM systems $S_1, ..., S_m$, we require at least u sessions of distinct users interacting with S_i on each test document set $D \in \{D_1, ..., D_n\}$. Assuming such sessions, we next define automatic and manual evaluation measures, and defer details on adequate session collection to §4. Importantly, all measures are based on established evaluation mechanisms used in static summarization and interactive systems, that we extend or adapt for the INTSUMM setting, and that are practically linear in time to the length of the session sequence. Together, the set of measures we define provide an encompassing assessment adequate for the evaluation of interactive summary systems.

3.1 Automatic Measures

Viewing a session as a sequence of incrementally expanding static summary snapshots, we would first like to obtain comparable scores for each static summary that will capture the information gained along the session up to the current interaction. Existing static MDS benchmarks provide reference summaries at a *single* length for the purpose of evaluating a summary at a *similar* length. This presumably means we would require a series of reference summaries that differ by small length gaps for the sequence of lengthening snapshots, which is difficult and costly to produce. To address this obstacle, we leverage a finding by Shapira et al. (2018) show-

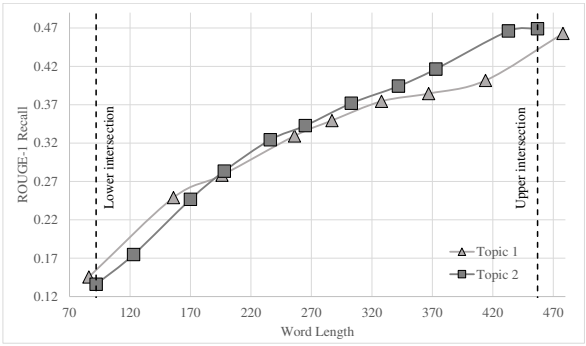

Figure 2: Example recall-curves of two sessions on an INTSUMM system. Points plotted per interaction snapshot within a session. Range of intersection between observed summary lengths is bounded by dashed lines.

ing that a reference summary of a single length can be used to relatively evaluate varying length summaries on a topic with a recall measure such as ROUGE. Thus, utilizing existing MDS datasets is indeed possible for measuring information gain throughout a session's snapshot sequence.

Based on this observation we now define three indicators for system performance, first over a single session and then aggregated over all sessions of a system.

Per-session indicators. (1) To illustrate the gradual information gain along a session we adopt a *recall-by-length curve* (Kelly and Lin, 2007; Lin, 2007), see for example Figure 2. The curve's x-axis is the snapshot word-length, chosen as the dominant factor affecting quality, as opposed to number of queries or interaction time, which are not necessarily comparable between sessions. The y-axis is a summary content recall score, such as ROUGE-recall against constant reference summaries.[2] For session Σ with snapshots $\sigma_0, \sigma_1, ..., \sigma_{|\Sigma|}$, each σ_i with word length x_i and content recall score y_i is plotted on the graph at (x_i, y_i).

(2) We consider the *area under the recall-curve* (AUC). Intuitively, it is desirable for an INTSUMM system to generate more salient information earlier: assuming salient information is more relevant to users, this property means interaction is ceased sooner, as soon as the information needs are met. Accordingly, AUC is higher when content is more relevant and is retrieved earlier. AUC is defined between start and end x-values, fixed for comparable measurement (see Figure 2), with y-value scores interpolated at these limits when a curve does not have a snapshot at the specific length(s).

(3) We consider the *Score@Length* metric that reports a score, such as standard ROUGE F_1, at pre-specified word-lengths, and demonstrates the informational effectiveness of a system at those lengths. This metric enables fair comparison to static summaries at the specified lengths. The inverse Length@Score measure is also examined, and detailed further in Appendix C.

Aggregated indicators. Our final system-level performance indicators are computed from the respective session indicators, as follows.

- The *average recall curve*, illustrating overall gradual information gain, is computed from individual session recall-at-length curves by interpolating y-values at constant x-value increments and averaging correspondingly. E.g., Figure 3.
- $[P.1]$ is the average AUC computed from the individual session AUCs by first averaging per topic and then averaging the results over all topics, to give equal weight to each topic.
- $[P.2]$ is the average Score@Length computed similarly to average AUC from individual session Score@Lengths.

3.2 Human Ratings

Automatic evaluation is convenient for fast assessment and consistent comparison, however manual appraisal more accurately forecasts the quality of a summarization system (Owczarzak et al., 2012). Thus, using manual metrics alongside automatic ones is important despite the higher cost it incurs.

Our evaluation framework allows doubly leveraging the involvement of human users by asking them to rate different system aspects during the session. We propose the following rating layout, with each measure being scored on a 1-to-5 scale.

- $[R.1]$ After reading the initial summary, the user rates how informative it is for the given topic. This resembles the DUC manual summary content responsiveness rating (Dang, 2006).
- $[R.2]$ To measure the information gain throughout the session, the user rates how much useful information each interaction's response adds. As this rating is scored per interaction, the session average measures overall ability to expose interesting information.
- $[R.3]$ After the session, the user rates how generally well the system responded to the requests

[2] Any standard summary content recall measure can be used as long as it is consistent, including, e.g., manual mechanisms like Pyramid or nugget-style scoring (Nenkova and Passonneau, 2004; Lin and Demner-Fushman, 2006).

throughout the session.

- $[R.4]$ As all human-involved systems should measure perceived usability, the user rates the two UMUX-Lite (Lewis et al., 2013) questionnaire statements: $[R.4a]$ the system's capabilities meet the requirements and $[R.4b]$ the system is easy to use. The UMUX-Lite score is a function of these two scores (although they are separately useful) and shows high correlation to the popular, and longer, SUS questionnaire (Brooke, 1996), thus offering a cheaper alternative.

Similarly to our automatic measures, these ratings are collected separately per session and then averaged, first per topic and then over all topics, to obtain comparable system scores.

The evident advantages of our proposed evaluation framework are: (1) scores are absolute and comparable from one session/system to another; (2) our framework fundamentally and conveniently extends upon prevailing static summarization evaluation practices and utilizes existing standard MDS dataset reference summaries.

4 Session Collection

The evaluation of interactive systems requires *real user sessions*, as explained in §3. Using a prototype INTSUMM system, described in §5.1, we conducted several cycles of session collection which uncovered multiple user-related challenges, in line with previous work on user task design (Christmann et al., 2019; Roit et al., 2020; Zuccon et al., 2013). In particular, recruited users may make undue interactions due to insincere or experimental behavior, yielding noisy sessions that do not reflect realistic system use. Additionally, without an objective informational goal, a user interacts with the system according to subjective interests, producing sessions that are objectively incomparable.

Controlled crowdsourcing method. Employing experts to use an interactive system in a user study is usually unnecessary and hinders scalability and accessibility for researchers, making crowdsourcing an appealing and less expensive alternative. While crowdsourcing is ordinarily used for annotation jobs, we show its suitability for system session collection. We designed a three-stage *controlled crowdsourcing* protocol that mitigates the aforementioned session collection challenges, while filtering out unsatisfactory workers (further details in Appendix B).

The first stage is a *trap task* whose aim is to efficiently filter out insincere workers, and, conversely, discover workers with an ability to apprehend salient information within text. The second stage assigns *practice tasks* that familiarize the workers to the INTSUMM system interface to prevent experimentation in the actual sessions to be evaluated. Here, the users are also presented with a *grounding use-case*, or 'cover-story' as termed by Borlund (2003). The use-case states an objective common goal to follow in interacting with the system, to minimize the effect of subjective preferences, and allow comparison against respective reference summaries with a similar objective goal. An example use-case to follow, applied in our experiments, is "produce an informative summary draft text which a journalist could use to best produce an overview of the topic". The use-case is strongly emphasized during practice sessions with integrated guidelines. Workers completing two practice assignments with predominantly relevant interactions are invited to continue on to the final stage.

The *evaluation session collection* stage involves interacting with the evaluated system, for a minimum amount of time per session (e.g., 150 seconds in our experiments), to produce a summary on a topic in light of the same assigned use-case as in the practice stage. Each worker may explore a topic once, and the overall goal is recording sufficiently many sessions per combination of system and topic. Generally in interactive tasks, systems are manually examined over a rather small number of instances (e.g. topics), with only a few users per instance, due to the high cost and complexity of collecting such sessions with experimenters. For example, Christensen et al. (2014) assessed their system on 10 topics, and the ciQA benchmark (Kelly and Lin, 2007) had 6 topics per tested subtask. Our session collection technique provides a more scalable approach, facilitating larger collection processes (e.g., in our experiments in §5 we used 20 topics and ≥ 3 sessions per system per topic).

We note that, in use cases or domains where experts *are* required, the proposed three-stage session collection protocol is still fully relevant. It is not limited to the crowdsourcing setting, and can be applied within controlled user-studies if needed.

