Towards Reliable and Factual Response Generation: Detecting Unanswerable Questions in Information-Seeking Conversations

Weronika Łajewska and Krisztian Balog

University of Stavanger, Stavanger, Norway,
{weronika.lajewska, krisztian.balog}@uis.no

Abstract. Generative AI models face the challenge of hallucinations that can undermine users' trust in such systems. We propose to approach the problem of conversational information seeking as a two-step process, where relevant passages in a corpus are identified first and then summarized into a final system response. This way we can automatically assess if the answer to the user's question is present in the corpus. Specifically, our proposed method employs a sentence-level classifier to detect if the answer is present, then aggregates these predictions on the passage level, and eventually across the top-ranked passages to arrive at a final answerability estimate. For training and evaluation, we develop a dataset based on the TREC CAsT benchmark that includes answerability labels on the sentence, passage, and ranking levels. We demonstrate that our proposed method represents a strong baseline and outperforms a state-of-the-art LLM on the answerability prediction task.

Keywords: Conversational search; Conversational response generation; Unanswerability

1 Introduction

Conversational information seeking (CIS) systems allow users to fulfill their complex information needs via a sequence of interactions. This problem is often approached as a passage retrieval task [5, 13], rather than employing generative AI techniques, to allow for the grounding of responses in supporting documents and to avoid hallucinations. However, the ultimate goal is to return informative, concise, and reliable answers instead of top-ranked passages. In an ideal scenario, when the passages from the top of the ranking answer the question, the task boils down to summarization [14]. However, it is often the case that the answer to the user's question is not contained in the top retrieved passages. In such cases, summaries generated from those passages would result in hallucinations [3, 21].

In this paper, we make the first step towards reliable and factual conversational response generation. We propose a mechanism for detecting unanswerable questions for which the correct answer is not present in the corpus or could not be retrieved. More specifically, given a set of top-ranked passages that have been

identified as most relevant to the given question, we predict if the question can be answered (at least partially) based on information contained in those passages. This enables us to move the notion of passage relevance and focus more on the actual presence of the information that answers the question.

Unanswerability detection has been addressed in the context of machine reading comprehension [9, 10] and question answering [4, 17, 18, 20], both of which differ significantly from the conversational search setup. Information-seeking dialogues pose additional challenges, such as open-ended questions requiring descriptive answers, indirect answers requiring an inference or background/context knowledge, and complex queries with partial answers spread across passages. Therefore, unanswerability detection is a novel, still unsolved task in CIS and, to the best of our knowledge, no public dataset exists for this problem.

As our first main contribution, we develop a dataset, based on the TREC CAsT benchmark, to train and evaluate methods for question answerability prediction. Utilizing an existing resource of snippet-level answer annotations, our dataset provides answerability labels on three levels: (1) sentences, (2) paragraphs, and (3) rankings (i.e., top-ranked paragraphs retrieved by a CIS system). Notably, we generate input passage rankings with various quality, mixing relevant and non-relevant results in a controlled way, which correspond to different degrees of difficulty in answerability prediction, ranging from all passages containing an answer to "no answer found in the corpus."

As our second main contribution, we provide a baseline approach for predicting answerability based on the top retrieved results. Our proposed approach predicts which sentences from top-ranked passages contribute to the answer and aggregates the obtained answerability scores on the passage and ranking levels. We demonstrate that this simple method provides a strong baseline, one that outperforms ChatGPT on the same task. Our benchmark dataset as well as the implementation of our proposed method are made publicly available at `https://anonymous.4open.science/r/answerability-prediction-0105/`.

2 Related Work

Research on information-seeking conversations is largely driven by collections released at the TREC Conversational Assistance Track (CAsT) [5–7, 14]. Unlike generative AI approaches, answers in this benchmark are grounded in paragraphs, hence the problem boils down to that of conversational passage retrieval [13, 16, 22]. Aggregating results from top-ranked paragraphs into a single answer is an open problem [2] that has been first piloted in the 2022 edition, where a subtask of generating summaries from retrieved results was introduced [14]. Ren et al. [19] propose an approach for response generation divided into three stages: (optional) query rewriting, finding supporting sentences in results displayed on a SERP, and summarizing them into a short conversational response. While the authors acknowledge the problem of unanswerability in conversational search, they do not address it in their proposed approach. In this paper, we aim to fill that gap.

