# ASQA: Factoid Questions Meet Long-Form Answers

**Ivan Stelmakh**[1*] **Yi Luan**[3]
**Bhuwan Dhingra**[2,3] **Ming-Wei Chang**[3]
[1]Yakov & Partners [2]Duke University [3]Google Research
stelmakh95@icloud.com
{luanyi,bdhingra,mingweichang}@google.com

## Abstract

An abundance of datasets and availability of reliable evaluation metrics have resulted in strong progress in *factoid question answering* (QA). This progress, however, does not easily transfer to the task of *long-form QA*, where the goal is to answer questions that require in-depth explanations. The hurdles include (i) a lack of high-quality data, and (ii) the absence of a well-defined notion of the answer's quality. In this work, we address these problems by (i) releasing a novel dataset and a task that we call ASQA (Answer Summaries for Questions which are Ambiguous); and (ii) proposing a reliable metric for measuring performance on ASQA. Our task focuses on factoid questions that are ambiguous, that is, have different correct answers depending on interpretation. Answers to ambiguous questions should synthesize factual information from multiple sources into a long-form summary that resolves the ambiguity. In contrast to existing long-form QA tasks (such as ELI5), ASQA admits a clear notion of correctness: a user faced with a good summary should be able to answer different interpretations of the original ambiguous question. We use this notion of correctness to define an automated metric of performance for ASQA. Our analysis demonstrates an agreement between this metric and human judgments, and reveals a considerable gap between human performance and strong baselines.

## 1 Introduction

In the last few years, the factoid question answering (QA) task—extracting short answers to *factoid* questions—has witnessed significant progress (Lee et al., 2019; Guu et al., 2020; Karpukhin et al., 2020; Lewis et al., 2020; Izacard and Grave, 2021). The progress was achieved in large part thanks to (i) the availability of high-quality datasets (Voorhees and Tice, 2000; Joshi et al., 2017; Yang et al., 2018; Abujabal et al., 2019; Kwiatkowski et al., 2019), and (ii) a well-defined notion of correctness. A key challenge for ongoing research now lies in long-form question answering where the goal is to generate detailed explanations in response to questions that require elaborate and in-depth answers.

There is much less data available for the task of long-form QA. One of the primary data sources is the ELI5 dataset (Fan et al., 2019) that pairs open-ended questions with paragraph-long answers written by users of the "Explain Like I'm Five" Reddit forum. However, questions in ELI5 are very general (e.g., `"How can different animals perceive different colors?"`) and can be answered in myriad different ways, making it hard to define objective criteria for a good answer. As a result, Krishna et al. (2021) identify several hurdles in using this data towards meaningful modeling progress, including a lack of reliable evaluation metrics.

In this work, we address the lack of data sources and unreliability of evaluations by constructing a *long-form QA dataset for factoid questions*. Our paper is motivated by the work of Min et al. (2020) who observe that more than half of the factoid questions that occur naturally are *ambiguous*. For example, a seemingly simple question: `"Who was the ruler of France in 1830?"` is ambiguous because there were two rulers of France in 1830. Min et al. (2020) collected the AMBIGQA dataset that connects ambiguous factoid questions with *disambiguations*: pairs of disambiguated questions and unique short answers to these questions (see example on the right side of Figure 1).

We note, however, that ambiguous questions often arise when a user lacks background knowledge about *why* there might be multiple answers to their question, and *how* those answers relate to each other. Thus, the list of disambiguations may not be satisfactory for the user. For example, the fact that in 1830 the ruler of France *changed due to the revolution* is highly salient but is not captured in

[*]Work done during an internship at Google Research.



*Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing*, pages 8273 - 8288

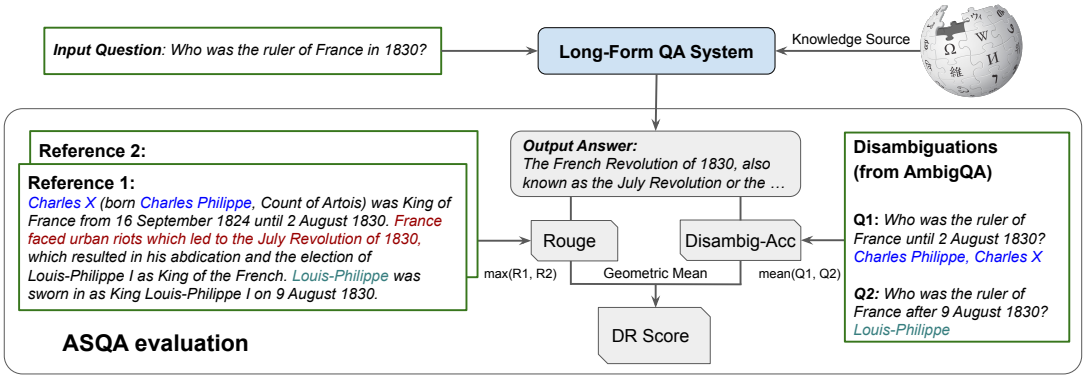


Figure 1: The input questions in ASQA are sourced from AMBIGQA. Long-form answers must be sufficient to answer disambiguated questions from AMBIGQA (short answers are marked in *blue* and *green*), and should introduce additional knowledge from Wikipedia (highlighted in *red*) to resolve ambiguity and clarify the relationship between different short answers. The DR score we propose combines ROUGE and Disambiguation-accuracy (Disambig-Acc) metrics, overcoming the issues with long-form QA evaluation outlined by Krishna et al. (2021).

the AMBIGQA disambiguations.

In this paper, we argue the importance of generating long-form answers to ambiguous factoid questions. In that, we present ASQA (**A**nswer **S**ummaries for **Q**uestions which are **A**mbiguous)—a novel dataset that pairs each ambiguous question from AMBIGQA with a crowdsourced long-form answer.[1] The answers we collect aim to (i) explain the source of ambiguity in the question, and (ii) connect all the valid short answers into a coherent passage. An example ASQA instance is shown in Figure 1.

The main feature of ASQA is a combination of (i) *a well-defined notion of correctness* pertinent to factoid QA and (ii) *the complexity* of long-form QA. First, observe that a good answer to an ambiguous question should be sufficient for the user to answer different interpretations of the question. This observation induces a notion of correctness that is conceptually similar to the conventional accuracy in factoid QA. Second, to answer an ambiguous question, a system needs to retrieve a diverse set of documents that talk about different interpretations of the question and synthesize this information into a coherent summary. Thus, the key challenges of long-form QA—precise retrieval and high-quality summarization—are present in ASQA.

**Contributions** Overall, our work makes several contributions:

- First, we carefully develop a crowdsourcing pipeline and collect ASQA—a dataset of high-quality long-form answers to 6,316 ambiguous factoid questions.
- Second, we design principled evaluation procedures for ASQA: (i) we propose a novel automated evaluation metric (DR) that combines the correctness aspect of factoid QA and the fluency aspect of long-form QA; (ii) we develop and release a convenient interface for human evaluations; (iii) we conduct a small-scale human study that shows a high agreement between our automated metric DR and human judgments.
- Third, we establish strong baselines for our task by combining joint passage retrieval (Min et al., 2021) and T5-large (Raffel et al., 2019). Our extensive evaluations demonstrate that there is a large gap between the baselines and human performance. Additionally, we highlight areas of improvement for future research on ASQA.

## 2 Related Work

In this section, we describe relevant works that propose new tasks, datasets, and methods for QA and summarization problems.

**Extractive QA** Much of the existing work on question answering, including *reading comprehension* (Rajpurkar et al., 2016, 2018; Trischler et al., 2017; Yang et al., 2018), *open-domain QA* (Kwiatkowski et al., 2019; Joshi et al., 2017) and *dialog-based QA* (Choi et al., 2018), assumes that questions have unique answers. Min et al. (2020) relax this assumption and propose a task that aims at identifying all possible short answers to the

[1]Data, evaluation scripts, and other supplementary materials are provided on the project's GitHub repository: https://github.com/google-research/language/tree/master/language/asqa

ambiguous subset of the open-domain version of the NQ dataset, denoted NQ-OPEN (Kwiatkowski et al., 2019; Lee et al., 2019). The AMBIGQA dataset constructed by Min et al. (2020) serves as a building block of the present work and we provide more details on this dataset in Section 3. Another related effort is the CONDITIONALQA task (Sun et al., 2021) that requires systems to identify *conditions* under which the extracted answers are valid. Unlike the ASQA task, the answers in CONDITIONALQA come from a document provided in advance and do not need to be summarized into a single response.