Wild versus controlled crowdsourcing. We illustrate the benefit of the controlled crowdsourcing procedure described above by comparing its results with a "wild" crowdsourcing preliminary experiment. The latter experiment applied basic worker-

Measure	Controlled	Wild
# interactions	12.3	7.0
Approx. explore time	250 sec.	170 sec.
% suggested query	36.2%	62.7%
% free-text query	25.3%	2.2%
% ΔAUC from lower bound	+1.8%	−1.4%

Table 1: Qualitative measures of improved session collection through controlled crowdsourcing against sincere wild crowdsourcing. Values were computed per session and averaged over all sessions on System S_1 (see §5.1).

filtering (99% approval rate and 1000 approved assignments on Amazon Mechanical Turk[3] (AMT)) and did not apply the trap and practice tasks. For quality control, a post-session questionnaire was assigned in order to catch insincere workers.

Analysis of the collected sessions showed a substantial improvement in querying behavior in controlled crowdsourcing over *sincere* wild crowdsourcing (filtering out the insincere wild crowdworkers) – the former scored higher than the latter on every evaluation metric. Table 1 presents some qualitative indications of this improvement: controlled users were more engaged (more iterations and more time exploring) and put more thought into their queries (more free-text queries and less suggested queries). Notably, unlike uncontrolled crowd-workers, controlled workers were able to do better than a comparable fully automated baseline, evident from the last table row: the percent difference in ROUGE-1 AUC score from a "lower bound" simulated baseline (explained in §5.3), is positive (better) for controlled sessions and negative (worse) for "wild" ones. Finally, the queries of controlled users almost exclusively adhered to the use-case and the many helpful comments from the workers indicated their attentiveness to the task (see Appendix C).

5 Experiments

We carried out experiments that assess our full evaluation framework and demonstrate its utility. As the few existing INTSUMM systems were not readily available or suitable for adaptation to our experimental setup, we developed an INTSUMM system of our own, shown in Figure 1, with two different algorithmic implementations for comparison. We gathered user sessions with our controlled crowdsourcing procedure and evaluated their quality with our defined measures.

5.1 Test-System Implementations

We developed a web application, enabling session collection with real users, that follows the INTSUMM schema described in §3: for an input document set, it first presents an initial summary, and then iteratively outputs a summary expansion response per given user request. Specifically, our application supports interactive requests in the form of textual queries from user free-text, summary span highlights and system suggested queries. A system response aims to simultaneously maximize relevance to the query and salience, while refraining from repeating previously presented contents.

An initial version of the application was assessed via a small-scale user study of 10 users, with an SUS questionnaire (Brooke, 1996) and the think-aloud protocol (Lewis, 1982) for feedback. Figure 1 displays the improved web application, on the topic "Native American Challenges". The left screenshot shows the initial summary with user rating $[R.1]$ in [1], an example of a free-text query in the query box [2] and the list of suggested queries in [3]. The right screenshot shows the response to the query entered in the first screenshot. [4] reiterates the last submitted query, with the system response and user rating $[R.2]$ in [5]. The last query can also be repeated via a button [6], to obtain additional information on that query. Users can highlight a span from the presented summary, to be automatically pasted to the query box. Initial summaries and expansions are extractive and in bullet-style.

In accordance to this interaction flow presented, we implemented two back-end algorithm schemes, denoted S_1 and S_2, to demonstrate comparison of two INTSUMM systems via our evaluation framework. Each implementation consists of three components: (1) the initial summary generation, (2) the query-response generation and (3) extraction of suggested queries from the source documents. All system outputs must comply to required interaction latency standards (Anderson, 2020; Attig et al., 2017), e.g., a few seconds for the initial summary and a few hundred milliseconds for a query response. While we experimented with some more advanced techniques for MDS generation (e.g., Christensen et al., 2013; Yasunaga et al., 2017), sentence representation (Reimers and Gurevych, 2019) and sentence similarity (Zhang* et al., 2020), we found that these are not practical for incorpo-

[3] https://www.mturk.com

ration within the interactive low-latency setting, or that they could not handle the relatively large document set inputs. Instead, we developed the two back-end schemas described next (with further details in Appendix A).

S_1 runs a sentence clustering initial summary algorithm. Query-responses are generated in MMR-style (Goldstein et al., 2000a) based on semantic similarity between query and sentences. Suggested queries are frequent bigrams and trigrams. S_2 uses TextRank (Mihalcea and Tarau, 2004) for both the initial summary and suggested queries, and a query-response generation approach combining semantic and lexical similarity between query and sentences.

The two systems enable experimentation on our evaluation framework and, as we show, demonstrates its viability. Moreover, as apparent in our experimental results (§5.4 and user feedback in Appendix C), users attest to the real-world usefulness of these systems. Using our framework, including the baseline systems, future work can develop and examine more advanced methods for INTSUMM, accounting for the latency and input-size challenges.

5.2 Crowd Experimental Setup

Following our controlled session collection procedure from §4, we released the trap task in AMT and found 48 of 231 workers qualified for the second stage, out of which 25 accepted. 10 workers passed the training stage, from which we recruited 8 highly qualified ones. For the third stage, we collected sessions for 20 topics from DUC 2006, on S_1 and S_2. Each worker could explore 10 different topics on each system, amounting to 160 possible sessions of which 153 were completed (with at least 3 sessions per combination of topic and system). Since S_1 and S_2 share a common frontend application, users were unaware of which system they are exploring on, and the order was randomized. A minimum exploration time constraint of 150 seconds was set. Initial summaries were $\geq$ 75 tokens (average of 85) and responses were two sentences long.

The full controlled crowdsourcing process took one author-work-week, and cost \$370. In comparison, "wild" crowdsourcing described in §4 required a couple days' work and \$240 (achieving, as discussed, inferior results), and running a non-crowdsourced user-study of the same magnitude would likely require more work time, and cost an estimated \$480 (32 net hours of 16 workers at a commonly acceptable \$15 hourly wage). Furthermore, the results of a user study would not necessarily be of higher quality (Zuccon et al., 2013). To our judgement, the controlled crowdworkers are more suitable since they fathom the task *before* choosing to complete it. In a user study, workers are often unaware of the task before commencing, and may not be fully qualified for or desiring of it.

5.3 Simulated Bounds

In addition to real user experiments, we simulate each of our two systems on scripted query lists. Simulated sessions provide a means for quick development cycles and quality estimation.

The first of two query lists, L^{Sug}, is constructed fully automatically: it consists of the top-10 ordered phrases in the system's suggested queries component per topic. This mimics a "lower bound" user who adopts the simplest strategy, namely, clicking the suggested queries in order without using judgment even to choose among these queries.

The second list, L^{Oracle}, consists of 10 randomly chosen crowdsourced summary content units (SCUs) (Shapira et al., 2019) for each of the topics. Since the SCUs were extracted from the reference summaries of the corresponding topics, they mimic a user who searches for the exact information required to maximize similarity to the same reference summaries which we then evaluate against. While this is not necessarily the optimal query list due to the randomized sampling of SCUs for queries, we consider it our (non-strict) "upper bound" for the sake of experimentation.

The two "bounds" are relative to the system on which the simulations are carried on. Also, for fair comparison to real sessions, the simulation initial summary and response lengths are similarly set at $\geq$ 75 words and two sentences respectively.

5.4 Experimental Results

We next present the results attained on the 153 sessions collected (§5.2), with the purpose of analyzing our full evaluation framework. We gain an understanding on the consistency between automatic and human measurement, and on the comprehensiveness of the full set of measures.

Figure 3 presents the average recall-curves and corresponding $[P.1]$ averaged AUC scores of the S_1 bounds (§5.3) and of the user sessions on S_1 and S_2. AUC is computed between word-lengths 105 to 333 (the maximum intersection of all sessions). Table 2 shows $[P.2]$ averaged ROUGE-1

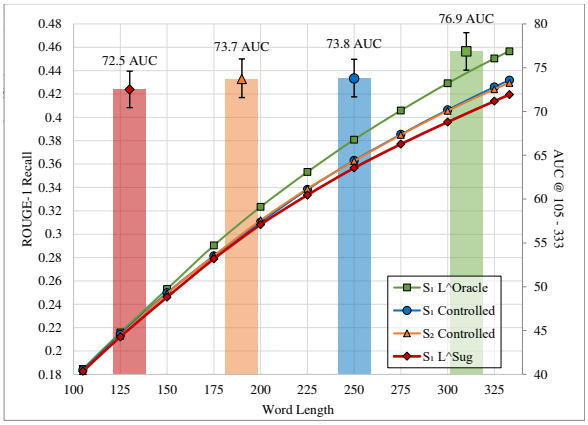

Figure 3: The average recall-curves, along with corresponding AUC scores (unrelated to the x-axis) and their confidence intervals ($\geq 95\%$), of the upper and lower bound sessions and of user sessions of the two systems.