Table 1. Statistics for the CAsT-answerability dataset.

	Answerable?	
	Yes	**No**
#question-sentence pairs (train+test)	6,395	19,043
#question-passage pairs (train+test)	1,778	1,932
#question-ranking pairs (test)	4,035	504

The problem of unanswerability has been addressed in the context of machine reading comprehension (MRC) [9, 10] and extractive question-answering (QA) [1, 8, 12]. Solutions proposed include answerability prediction using prompt-tuning [12], modeling high-level semantic relationships between objects from question and context [10], and combining the output of reading and verification modules in MRC systems [9, 23]. Our proposed solution is based on a sentence-level classifier that is learned on CIS-specific training data, and can further be augmented with QA answerability data.

3 Dataset

This paper builds upon an existing dataset, referred to as *CAsT-snippets*,[1] which extends the TREC CAsT'20 and '22 datasets with snippet-level annotations for the top-retrieved results. Specifically, it contains annotations of information nuggets defined as "minimal, atomic units of relevant information" [15], representing key pieces of information required to answer the given question. Snippets in the dataset are identified for every question in the 5 most relevant passages according to ground truth judgments. To balance the collection, we also include 5 randomly selected non-relevant passages to each question. The resulting dataset, named *CAsT-answerability*, contains around 1.8k answerable and 1.9k unanswerable question-passage pairs. We further consider answerability on the level of sentences and on the level of rankings, as follows. For sentence-level answerability, we leverage annotations of information nuggets from the CAsT-snippets dataset as follows: each sentence that overlaps with an information nugget, as per annotations in the originating CAsT-snippets dataset, is labeled as 1 (answer in the sentence), otherwise as 0 (no answer in the sentence).

For ranking-level answerability, which is the ultimate task we are addressing, we consider different input rankings, i.e., sets of $n = 3$ passages, for the same input question. Specifically, for each unique input test question (38), we generate all possible n-element subsets of passages available for this question (both containing and not containing an answer), thereby simulating passage rankings of varying quality. These rankings represent inputs with various degrees of difficulty for the same question, ranging from all passages containing an answer to a single passage with an answer to "no answer found in the corpus." This yields a total of 4.5k question-ranking pairs, of which 0.5k are unanswerable.

[1] Reference and link to the GitHub repository are removed to preserve anonymity; it will be added upon acceptance.

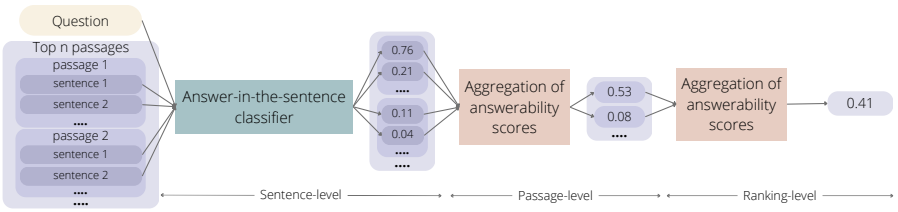

Fig. 1. Overview of our answerability detection approach.

Overall, our CAsT-answerability dataset contains binary answerability labels on three levels: sentence, passage, and ranking. Sentence- and passage-level answerability is divided into train (90%), and test (10%) portions; the splitting is done on the question level to avoid information leakage. Ranking-level answerability has only a test set. See Table 1 for a summary.