**Generative QA** Extractive models achieve good results when the answer to the question is readily available on the web. However, in many settings, including ambiguous factoid questions, a system needs to combine information from many (unknown) sources to present the answer to the user in a convenient way. Hence, in this work, we focus on the *generative QA* setting where a model needs to generate a textual answer rather than extract it.

Datasets for generative QA include NARRATIVEQA (Kočiský et al., 2018) and COQA (Reddy et al., 2019), but the average answer length in these datasets is small: 4.7 and 2.7 tokens, respectively. The MS MARCO Natural Language Generation (MS-NLG) dataset by Nguyen et al. (2016) combines both extractive and generative tasks and contains slightly longer human-generated answers (usually, a sentence-long) that can be read by a smart assistant. Fan et al. (2019) proposed a more challenging task of answering open-ended (e.g., "why?") questions. They scraped the *"Explain Like I'm Five"* Reddit forum and released a dataset of ∼272K questions, where each question is supplied with several paragraph-long answers generated by the Reddit users. We overview the differences between ASQA, ELI5 and MS-NLG in Section 3.3.

Recently, large language models such as GPT-3 (Brown et al., 2020) have been successfully applied to the task of long-form QA using the ELI5 dataset (Nakano et al., 2021). For this, a two-step human-in-the-loop approach was involved: first, demonstrations of annotators navigating the web to write answers were collected; second, a reward model (Stiennon et al., 2020) was trained by manual pairwise comparisons of answers. In ASQA, relevant passages for the answer are already provided by the annotators and we show that the proposed DR score correlates well with the human judgment of answer quality. Using this automated metric in place of the reward model in the approach of Nakano et al. (2021) is a potential direction for future work.

**Summarization** Given a set of documents relevant to the question (either ground truth or obtained using retrieval) the problem of generating a long-form answer reduces to query-based multi-document summarization. A small-scale dataset for this task was introduced as part of the DUC tasks (Dang, 2005). Recent work on building large-scale datasets has instead focused either on query-based summarization from a single document (Nema et al., 2017; Zhong et al., 2021) or on multi-document summarization without queries (Liu et al., 2018; Fabbri et al., 2019). In addition to the QA task, the ASQA dataset is suitable for the evaluation of systems' accuracy in the summarization setting, where the ground-truth passages containing the relevant information are assumed to be given.

**QA-Based Evaluation** Prior work has looked at using question answering techniques to evaluate factual consistency in summarization (Wang et al., 2020; Durmus et al., 2020) and dialogue (Honovich et al., 2021). These works automatically generate questions from the system-produced text and search for answers in some reference text (e.g., the input being summarized) to evaluate the quality of the output. Instead, to evaluate generated long-form answers to ambiguous questions, in ASQA we use questions created by AMBIGQA annotators.

## 3 ASQA Task and Data

In this section, we introduce the ASQA task and the underlying data-collection process. The ASQA task is illustrated in Figure 1. The goal of the task is to write a comprehensive paragraph-long answer $\hat{a}$ to a given ambiguous question $q$.

**Source Data** We build ASQA on top of the subset of ambiguous questions identified in the AMBIGQA dataset. Out of a total of 14,042 AMBIGQA questions, 7,207 are identified as ambiguous by at least one AMBIGQA annotator. Each of these ambiguous questions $q$ is paired with a list of $n$ *disambiguations* $\{(x_i, y_i)\}_{i=1}^{n}$, where $x_i$ denotes a disambiguated question and $y_i$ denotes