Sessions	S@L 150	S@L 250	S@L 350
S_1 L^{Oracle}	.328 ($\pm$.012)	.400 ($\pm$.010)	.414 ($\pm$.012)
S_1 Real	.324 ($\pm$.010)	.382 ($\pm$.011)	.392 ($\pm$.012)
S_1 L^{Sug}	.319 ($\pm$.011)	.375 ($\pm$.012)	.382 ($\pm$.011)
S_2 L^{Oracle}	.333 ($\pm$.011)	.402 ($\pm$.013)	.412 ($\pm$.014)
S_2 Real	.321 ($\pm$.009)	.379 ($\pm$.013)	.388 ($\pm$.012)
S_2 L^{Sug}	.320 ($\pm$.011)	.374 ($\pm$.014)	.386 ($\pm$.013)

Table 2: ROUGE-1 F_1-based average Score@Length of simulated sessions vs. real user sessions. Scores at 350 words are approximate as few sessions were shorter. Scores rank consistently on ROUGE-2, -L and -SU. Intervals at $\geq 95\%$ confidence.

based Score@Length. Scores rank consistently on ROUGE-2, ROUGE-L and ROUGE-SU (see Appendix C).

It is evident from Figure 3 and Table 2 that the results on collected sessions indeed fall between the two bounds in all measures. This demonstrates the effectiveness of interactive summarization, even when using relatively simple algorithms: the algorithm enables fast information processing of input texts, and users effectively direct the algorithm to salient areas.

Additionally, the scores of S_1 and S_2 are close, providing no significant insights when comparing these two systems, which is surprising due to their distinct implementations. Manually reviewing the results, we were convinced that the systems indeed happen to perform at similar quality *overall*. However, when assessing the systems' separate components and inspecting user-provided ratings, we gain awareness of some interesting distinctions.

Table 3 shows a trend of consistency between ROUGE scores on each separate component and

	Metric	S_1	S_2
Initial	Initial summary ROUGE-1	**0.232**	0.225
	$[R.1]$ Initial summary rating	**3.89** (0.98)	3.71 (1.0)
Query	Que-Resp L^{Oracle} ROUGE-1	0.156	**0.161**
	$[R.2]$ Avg. response rating	3.17 (1.32)	**3.35** (1.28)
	$[R.3]$ Query responsiveness	3.61 (1.02)	**3.83** (1.03)
Overall	$[R.4a]$ System effectiveness	3.81 (1.0)	**4.05** (0.80)
	$[R.4a]$ System ease of use	4.51 (0.71)	**4.63** (0.62)
	$[R.4]$ System UMUX-Lite	74.2 (12.5)	**77.1** (10.3)

Table 3: Average and (StD) scores of metrics comparing S_1 and S_2. Users appear to be more satisfied with S_2 overall, likely due to the query response component.

the ratings provided by the users. The initial summaries' ROUGE-1 F_1 scores are computed against the reference summaries, with a slight advantage for S_1 over S_2 – similar to the users' initial summary ratings. For the query-response component, we compute the average ROUGE F_1 score of the independent responses to the queries in L^{Oracle}, against the reference summaries. Again, user ratings reflect a similar trend that the query-response component of S_2 slightly outscores that of S_1. Overall we see that S_1 provides a better initial summary while S_2 handles queries better. Also, users tend to be more satisfied by S_2, likely due to its ability to respond better to queries. This claim is evident from the positive correlation between $[R.3]$ and $[R.4a]$, $r = 0.68, p < 0.001$ in S_1 and $r = 0.63, p < 0.001$ in S_2. In terms of absolute UMUX-Lite scores $[R.4]$, 68 is considered average, and above 80 is considered excellent, meaning both S_1 and S_2 got high usability scores, with a preference for S_2.

An additional analysis finds a positive correlation between per-iteration response $[R.2]$ scores and the relative per-iteration increase in ROUGE recall (e.g. for ROUGE-1 $r = 0.36, p < 0.001$ in S_1 and $r = 0.33, p < 0.001$ in S_2), hinting at the credibility of correlation between human ratings and relative increase in ROUGE within sessions.

To conclude, our findings are favorable in terms of the framework's internal consistency of measures and soundness of the computed scores. For a more conclusive appraisal of the full evaluation framework, additional systems are to be run through the process, regardless of the accidental similarity between our two baselines.

6 Discussion and Conclusion

We proposed a comprehensive evaluation framework for user-guided expansion-based interactive

summarization – a vital ingredient for the methodological advancement of interactive summarization research which was unaccounted for until now. Our controlled crowdsourcing procedure makes INTSUMM system session collection accessible, scalable and replicable. The evaluation measures in our framework provide a thorough assessment with absolute scores that enable comparison of INTSUMM systems. Our framework provides the means to advance INTSUMM research on system development and improved evaluation. All solutions, including our implemented baseline systems, are publicly available to enable comparison of new INTSUMM systems to ours on any MDS dataset.

In future work, it is worthwhile to separately assess the effectiveness of individual interaction modes, including ones incorporated in our implementation and others, e.g., full questions input by users. These would require further experimentation, additional evaluation metrics, and the possible use of datasets from tasks other than MDS. Within our expansion-based framework, we can consider additional measures of textual consistency, coherence, and relevance of responses to queries. We may also test additional approaches for summarization: e.g., *abstractive* summarization for flexible synthetic summary generation, requiring further evaluation of factuality and truthfulness. Beyond our framework, that targets objective quality, INTSUMM systems should also be evaluated according to their compatibility with personalized, subjective use.

7 Ethical Considerations

User-study. Our system-testing user-study (mentioned in §5.1) was conducted on a university campus, and students within different age groups and from different backgrounds were recruited through a social media group for hiring for experiments and user studies. We required a high level of English for participation. People were accepted until the required amount of participants (10) was reached, without any targeted filtering. An individual study lasted around 30 minutes for a payment of around $10.

Crowdsourcing. There were several rounds of crowdsourcing, with varying tasks. Due to the need for fluent English speaking workers, a location filter was set on the AMT platform for English (as primary language) speaking countries. At least one of the authors tested each task before its release to estimate worst-case task completion duration. The payment was then set according to $9 per hour for the estimated required time. In practice, almost all tasks were completed in less than the time estimated, and payment was well above $9 per hour. Very few assignments were rejected in cases of clear insencereness (unreasonably fast submission or senseless behavior).

Dataset usage. As pointed out throughout the paper, the DUC 2006 dataset was utilized. It was obtained through the required means on the DUC website (duc.nist.gov). There was no possibility to reconstruct the dataset (document sets and reference summaries) within any of the conducted user study and crowdsourcing tasks.

Application. Our INTSUMM systems' outputs are extracts from the input document sets. As described in Appendix A, the algorithms for initial summary and query-response generation do not contain any intentional biasing.

The intended purpose of any INTSUMM system is to allow readers to make sense of large bodies of text through assisted exploration. *Future work* may open the door to more personalized algorithms and abstractive outputs. This would require extra care in making sure systems are ethically sound by adding targeted evaluation measures.

Compute time. As emphasized in the paper, INTSUMM systems require low latency and are hence relatively computationally cheap. During our research we ran some algorithms, to test for our systems, that required up to several hours of compute time per run, on a standard server.

Acknowledgments

We would like to thank Guiseppe Carenini for his helpful advice, and the anonymous reviewers for their constructive comments. This work was supported in part by the German Research Foundation through the German-Israeli Project Cooperation (DIP, grants DA 1600/1-1 and GU 798/17-1); by the BIU Center for Research in Applied Cryptography and Cyber Security in conjunction with the Israel National Cyber Bureau in the Prime Minister's Office; by the Israel Science Foundation (grants 1157/16 and 1951/17); by a grant from the Israel Ministry of Science and Technology; by the NSF-CAREER Award #1846185; and by a Microsoft PhD Fellowship.

References

Shaun Anderson. 2020. How fast should a website load? https://www.hobo-web.co.uk/your-website-design-should-load-in-4-seconds. Accessed: 2020-05-19.

Christiane Attig, Nadine Rauh, Thomas Franke, and Josef F Krems. 2017. System latency guidelines then and now–is zero latency really considered necessary? In *International Conference on Engineering Psychology and Cognitive Ergonomics*, pages 3–14. Springer.

Tal Baumel, Raphael Cohen, and Michael Elhadad. 2014. Query-chain focused summarization. In *Proceedings of the 52nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 913–922, Baltimore, Maryland. Association for Computational Linguistics.

Tal Baumel, Matan Eyal, and Michael Elhadad. 2018. Query focused abstractive summarization: Incorporating query relevance, multi-document coverage, and summary length constraints into seq2seq models. *arXiv preprint arXiv:1801.07704*.