4 Answerability Detection

The challenge of answerability in CIS arises from the fact that the answer is typically not confined to a single entity or text snippet, but rather spans across multiple sentences or even multiple paragraphs. Note that answerability extends beyond the general notion of relevance and asks for the presence of a specific answer. At the core of our approach is a sentence-level classifier that can distinguish sentences that contribute to the answer from ones that do not. These sentence-level estimates are then aggregated on the passage level and then further on the ranking level (i.e., set of top-n passages) to determine whether the question is answerable; see Figure 1. Operating on the sentence level is a design decision that has the added benefit that a future summary generation step may take a filtered set of sentences that contribute to the final answer as input.

4.1 Answer-in-the-Sentence Classifier

The answer-in-the-sentence classifier is trained on sentence-level data from the train portion of the CAsT-answerability dataset. In some of the experiments, this data is augmented by data from the SQuAD 2.0 dataset [17] to provide the classifier with additional training material and thus guidance in terms of questions that can be answered with a short snippet contained in a single sentence. Data from SQuAD 2.0 is downsampled to be balanced in terms of the number of answerable and unanswerable question-sentence pairs. The classifier is built using a BERT transformer model with a sequence classification head on top (BertForSequenceClassification provided by HuggingFace[2]). Each data sample contains `question [SEP] sentence` as input and a binary answerability label. The output of the classifier is the probability that the sentence contains (part of) the answer to the question.

[2] `https://huggingface.co/docs/transformers/model_doc/bert#transformers.BertForSequenceClassification`

Table 2. Answerability detection results in terms of classification accuracy. Best scores for each level are in boldface. For the augmented classifier (rows 5–8), significant differences against the respective method (rows 1–4) are indicated by *. ChatGPT results are tested against the best classifier in rows 1–8. We use McNemar's test with $p < 0.05$.

Classifier	**Sentence**	**Passage**		**Ranking**	
	Acc.	**Aggr.**	**Acc.**	**Aggr.**	**Acc.**
CAsT-answerability	0.752	Max	0.634	Max	0.790
				Mean	**0.891**
		Mean	0.589	Max	0.332
				Mean	0.829
CAsT-answerability augmented with SQuAD 2.0	**0.779***	Max	0.676*	Max	0.810*
				Mean	0.848*
		Mean	0.639*	Max	0.468*
				Mean	0.672*
ChatGPT passage-level (zero-shot)			**0.787***	T=0.33	0.839*
				T=0.66	0.623*
ChatGPT ranking-level (zero-shot)					0.669*
ChatGPT ranking-level (two-shot)					0.601*

4.2 Aggregation of Sentence-level Answerability Scores

In reality, answers are not confined to a single sentence but can be spread across several passages. We thus need a method to aggregate results obtained from the sentence-level classifier to decide whether the question can be answered given (1) a particular passage or (2) a set of top-ranked passages, referred to as a ranking.

We consider two simple aggregation functions, *max* and *mean*, noting that more advanced score- and/or content-based fusion techniques could also be applied in the future [11]. Intuitively, *max* is expected to work particularly well for factoid questions where the answer is relatively short and usually contained in a single sentence, while *mean* should capture the cases where pieces of the answer are spread over several sentences within the passage or across passages. The aggregated answerability score for a given passage is compared against a fixed threshold. We set the threshold values on a validation partition (10% of the dataset, sampled from the training parition); 0.5 for max and 0.25 for mean.

An analogous procedure is repeated for the top $n = 3$ passages in the ranking to decide on ranking-level answerability. Here, the aggregation methods take the passage-level answerability scores as input (obtained using max or mean aggregation of sentence-level probabilities). The resulting values are compared against the fixed threshold (using the same values as for passage-level aggregation) to yield a final ranking-level answerability prediction.

5 Results

Table 2 presents the answerability results on the sentence-, passage-, and ranking-levels on the test partition of CAsT-answerability in terms of accuracy.

Does data augmentation help answerability detection? On the sentence level, we find that augmenting the CAsT-answerability dataset with additional train-

ing examples from SQuAD 2.0 improves performance. These improvements also carry over to the first aggregation step on the passage level. However, the best ranking-level results are obtained by aggregating results obtained from the classifier trained only on CAsT-answerability.