Pia Borlund. 2003. The iir evaluation model: a framework for evaluation of interactive information retrieval systems. *Information research*, 8(3):8–3.

John Brooke. 1996. Sus-a quick and dirty usability scale. *Usability evaluation in industry*, 189(194):4–7.

Ziqiang Cao, Wenjie Li, Sujian Li, Furu Wei, and Yanran Li. 2016. Attsum: Joint learning of focusing and summarization with neural attention. In *Proceedings of COLING 2016, the 26th International Conference on Computational Linguistics: Technical Papers*, pages 547–556.

Janara Christensen, Mausam, Stephen Soderland, and Oren Etzioni. 2013. Towards coherent multi-document summarization. In *Proceedings of the 2013 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 1163–1173, Atlanta, Georgia. Association for Computational Linguistics.

Janara Christensen, Stephen Soderland, Gagan Bansal, and Mausam. 2014. Hierarchical summarization: Scaling up multi-document summarization. In *Proceedings of the 52nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 902–912, Baltimore, Maryland. Association for Computational Linguistics.

Philipp Christmann, Rishiraj Saha Roy, Abdalghani Abujabal, Jyotsna Singh, and Gerhard Weikum. 2019. Look before you hop: Conversational question answering over knowledge graphs using judicious context expansion. In *Proceedings of the 28th ACM International Conference on Information and Knowledge Management*, pages 729–738.

Hoa Trang Dang. 2005. Overview of duc 2005. In *Proceedings of the document understanding conference*, volume 2005, pages 1–12.

Hoa Trang Dang. 2006. Overview of duc 2006. In *Document Understanding Conference*.

Hoa Trang Dang and Karolina Owczarzak. 2008. Overview of the tac 2008 update summarization task. In *TAC*.

Hal Daumé III and Daniel Marcu. 2006. Bayesian query-focused summarization. In *Proceedings of the 21st International Conference on Computational Linguistics and the 44th annual meeting of the Association for Computational Linguistics*, pages 305–312. Association for Computational Linguistics.

Guy Feigenblat, Haggai Roitman, Odellia Boni, and David Konopnicki. 2017. Unsupervised query-focused multi-document summarization using the cross entropy method. In *Proceedings of the 40th International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 961–964.

Jade Goldstein, Vibhu Mittal, Jaime Carbonell, and Jamie Callan. 2000a. Creating and evaluating multi-document sentence extract summaries. In *Proceedings of the ninth international conference on Information and knowledge management*, pages 165–172.

Jade Goldstein, Vibhu Mittal, Jaime Carbonell, and Mark Kantrowitz. 2000b. Multi-document summarization by sentence extraction. In *Proceedings of the 2000 NAACL-ANLP Workshop on Automatic summarization*, pages 40–48. Association for Computational Linguistics.

Aria Haghighi and Lucy Vanderwende. 2009. Exploring content models for multi-document summarization. In *Proceedings of Human Language Technologies: The 2009 Annual Conference of the North American Chapter of the Association for Computational Linguistics*, pages 362–370.

Abram Handler and Brendan O'Connor. 2017. Rookie: A unique approach for exploring news archives. In *Proceedings of Data Science + Journalism workshop at KDD*, Halifax, Nova Scotia, Canada. Association for Computing Machinery.

Chathra Hendahewa and Chirag Shah. 2017. Evaluating user search trails in exploratory search tasks. *Information Processing & Management*, 53(4):905–922.

Kai Hong and Ani Nenkova. 2014. Improving the estimation of word importance for news multi-document summarization. In *Proceedings of the 14th Conference of the European Chapter of the Association for Computational Linguistics*, pages 712–721, Gothenburg, Sweden. Association for Computational Linguistics.

Peter Ingwersen. 1992. *Information retrieval interaction*, volume 246. Taylor Graham London.

Diane Kelly and Jimmy Lin. 2007. Overview of the trec 2006 ciqa task. In *ACM SIGIR Forum*, volume 41, pages 107–116. ACM New York, NY, USA.

Anton Leuski, Chin-Yew Lin, and Eduard Hovy. 2003. iNeATS: Interactive multi-document summarization. In *The Companion Volume to the Proceedings of 41st Annual Meeting of the Association for Computational Linguistics*, pages 125–128, Sapporo, Japan. Association for Computational Linguistics.

Vladimir I Levenshtein. 1966. Binary codes capable of correcting deletions, insertions, and reversals. In *Soviet physics doklady*, volume 10, pages 707–710.

Clayton Lewis. 1982. *Using the" thinking-aloud" method in cognitive interface design*. IBM TJ Watson Research Center Yorktown Heights, NY.

James R Lewis, Brian S Utesch, and Deborah E Maher. 2013. Umux-lite: when there's no time for the sus. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, pages 2099–2102.

Wenjie Li, Furu Wei, Qin Lu, and Yanxiang He. 2008. Pnr2: ranking sentences with positive and negative reinforcement for query-oriented update summarization. In *Proceedings of the 22nd international conference on computational linguistics (Coling 2008)*, pages 489–496.

Chin-Yew Lin. 2004. ROUGE: A package for automatic evaluation of summaries. In *Text Summarization Branches Out*, pages 74–81, Barcelona, Spain. Association for Computational Linguistics.

Jimmy Lin. 2007. Is question answering better than information retrieval? towards a task-based evaluation framework for question series. In *Human Language Technologies 2007: The Conference of the North American Chapter of the Association for Computational Linguistics; Proceedings of the Main Conference*, pages 212–219.

Jimmy Lin and Dina Demner-Fushman. 2006. Will pyramids built of nuggets topple over? In *Proceedings of the main conference on Human Language Technology Conference of the North American Chapter of the Association of Computational Linguistics*, pages 383–390. Association for Computational Linguistics.

Jimmy Lin, Nitin Madnani, and Bonnie Dorr. 2010. Putting the user in the loop: Interactive maximal marginal relevance for query-focused summarization. In *Human Language Technologies: The 2010 Annual Conference of the North American Chapter of the Association for Computational Linguistics*, pages 305–308, Los Angeles, California. Association for Computational Linguistics.

James MacQueen et al. 1967. Some methods for classification and analysis of multivariate observations. In *Proceedings of the fifth Berkeley symposium on mathematical statistics and probability*, volume 1, pages 281–297. Oakland, CA, USA.

Gary Marchionini. 2006. Exploratory search: from finding to understanding. *Communications of the ACM*, 49(4):41–46.

Richard McCreadie, Craig Macdonald, and Iadh Ounis. 2014. Incremental update summarization: Adaptive sentence selection based on prevalence and novelty. In *Proceedings of the 23rd ACM international conference on conference on information and knowledge management*, pages 301–310.

Rada Mihalcea and Paul Tarau. 2004. Textrank: Bringing order into text. In *Proceedings of the 2004 conference on empirical methods in natural language processing*, pages 404–411.

Tomas Mikolov, Ilya Sutskever, Kai Chen, Greg S Corrado, and Jeff Dean. 2013. Distributed representations of words and phrases and their compositionality. In *Advances in neural information processing systems*, pages 3111–3119.

Ani Nenkova and Rebecca Passonneau. 2004. Evaluating content selection in summarization: The pyramid method. In *Proceedings of the Human Language Technology Conference of the North American Chapter of the Association for Computational Linguistics: HLT-NAACL 2004*, pages 145–152, Boston, Massachusetts, USA. Association for Computational Linguistics.

NIST. 2014. Document understanding conferences. `https://duc.nist.gov/`. Accessed: 2020-05-19.

Karolina Owczarzak, John M. Conroy, Hoa Trang Dang, and Ani Nenkova. 2012. An assessment of the accuracy of automatic evaluation in summarization. In *Proceedings of Workshop on Evaluation Metrics and System Comparison for Automatic Summarization*, pages 1–9, Montréal, Canada. Association for Computational Linguistics.

Emilie Palagi, Fabien Gandon, Alain Giboin, and Raphaël Troncy. 2017. A survey of definitions and models of exploratory search. In *Proceedings of the 2017 ACM Workshop on Exploratory Search and Interactive Data Analytics*, pages 3–8.

Dragomir R Radev, Hongyan Jing, Małgorzata Styś, and Daniel Tam. 2004. Centroid-based summarization of multiple documents. *Information Processing & Management*, 40(6):919–938.

Siva Reddy, Danqi Chen, and Christopher D. Manning. 2019. CoQA: A conversational question answering challenge. *Transactions of the Association for Computational Linguistics*, 7:249–266.

Nils Reimers and Iryna Gurevych. 2019. Sentence-BERT: Sentence embeddings using Siamese BERT-networks. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 3982–3992, Hong Kong, China. Association for Computational Linguistics.

Paul Roit, Ayal Klein, Daniela Stepanov, Jonathan Mamou, Julian Michael, Gabriel Stanovsky, Luke Zettlemoyer, and Ido Dagan. 2020. Controlled crowdsourcing for high-quality qa-srl annotation. *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*. To Appear.

Gaetano Rossiello, Pierpaolo Basile, and Giovanni Semeraro. 2017. Centroid-based text summarization through compositionality of word embeddings. In *Proceedings of the MultiLing 2017 Workshop on Summarization and Summary Evaluation Across Source Types and Genres*, pages 12–21, Valencia, Spain. Association for Computational Linguistics.

Ori Shapira, David Gabay, Yang Gao, Hadar Ronen, Ramakanth Pasunuru, Mohit Bansal, Yael Amsterdamer, and Ido Dagan. 2019. Crowdsourcing lightweight pyramids for manual summary evaluation. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 682–687, Minneapolis, Minnesota. Association for Computational Linguistics.

Ori Shapira, David Gabay, Hadar Ronen, Judit Bar-Ilan, Yael Amsterdamer, Ani Nenkova, and Ido Dagan. 2018. Evaluating multiple system summary lengths: A case study. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 774–778, Brussels, Belgium. Association for Computational Linguistics.

Ori Shapira, Hadar Ronen, Meni Adler, Yael Amsterdamer, Judit Bar-Ilan, and Ido Dagan. 2017. Interactive abstractive summarization for event news tweets. In *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing: System Demonstrations*, pages 109–114, Copenhagen, Denmark. Association for Computational Linguistics.

Ellen M Voorhees et al. 1999. The trec-8 question answering track report. In *Trec*, volume 99, pages 77–82.

Dingding Wang and Tao Li. 2010. Document update summarization using incremental hierarchical clustering. In *Proceedings of the 19th ACM international conference on Information and knowledge management*, pages 279–288.

Patrick Wessa. 2020. Free statistics software, office for research development and education, version 1.2.1. https://www.wessa.net/. Accessed: 2020-05-19.

Ryen W White, Gary Marchionini, and Gheorghe Muresan. 2008. Evaluating exploratory search systems. *Information Processing and Management*, 44(2):433.

Ryen W White and Resa A Roth. 2009. *Exploratory search: Beyond the query-response paradigm*. 3. Morgan & Claypool Publishers.

Svante Wold, Kim Esbensen, and Paul Geladi. 1987. Principal component analysis. *Chemometrics and intelligent laboratory systems*, 2(1-3):37–52.

Rui Yan, Jian-Yun Nie, and Xiaoming Li. 2011. Summarize what you are interested in: An optimization framework for interactive personalized summarization. In *Proceedings of the 2011 Conference on Empirical Methods in Natural Language Processing*, pages 1342–1351, Edinburgh, Scotland, UK. Association for Computational Linguistics.

Michihiro Yasunaga, Rui Zhang, Kshitijh Meelu, Ayush Pareek, Krishnan Srinivasan, and Dragomir Radev. 2017. Graph-based neural multi-document summarization. In *Proceedings of the 21st Conference on Computational Natural Language Learning (CoNLL 2017)*, pages 452–462, Vancouver, Canada. Association for Computational Linguistics.

Wenpeng Yin and Yulong Pei. 2015. Optimizing sentence modeling and selection for document summarization. In *Twenty-Fourth International Joint Conference on Artificial Intelligence*.

Tianyi Zhang*, Varsha Kishore*, Felix Wu*, Kilian Q. Weinberger, and Yoav Artzi. 2020. Bertscore: Evaluating text generation with bert. In *International Conference on Learning Representations*.

Lin Zhao, Lide Wu, and Xuanjing Huang. 2009. Using query expansion in graph-based approach for query-focused multi-document summarization. *Information processing & management*, 45(1):35–41.

Markus Zopf. 2018. Estimating summary quality with pairwise preferences. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long Papers)*, pages 1687–1696, New Orleans, Louisiana. Association for Computational Linguistics.

Markus Zopf, Eneldo Loza Mencía, and Johannes Fürnkranz. 2016. Sequential clustering and contextual importance measures for incremental update summarization. In *Proceedings of COLING 2016, the 26th International Conference on Computational Linguistics: Technical Papers*, pages 1071–1082.

Guido Zuccon, Teerapong Leelanupab, Stewart Whiting, Emine Yilmaz, Joemon M Jose, and Leif Azzopardi. 2013. Crowdsourcing interactions: using crowdsourcing for evaluating interactive information retrieval systems. *Information retrieval*, 16(2):267–305.

A INTSUMM System Implementation Details

According to the INTSUMM schema described, we implemented two back-end baseline systems, sharing a front-end application. The user interactions supported are textual queries and the responses aim to maximize relevance to the query, similarly to query-focused static summarization, while refraining from repeating previously presented contents, somewhat similarly to update summarization. Specifically, the implementations support queries from free-text, highlights and system suggestions. Both systems output extractive bullet-style initial summaries and summary expansion responses.

A.1 Algorithm Variants

Each of our back-end INTSUMM implementations consists of three main components, as follows.

A.1.1 Initial Summary

We first consider the generation of the initial summary σ_0. Our experimentation with some classic *and* modern MDS implementations have indicated that most do not meet the interactivity response-time requirements, and hence we provide two implementations for this component based on standard extractive MDS methods.

The first algorithm denoted I^{CL}, is clustering-based with ideas inspired by Rossiello et al. (2017) and Hong and Nenkova (2014). All sentences in the document set are separately assigned a representation by averaging the 300-dimensional word2vec (w2v) embeddings (Mikolov et al., 2013) within each sentence. For this we use the SpaCy library.[4] Each vector's dimension is reduced to 20 with PCA (Wold et al., 1987), and then sentence vectors are clustered to 30 components with k-means (MacQueen et al., 1967). PCA and k-means are implemented with sklearn.[5] We then order clusters by size, and select a representing sentence starting from the largest cluster, and continuing to following clusters, until the summary word-limit has reached. A cluster is skipped if its representing sentence is too similar (cosine similarity of 0.95) to previously selected sentences. The sentence selected to represent a cluster is the one whose words are on average most frequent within the full document set. On a standard server, the implementation can generate a summary of 25 news articles from a common MDS dataset in a few seconds (2 to 10 seconds).

Hyper-parameters: For reduced sentence-representative vector dimension (20) we tested several values between 10 and 100. For number of sentence clusters (30) we tested values between 10 and 50, and for similarity threshold (0.95) we tested several options within the 0 to 1 range. The hyperparameters were lightly adjusted by computing ROUGE scores of some outputs against reference summaries, ensuring fast processing, and rational eyeballing.

We also experimented with Sentence-BERT (Reimers and Gurevych, 2019) representations, drawing on the 'roberta-base-nli-stsb-mean-tokens' model[6] in place of the word2vec-based ones, however this did not improve output summaries, and, more importantly, had high latency (tens of seconds to a few minutes).

The second algorithm is TextRank (Mihalcea and Tarau, 2004), denoted I^{TR}, coded with the pytextrank pipeline component in the SpaCy library.[7] The implementation supports an argument with which we can limit the initial summary word-length. It is slightly slower than I^{CL}, but runs within approximately the same time range.

A.1.2 Query Response Generation

This component computes similarity of a given query to all sentences not yet presented in previous iterations, and outputs a few of the best matches. Process-time is up to a few hundred milliseconds.

The first variant Q^{SEM}, utilizes the semantic (w2v-based) sentence representations prepared in the initialization process (either averaged word2vec representation or Sentence-BERT). It computes the cosine similarity between the query's representation and all unused source sentences. In MMR-style (Goldstein et al., 2000a), the sentences most similar to the query which pass a dissimilarity threshold from already selected sentences are selected. An MMR dissimilarity of ≤ 0.05 gave the best results (on simulated sessions).

The second variant Q^{LEX}, additionally considers lexical similarity by scoring the similarity of a query and a sentence as the product of the

[4] https://spacy.io/usage/vectors-similarity

[5] https://scikit-learn.org

[6] https://github.com/UKPLab/sentence-transformers

[7] https://spacy.io/universe/project/spacy-pytextrank

w2v similarity with three ROUGE-precision scores. I.e., for query q and sentence s, $\mathrm{sim}(q,s) = (\mathrm{cosine}(w2v(q), w2v(s)) + 1) * (\mathrm{R1}_p(q,s) + 1) * (\mathrm{R2}_p(q,s) + 1) * (\mathrm{RL}_p(q,s) + 1)$, where $\mathrm{R1}_p$, $\mathrm{R2}_p$ and RL_p are ROUGE-1, ROUGE-2 and ROUGE-L precision respectively. The highest scoring unused sentences are output. Compared to the first variant, this method yields lexically-stricter search results.