Which of the two aggregation methods performs better? In all cases, max aggregation on the passage level followed by mean aggregation on the ranking level gives the best results. Intuitively, this configuration captures single sentences with high answerability scores in individual passages (max aggregation on passage level) that give a high average score for the whole ranking (mean aggregation on ranking level) for answerable questions.

How competitive are these baselines in absolute terms? For reference, we compare against a state-of-the-art large language model (LLM), using the most recent snapshot of GPT-3.5 (gpt-3.5-turbo-0301) via the ChatGPT API. We consider two settings: given a passage (analogous to the passage-level setup) and given a set of passages as input (analogous to the ranking-level setup). We prompt the model to verify whether the question is answerable in the provided passage(s) and return 0 or 1 accordingly.[3] In the passage-level setup, the passage-level predictions returned by ChatGPT are aggregated using fixed thresholds to obtain a ranking-level prediction (0.33 or 0.66, based on the fact that binary values are returned for passage-level answerability predictions). In the ranking-level setup, we experiment with both a zero-shot setting and a two-shot setting (one positive and one negative example). We observe that the passage-level answerability scores of ChatGPT are higher than ours, but after ranking-level aggregation, it is no longer the case. Further, performing the ranking-level task directly results in significantly lower performance. These results indicate that LLMs have a limited ability to detect answerability without additional guidance.

6 Conclusion

Unanswerable questions pose a challenge in conversational information seeking. To study this problem, we have developed a test collection, based on two editions of the TREC CAsT benchmark, with sentence-, passage-, and ranking-level answerability labels. We have also presented a baseline approach based on the idea of sentence-level answerability classification and multi-step results aggregation, and evaluated multiple instantiations of this approach with different configurations. Despite their simplicity, our baselines have been shown to outperform a state-of-the-art LLM on the task of answerability prediction.

In this paper, we have simplified the scenario by treating answerability as binary: a question is answerable if any sentence in the returned paragraphs contains the answer. In practice, answerability is more nuanced, with some pieces of the information found but not all. A more realistic future approach would involve an ordinal scale (e.g., unanswerable, partially answerable, fully answerable), which would necessitate ground truth assessments with an explicit specification of the different facets/aspects of the answer.

[3] The prompts can be found in the documentation accompanying the repository.

Bibliography

[1] A. Asai and E. Choi. Challenges in information-seeking qa: Unanswerable questions and paragraph retrieval. In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing*, volume 1 of *ACL-IJNLP '21*, pages 1492–1504, 2021.

[2] V. Bolotova-Baranova, V. Blinov, S. Filippova, F. Scholer, and M. Sanderson. Wikihowqa: A comprehensive benchmark for multi-document non-factoid question answering. In *Annual Meeting of the Association for Computational Linguistics*, ACL '23, 2023.

[3] Z. Cao, W. Li, S. Li, F. Wei, and Y. Li. Attsum: Joint learning of focusing and summarization with neural attention. In *Proceedings of COLING 2016, the 26th International Conference on Computational Linguistics*, COLING '16, pages 547–556, 2016.

[4] E. Choi, H. He, M. Iyyer, M. Yatskar, W.-t. Yih, Y. Choi, P. Liang, and L. Zettlemoyer. Quac: Question answering in context. In *Findings of the Asso-492 ciation for Computational Linguistics: EMNLP 2018*, EMNLP '18, pages 2174–2184, 2018.

[5] J. Dalton, C. Xiong, and J. Callan. Trec cast 2019: The conversational assistance track overview. In *The Twenty-Eighth Text REtrieval Conference Proceedings*, TREC '19, 2019.

[6] J. Dalton, C. Xiong, and J. Callan. Cast 2020: The conversational assistance track overview. In *The Twenty-Ninth Text REtrieval Conference Proceedings*, TREC '20, 2020.

[7] J. Dalton, C. Xiong, and J. Callan. Trec cast 2021: The conversational assistance track overview. In *The Thirtieth Text REtrieval Conference Proceedings*, TREC '21, 2021.