Finally, we also experimented with BERTScore (Zhang* et al., 2020) as the similarity score between the query and each of the relevant sentences. Here too, the high latency makes this approach impractical for our purposes.

A.1.3 Suggested Queries

Our systems support an interaction mode enabling information expansion by clicking a system-suggested query from a list.

The first approach for preparing the list, $\mathrm{Sug}^{\mathrm{FREQ}}$, is selecting the most frequent bigrams and trigrams within the source documents, disregarding stop words. A trigram is preferred when it contains a bigram with the same frequency. An n-gram with a Levenshtein distance (Levenshtein, 1966) of less than 2 from an already selected n-gram is skipped.

The second approach $\mathrm{Sug}^{\mathrm{TR}}$, utilizes the top-ranked phrases extracted from the TextRank algorithm (as part of the graph-based extractive summarization procedure).

A.1.4 Overall Systems

For the purpose of applying our evaluation framework, we picked two combinations of the above three components. (Assessing additional combinations is out of the scope of this paper.) The first combination, denoted System S_1, is comprised of I^{CL}, Q^{SEM} and $\mathrm{Sug}^{\mathrm{FREQ}}$. The second combination, System S_2, consists of I^{TR}, Q^{LEX} and $\mathrm{Sug}^{\mathrm{TR}}$.

A.2 Web Application

The front-end web application, which communicates with an INTSUMM system in the back-end (in our case S_1 or S_2), is seen in Figure 1 in the main paper. Some further details regarding the application:

A previous version of the web application included a button for *additional general information*, which was originally supported by System S_1. This sent an empty query to the system, for which the sentence from the next unused cluster in the I^{CL} algorithm was returned (rotating to the first cluster if all clusters have been used, and taking the next best unused representing sentence). In our preliminary user study and crowdsourcing experiments, we found that this feature was mainly a distraction and induced exploration laziness.

The Web application is implemented in HTML, CSS and Javascript, and the backend in Python. The app communicates with the backend over standard HTTP Post requests in JSON format.

A.3 Server Specifications

We ran experiments, and run our INTSUMM systems on an Intel Xeon CPU E5-2670 v3 @ 2.30GHz server with 50GB RAM running CentOS Linux 7. Run times are similar on an Intel Core i7-6600 CPU @ 2.60GHz laptop with 16GB RAM running Windows 10. We noticed some differences in the I^{CL} algorithm's outputs when run on the different hardware (consistent within but diverging between), even though the software environments were identical. This is likely due to a different 'random' implementations in the two settings.

To check if the query-response component with BERTScore would be more practical on a GPU server, we tested it on an Nvidia TITAN X GPU server. A single query response took 40-50 seconds to compute.

B Controlled Crowdsourcing Details

The controlled crowdsourcing protocol finds high quality users for the collection of system sessions. The *use-case* we enforced was producing an informative summary draft text which a journalist could use to best produce an overview of the given topic for the general public. This use case attempts to follow the informational goal of the reference summaries, which are practically generically written.

B.1 Trap Task

This task, consisting of three questions, aims to discover workers with an ability to apprehend salient information within text. It was implemented standardly within the Amazon Mechanical Turk[8] platform.

Question 1. This question tests apprehension of the notion of a general summary, by asking the user to choose a sentence that would be best to

[8] https://www.mturk.com

include in an overview of some topic, given the topic name and four relevant sentences of varying informativeness.

For our journalistic scenario, we randomly selected 10 topics from the DUC[9] 2007 MDS dataset that include Pyramid SCUs (Nenkova and Passonneau, 2004), and for each topic manually chose one SCU with a weight of 4, and three SCUs with a weight of 1. The SCU with the higher weight would be expected to be the more appropriate choice for the journalistic overview, since all four reference summaries of the topic include it.

Question 2. This question simulates a scenario closer to interactive summarization. A three-sentence "initial summary" of the same topic (as Question 1) is presented, and the worker is asked to choose the best of three possible interaction possibilities that would provide more information on the topic from a theoretical search tool. We presented three pairs of queries of varying relevance and informativeness as interaction possibilities.

The "initial summary" is the lead-three sentences of one of the reference summaries of the topic. We prepared the three choices of two queries based on salient phrases manually found within the reference summaries. One pair of queries is worthy, one pair is somewhat worthy, and a third pair is unworthy.

Question 3. This question tests attentiveness to the task and creativity by asking users to suggest another interaction (query) to the theoretical search tool. Such an open-ended question requires more thought and hence filters careless or guessing workers.

Task preparation. We ran the 10 tasks internally with research colleagues to find vulnerabilities, and edited some questions accordingly. In addition we recorded the average work time to set a fair task payment on the crowdsourcing platform. In retrospect, the time it took crowd-workers was about two-thirds of the time it took internal workers. Moreover, the time was an additional indication of better workers – those completing the task correctly in shorter time were likely superior.

We paid \$0.50 for each trap task assignment, estimating about 3.5 minutes of work time. Good workers completed the task in 2.5 minutes on average, which should fairly pay only \$0.30.

[9] https://duc.nist.gov/

Task assessment. The first two questions are automatic filters for insincere workers. A meaningful answer to the third question, assessed manually, serves as a sanity check which we found useful for additional filtering.

The workers passing this phase were contacted via email. The message included an explanation and estimated payment of the subsequent tasks.

B.2 Practice Task

This task is an external question done within an IFrame on Mechanical Turk.

Two practice tasks were prepared from DUC 2006, separate from the 20 used for real session collection. Workers completing both tasks with predominantly relevant queries (checked manually) were asked to continue on to the final task.

We emphasized the use case of preparing a journalistic overview by instructing to "produce an informative summary draft text which a journalist could use to best produce an overview of the topic".

We paid \$0.90 for each practice task, estimating 6-7 minutes of work per assignment. Our estimate was about right.

B.3 Evaluation Session Collection Task

As before, this external question task is done within an IFrame on Mechanical Turk.

For the session collection tasks we paid \$0.70 per topic, estimating 5 minutes of work. We promised to give a bonus for good work, to motivate completion of more assignments, and in higher standards. We awarded \$0.15 to \$0.30 bonus according to the quality (assessed manually), per assignment. All sessions were of very high quality, but some made an extra effort and provided comments and feedback.

B.4 Wording of Human Ratings

For our journalistic use-case, the ratings within a session are worded as follows:

- $[R.1]$ "How useful is this for the journalist's generic overview of the topic?"
- $[R.2]$ "How much useful info does this add to the journalist's overview (regardless of how well it matched your query)?"
- $[R.3]$ "During the interactive stage, how well did the responses respond to your queries?"
- $[R.4]$ "As a system for exploring information on a topic,

– $[R.4a]$ "its capabilities meet the need to efficiently collect useful information for a journalistic overview."
– $[R.4b]$ "it is easy to use."

B.5 Wild Crowdsourcing

For quality control, at the end of a session the user filled a questionnaire, in which they mark whether 10 statements are covered in their generated session. Of those statements, five were (separately) crowdsourced summary content units (SCUs) from the topic's reference summaries (Shapira et al., 2019), one of those SCUs was repeated to test for identical markings, two statements were SCUs from another topic, and two statements were the two shortest sentences output by the session. We thus know the answers to 4 statements and have a repeating statement test. Sessions with minimal mistakes could hypothetically be considered sincere.

C Experiments

C.1 Data

Systems were evaluated using data from the DUC 2006 MDS dataset. 20 topics were used (all those with Pyramid (Nenkova and Passonneau, 2004) evaluations). These are: D0601, D0603, D0605, D0608, D0614, D0615, D0616, D0617, D0620, D0624, D0627, D0628, D0629, D0630, D0631, D0640, D0643, D0645, D0647, D0650. The practice tasks in the controlled crowdsourcing procedure used topics D0602, D0606. The 10 topics in the trap task are based on document sets with Pyramid evaluations from DUC 2007. These are: D0701A, D0703A, D0704A, D0705A, D0706B, D0707B, D0710C, D0711C, D0714D, D0716D.

C.2 More Results

Similar to the results on ROUGE-1 in the main paper, Tables 4, 5 and 6 are for metrics ROUGE-2, ROUGE-L and ROUGE-SU respectively.

Sessions	S@L 150	S@L 250	S@L 350
S_1 L^{Oracle}	.063 (±.011)	.085 (±.013)	.096 (±.013)
S_1 Real	.064 (±.010)	.077 (±.010)	.082 (±.010)
S_1 L^{Sug}	.065 (±.009)	.078 (±.010)	.082 (±.012)
S_2 L^{Oracle}	.067 (±.011)	.085 (±.013)	.094 (±.013)
S_2 Real	.058 (±.010)	.072 (±.011)	.077 (±.013)
S_2 L^{Sug}	.056 (±.010)	.068 (±.011)	.073 (±.011)

Table 4: ROUGE-2 F_1-based average scores of simulated sessions vs. controlled crowdsourced sessions. Scores at 350 words are approximate as a few sessions were shorter. Intervals at $\geq 95\%$ confidence.

Sessions	S@L 150	S@L 250	S@L 350
S_1 L^{Oracle}	.270 (±.013)	.328 (±.012)	.333 (±.014)
S_1 Real	.271 (±.010)	.314 (±.010)	.319 (±.010)
S_1 L^{Sug}	.258 (±.010)	.299 (±.011)	.302 (±.011)
S_2 L^{Oracle}	.275 (±.012)	.327 (±.015)	.332 (±.015)
S_2 Real	.270 (±.011)	.313 (±.014)	.315 (±.014)
S_2 L^{Sug}	.271 (±.011)	.311 (±.014)	.313 (±.013)

Table 5: ROUGE-L F_1-based average scores of simulated sessions vs. controlled crowdsourced sessions. Scores at 350 words are approximate as a few sessions were shorter. Intervals at $\geq 95\%$ confidence.

Sessions	S@L 150	S@L 250	S@L 350
S_1 L^{Oracle}	.091 (±.008)	.145 (±.008)	.156 (±.011)
S_1 Real	.090 (±.007)	.137 (±.009)	.145 (±.009)
S_1 L^{Sug}	.089 (±.007)	.133 (±.009)	.139 (±.008)
S_2 L^{Oracle}	.093 (±.008)	.145 (±.010)	.156 (±.013)
S_2 Real	.090 (±.006)	.137 (±.010)	.141 (±.011)
S_2 L^{Sug}	.090 (±.007)	.133 (±.012)	.140 (±.011)

Table 6: ROUGE-SU F_1-based average scores of simulated sessions vs. controlled crowdsourced sessions. Scores at 350 words are approximate as a few sessions were shorter. Intervals at $\geq 95\%$ confidence.

C.3 Length@Score Metric

Sessions	R1 .37	R2 .075	RL .31	RSU .14
S_1 L^{Oracle}	193	191	199	232
S_1 Real	218	233	233	266
S_1 L^{Sug}	231	200	253	N/A
S_2 L^{Oracle}	192	180	197	232
S_2 Real	221	288	237	269
S_2 L^{Sug}	236	N/A	245	310

Table 7: The Length@Score measurements for ROUGE-1, ROUGE-2, ROUGE-L and ROUGE-SU F_1 scores. This answers how many words on average are needed to reach the specified ROUGE F_1 score. Values are calculated from the averaged overall session of a system, and not as a macro-average. When a value is 'N/A', the system did not reach the score.

We also computed a "Length@Score" measurement assessing at what word length, on average, a given content score can be reached by the system. It might forecast how much interaction is required to reach a certain information coverage, indicating system effectiveness. This is computed on the overall averaged session, and not as a macro-average, since some topics do not reach the specified score. By looking at the averaged session, we have a better overlook at the system's capability. Table 7 presents these values. Here, the ROUGE scores for which to compute the resulting lengths are chosen based on numbers within a range of scores found in

MDS literature employing the DUC 2006 dataset on extractive summarization systems. We see an overall similar trend where the controlled user sessions fall between the upper and lower bounds, and that S_1 scores slightly better than S_2. As an analysis, we can compare, e.g., the upper and lower bounds on ROUGE-1, and observe that the lower bound requires about 40 words more (20%) to contain a similar amount of salient content.

C.4 Standard Metric Implementations

Confidence intervals. All confidence intervals were calculated with a Python bootstrapping library,[10] and sometimes validated with an online tool (Wessa, 2020). The confidence level is $\geq 95\%$ throughout the paper.

ROUGE. ROUGE scores were obtained either with the rouge Python package[11] for quick sentence comparisons, or the pyrouge Python package[12] for file comparison (long text to reference summaries).

UMUX-Lite When on a 5-point scale, the UMUX-Lite (Lewis et al., 2013) score is computed:

$$0.65 * (([R.4a] + [R.4b] - 2) * (100/8)) + 22.9$$

68 is considered average, and above 80 is excellent.

C.5 Real Session Examples

Tables 8 and 9 present real sessions on Systems S_1 and S_2 respectively, both on the topic "El Niño". The two users here have different querying strategies for acquiring information. Their strategies are not consistent across topics, as they depend on the complexity and breadth of the topics.

C.6 Simulation Samples

Table 10 shows the lists of queries in the simulations used for upper and lower bounds, for the topic "El Niño".

C.7 Feedback from Users

The session collection task also had a comment box to send any kind of feedback. Table 11 shows a few of these comments (some shortened for brevity). The comments strongly emphasize the users' sincerity in following the use-case, and note usefulness of the system as well as provide ideas for improvements. Some comments attest to the need for prolonged exploration and human assistance for finding salient information.

[10] https://pypi.org/project/bootstrapped/

[11] https://pypi.org/project/rouge/

[12] https://pypi.org/project/pyrouge/

Query	Response
[I]	"The weather pattern comes on the heels of an El Nino that dissipated in 1998. El Nino is a natural phenomenon that involves a surface warming of the eastern and central Pacific Ocean around the equator. For the United States, an El Nino brings unusually warm temperatures to the northern states and cooler, wetter conditions to the South. By the time the next El Nino arrives, scientists hope to be able to give much better forecasts."
"El Nino" [H]	"La Ninas sometimes occur between El Ninos. Will La Nina wane as El Nino lingers?" he said."
"Pacific Ocean" [H]	"Those conditions are warming in the eastern Pacific Ocean; shifting of precipitation from the western to the eastern Pacific; slowing of trade winds; and changes in sea level. El Ninos increase storm activity in California and South America while calming things down in the Atlantic and Caribbean."
"equator" [H]	"Scientists at NASA's Jet Propulsion Laboratory came the conclusion after analyzing the data collected from satellite last week. The weather-altering La Nina phenomenon has all but vanished from the Pacific Ocean, according to NASA satellite observations."
"temperatures" [H]	"Sea surface temperatures throughout the equatorial east-central Pacific increased during April and May, when temperatures normally decrease in this region. He said the center predicted that after rapid cooling during the summer, ocean temperatures would rebound before cooling again."
"forecast" [H]	"It disrupts weather worldwide, causing storms, floods and droughts. It is blamed for flooding and heavy snow in the West, abnormally warm weather in the East and extreme weather from South America to Asia."
"la nina" [S]	"La Nina and El Nino form the opposite ends of the same climatic cycle, with El Nino fueled by unusually warm water in the eastern Pacific and La Nina driven by cold. A La Nina can drop it nearly 8 degrees."
"weather patterns" [S]	"El Nino causes reversals of normal weather patterns, resulting in drought in usually wet locales and flooding in arid areas. He said that soil moisture and snow cover appear perceptibly to influence seasonal weather patterns, adding that computer models at Scripps are beginning to incorporate some of these variables with some success."
"equatorial pacific" [S]	"Wetter conditions than usual will continue over the central and eastern equatorial Pacific and over southeastern South America. La Nina refers to the phenomenon of rapidly cooling equatorial waters in the central Pacific."
"weather phenomenon" [S]	"El Nino phenomenon, which comes at an interval of roughly four years, refers to abnormal temperature rises in tropical waters which normally leads to global anomalies in weather and climate. El Nino is a periodic warming of the water in the eastern Pacific Ocean, which shifts weather patterns arriving from that area and can affect the climate worldwide."
"surface temperatures" [S]	"During August and September, ocean surface temperatures reached near-record levels in many sections of the equatorial Pacific. The pool of unusually cool tropical water marked by low sea levels has shriveled and the equatorial Pacific is warming to normal temperatures, NASA's Jet Propulsion Laboratory said this week."
"global warming" [S]	"Despite causing anomaly of weather and climate, the El Nino event may help cut global warming by temporarily stemming the release of carbon dioxide from the Pacific Ocean, scientists suggested Thursday. The program said experiments on the relationship between increasing carbon dioxide levels and global warming being done at the Mauna Loa Observatory in Hawaii supported the Australian claims."
"carbon dioxide" [S]	"The oceans exchange large amounts of carbon dioxide with the atmosphere each year. Scientists usually estimate how much carbon dioxide should be in the atmosphere by calculating how much fossil fuel is burned."

Table 8: A real session in System S_1 on the topic "El Niño". [I] stands for 'initial summary', [H] stands for 'highlight' and [S] stands for 'suggested query'. The user here starts out with highlighted queries, and then uses suggestions.