[8] F. Godin, A. Kumar, and A. Mittal. Learning when not to answer: a ternary reward structure for reinforcement learning based question answering. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, volume 2 of *NAACL-HLT '19*, pages 122–129, 2019.

[9] M. Hu, F. Wei, Y. Peng, Z. Huang, N. Yang, and D. Li. Read + verify: Machine reading comprehension with unanswerable questions. *arXiv:1808.05759*, 2018.

[10] K. Huang, Y. Tang, J. Huang, X. He, and B. Zhou. Relation module for non-answerable predictions on reading comprehension. In *Proceedings of the 23rd Conference on Computational Natural Language Learning*, CoNLL '19, pages 747–756, 2019.

[11] O. Kurland and J. S. Culpepper. Fusion in information retrieval: Sigir 2018 half-day tutorial. In *The 41st International ACM SIGIR Conference on Research & Development in Information Retrieval*, SIGIR '18, pages 1383–1386, 2018.

[12] J. Liao, X. Zhao, J. Zheng, X. Li, F. Cai, and J. Tang. Ptau: Prompt tuning for attributing unanswerable questions. In *Proceedings of the 45th International ACM SIGIR Conference on Research and Development in Information Retrieval*, SIGIR '22, pages 1219–1229, 2022.

[13] Y. Luan, J. Eisenstein, K. Toutanova, and M. Collins. Sparse, dense, and attentional representations for text retrieval. *Transactions of the Association for Computational Linguistics*, 9: 329–345, 2021.

[14] P. Owoicho, J. Dalton, M. Aliannejadi, L. Azzopardi, J. R. Trippas, and S. Vakulenko. Trec cast 2022: Going beyond user ask and system retrieve with initiative and response generation. In *The Thirty-First Text REtrieval Conference Proceedings*, TREC '22, 2022.

[15] V. Pavlu, S. Rajput, P. B. Golbus, and J. A. Aslam. Ir system evaluation using nugget-based test collections. In *Proceedings of the fifth ACM international conference on Web search and data mining*, WSDM '12, pages 393–402, 2012.

[16] R. Pradeep, R. Nogueira, and J. Lin. The expando-mono-duo design pattern for text ranking with pretrained sequence-to-sequence models. *arXiv:2101.05667*, 2021.

[17] P. Rajpurkar, R. Jia, and P. Liang. Know what you don't know: Unanswerable questions for squad. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics*, volume 2 of *ACL '18*, pages 784–789, 2018.

[18] S. Reddy, D. Chen, and C. D. Manning. Coqa: A conversational question answering challenge. *Transactions of the Association for Computational Linguistics*, 7:249–266, 2019.

[19] P. Ren, Z. Chen, Z. Ren, E. Kanoulas, C. Monz, and M. De Rijke. Conversations with search engines: Serp-based conversational response generation. *ACM Transactions on Information Systems*, 39(4):1–29, 2021.

[20] E. Sulem, J. Hay, and D. Roth. Yes, no or idk: The challenge of unanswerable yes/no questions. In *Proceedings of the 2022 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, NAACL-HLT '22, pages 1075–1085, 2022.

[21] L. Tang, T. Goyal, A. R. Fabbri, P. Laban, J. Xu, S. Yavuz, W. Kryściński, J. F. Rousseau, and G. Durrett. Understanding factual errors in summarization: Errors, summarizers, datasets, error detectors. *arXiv:2205.12854*, 2023.

[22] S. Vakulenko, S. Longpre, Z. Tu, and R. Anantha. Question rewriting for conversational question answering. In *Proceedings of the 14th ACM International Conference on Web Search and Data Mining*, WSDM '21, pages 355–363, 2021.

[23] J. Zhang, Y. Zhao, M. Saleh, and P. J. Liu. Pegasus: pre-training with extracted gap-sentences for abstractive summarization. *arXiv:1912.08777*, 2020.