Query	Response
[I]	"The good news is that they can now forecast an El Nino with some precision, and during El Nino years predict its effect on the world's weather months in advance. The conclusion was made by Song Jiaxi and his colleagues with the National Marine Environmental Forecasting Center in their annual marine disaster forecast report, which was released today. Despite causing anomaly of weather and climate, the El Nino event may help cut global warming by temporarily stemming the release of carbon dioxide from the Pacific Ocean, scientists suggested Thursday."
"climate change" [F]	"In general, a large proportion of infectious disease agents are very sensitive to slight changes in climate,s̈aid McMichael, who was not involved in the research. Conditions like El Nino might settle in almost permanently if global warming gets bad enough, making climate disruptions such as droughts or excessive winter rain essentially the norm, a computer study suggests."
[R]	"El Nino is a periodic warming of the water in the eastern Pacific Ocean, which shifts weather patterns arriving from that area and can affect the climate worldwide. Scientists cautioned that like its warm counterpart, El Nino, a La Nina condition will influence global climate and weather until it has completely subsided."
[R]	"WMO added that uncertainty over surface temperatures in the Atlantic and Indian Oceans, which contribute to changing rainfall over Africa and South America, makes an accurate forecast difficult. El Nino phenomenon, which comes at an interval of roughly four years, refers to abnormal temperature rises in tropical waters which normally leads to global anomalies in weather and climate."
"countries affected" [F]	"Diarrhea kills as many as 3 million children under the age of 5 worldwide every year and sickens millions more, mostly in developing countries. The phenomenon had been responsible for only 40 percent rainfall in the country in June, he said."
[R]	"When the present levels of the greenhouse gas carbon dioxide were doubled in the experiment, the number of El Ninos affecting Australia nearly doubled too, the scientist said. La Ninas, by contrast, reduce storms in California but stir up trouble in other parts of the country as well as in India and southeast Asia."
"la nina" [F]	"La Ninas sometimes occur between El Ninos. Will La Nina wane as El Nino lingers?" he said."
[R]	"La Nina and El Nino form the opposite ends of the same climatic cycle, with El Nino fueled by unusually warm water in the eastern Pacific and La Nina driven by cold. A La Nina can drop it nearly 8 degrees."
[R]	"If La Nina dissipates before it hits Los Angeles, the area could face a more typical wet winter. La Nina, Spanish for "little girl," is just the opposite, with the warm conditions of El Nino returning to the west."
"global warming" [F]	"The program said experiments on the relationship between increasing carbon dioxide levels and global warming being done at the Mauna Loa Observatory in Hawaii supported the Australian claims. Australian scientists have uncovered a link between global warming and the increasing frequency of the El Nino weather system, the Australian Broadcasting Corporation (ABC) reported tonight."
[R]	"Gerald Meehl of the National Center for Atmospheric Research in Boulder, Colo., agreed it will take more study to understand how global warming might affect El Nino. The potential effect of global warming on disease is controversial,s̈aid William Checkley of Baltimore-based Johns Hopkins, who led the study."
[R]	"The world was also warmer in that quiet El Nino period, further pointing to a possible relationship between El Nino and global warming. The computer simulation suggests that unrestrained global warming could set up the same kind of pattern in ocean surface temperatures, but as a more or less constant condition."
[R]	"The Four Corners TV program claimed scientific data were showing that a man-made permanent drought was in the offing as a result of global warming. El Nino is the name given to the unusual warming of waters in the Pacific Ocean that affects global wind and temperature patterns."
[R]	"The pool of unusually cool tropical water marked by low sea levels has shriveled and the equatorial Pacific is warming to normal temperatures, NASA's Jet Propulsion Laboratory said this week. El Nino is a natural phenomenon that involves a surface warming of the eastern and central Pacific Ocean around the equator."
[R]	"El Nino is the name given to the unusual warming of waters in the Pacific Ocean that affects wind and temperature patterns. BEIJING, May 18 (Xinhua) – Chinese oceanographers today expect climatic repercussions on global weather as El Nino is expected to come sometime this year or the next."
"oceans" [F]	"It is estimated that the equatorial Pacific Ocean is the source of 72 percent of the annual flux of carbon to the atmosphere from the equatorial oceans. Those conditions are warming in the eastern Pacific Ocean; shifting of precipitation from the western to the eastern Pacific; slowing of trade winds; and changes in sea level."
[R]	"The oceans exchange large amounts of carbon dioxide with the atmosphere each year. During August and September, ocean surface temperatures reached near-record levels in many sections of the equatorial Pacific."
[R]	"The most important of gas-releasing areas is the equatorial Pacific, where is also the home of El Nino, the quasi-biennial warming event in the surface waters of the eastern equatorial Pacific Ocean. Oklahoma's catastrophic tornadoes were influenced by La Nina, the weather phenomenon 5,000 miles west of the prairie in the Pacific Ocean, scientists say."

Table 9: A real session in System S_2 on the topic "El Niño". [I] stands for 'initial summary', [F] stands for 'free-text' and [R] stands for 'repeat last query'. The user's strategy here seems to be repeating a query until enough information is covered on the matter.

L^{Oracle}	
[I] "El Nino can cause storms in California, tornadoes in Florida, a mild winter in the northern states." "Scientists in the United States, Australia, Israel, and Germany are using sophisticated computer simulations." "La Nina works in reverse of El Nino." "La Nina is the phenomenon of rapidly cooling equatorial waters in the central Pacific." "Computer modeling a simulation are used to study El Nino and El Nina patterns." "El Nino typically lasts a year." "Scientific technologies and techniques for studying these phenomena include computer modeling." "El Nino may lessen global warming by temporarily stemming the release of carbon dioxide from the Pacific." "Computer module studies and satellite systems allow for a better understanding of how El Nino and La Nina form." "The results of El Nino and El Nina can have severe economic impacts, disease and death."	
L^{Sug}	
Sug^{FREQ}	Sug^{TR}
[I] "el nino" "la nina" "pacific ocean" "carbon dioxide" "weather patterns" "equatorial pacific" "south america" "global warming" "weather phenomenon" "surface temperatures"	[I] "el nino years" "el nino phenomenon" "el nino events" "el nino activity" "next el nino" "el nino behavior" "el nino update" "el ninos" "global weather" "normal weather patterns"

Table 10: The query lists for the topic "El Niño" used in the simulations for the upper (L^{Oracle}) and lower (L^{Sug}) bounds. Sug^{FREQ} is used in System S_1 and Sug^{TR} is used in System S_2. [I] stands for "initial summary".

	Topic	Comment
S_1	School Safety	"A...mix of information...from...statistics...to facts about... specific incidents, probably because it's such a large topic."
S_1	EgyptAir Crash	"Searching for "time" didn't give any kind of date or actual time of crash, however, searching for "date" tended to give actual date and time together..."
S_1	EgyptAir Crash	"I noticed the search engine returned flexible dates this time (I searched 1991 and got 1996 results, for example) and I really appreciated that."
S_1	Osteoarthritis	"There wasn't much...when trying to find specifics like symptoms..."Treatment" pulled up the closest and most relevant results, but the others went into the weeds or pulled up things that were tangential to the search terms."
S_1	Evolution Teaching	"...This is the first of these where I thought the system really did not meet the need to efficiently collect useful info for a journalistic overview."
S_1	Stephen Lawrence Killing	"I was very satisfied with the information provided...a topic with which I was totally unfamiliar. More generally...the system consistently did quite well...if I were a journalist writing overviews of these topics...I would be very pleased with...the information provided by the system [and] its ease of use!..."
S_1	Quebec Separatist Movement	"I think anything that requires a higher level of background knowledge is a lot harder to research with this system, since you only get snippets."
S_2	Wetlands	"I was able to find information on many different aspects of the topic."
S_2	EgyptAir Crash	"...I [tried to] find out if there were other crashes...which did not turn up any info, but then later found that information when looking up a different search term."
S_2	Concorde Aircraft	"The search results don't always seem to correspond to the terms keyed in..."
S_2	Concorde Aircraft	"It seems like I'm always looking for more general info...Things I would want included in an overview or even in an article dealing with a specific incident such as in this case..."
S_2	Elian Gonzales	"Most of the responses matched pretty well with the keyword search..."
S_2	Elian Gonzales	"I find that I'm using the system the same way I use Google; whatever I'm wondering about I just ask in the form of a question."
S_2	US Affordable Housing	"This set was very responsive and got results that I had not expected..."
S_2	Kursk Submarine	"Outstanding! I feel like I could write an overview of this right now!"
S_2	Jimmy Carter International	"In this one, the topic...was so general that it took me a bit to figure out exactly what I was supposed to be looking for. Once I got it, everything worked fine!"
S_2	El Niño	"Great! Tons of useful information for a journalistic overview!"

Table 11: Some of many comments provided by the controlled crowdsourcing users for the two systems S_1 and S_2 on different topics (some shortened for brevity). The comments indicate that users follow the use-case. Notice that some comments show the need for prolonged exploration and human assistance for finding salient information.