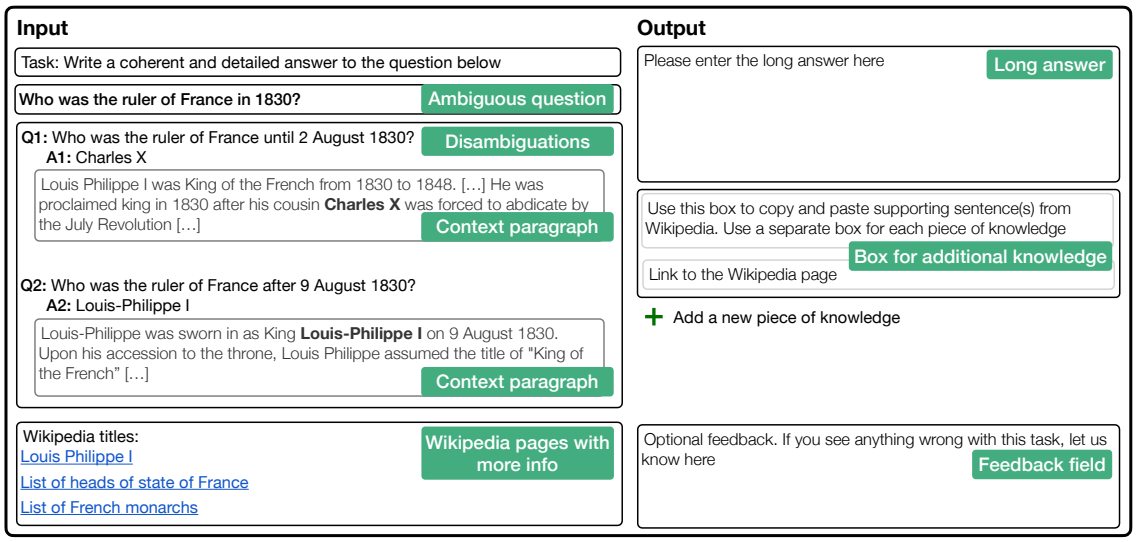


Figure 2: Schematic representation of the annotation interface.

a unique *short answer* to $x_i$. The number of disambiguations ranges from 2 to 46 per ambiguous question. To ensure that it is feasible to put all this information into a coherent story, we remove 417 questions with more than six disambiguations from consideration, thereby focusing on 6,790 AMBIGQA instances that we use as a starting point for building our task.

### 3.1 ASQA Annotation Objectives

At a high level, the goal of the annotation process is to obtain high-quality long answers to ambiguous questions. We begin with a formulation of criteria for what counts as a good long answer to an ambiguous question:

- **Completeness** The long answer should contain all valid short answers $y_1, \ldots, y_n$ to the disambiguated questions $x_1, \ldots, x_n$ in an appropriate context.
- **Comprehensiveness** The long answer should provide enough details for the user to (i) understand the source of ambiguity in the original question and (ii) understand the relationship between different short answers.
- **Fluency** The long answer should be coherent and fluent.
- **Attributability** The long answer should be grounded in an underlying source of information (in our case, Wikipedia).

### 3.2 ASQA Annotation Process

To ensure that annotations satisfy the aforementioned objectives, we develop a custom annotation interface (Figure 2) and recruit native English speakers to perform our task. We then collect long-form answers for each target instance of AMBIGQA using a commercial crowdsourcing platform where it is possible to interact with the annotators on an ongoing basis. Let us now discuss the key components of our annotation pipeline.

**Input to Annotators** The left side of Figure 2 illustrates the input to our annotation procedure. Annotators are given relevant aspects of the target AMBIGQA instance: the ambiguous question $q$, list of disambiguations $\{(x_i, y_i)\}_{i=1}^n$, and the Wikipedia pages $W$ visited by AMBIGQA annotators. Additionally, to help annotators understand the context behind the disambiguations without reading full Wikipedia articles, for each disambiguation $i$ we provide a (possibly empty) Wikipedia passage $C_i$ with information relevant to the disambiguation. Details on the procedure used to find these context passages $\{C_i\}_{i=1}^n$ are given in Appendix A.

**Output of Annotation** The key output of annotation is a long-form answer $a$ to a given ambiguous question $q$. Additional elements of the output are introduced to facilitate the requirement of attributability. In that, we require annotators to provide the source Wikipedia passage $e$ for each piece of additional information they bring to their answer. Our interface has designated fields for additional knowledge (see Figure 2) and annotators can add

| SPLIT | # QUESTIONS | # ANNOTATIONS |
|---|---|---|
| TRAIN | 4,353 | 1 |
| DEV | 948 | 2 |
| TEST | 1,015 | 2 |

Table 1: Summary statistics of the ASQA dataset.

as many of these fields as they need to include any number $m$ of evidence passages $\{e_j\}_{j=1}^{m}$.

**Instructions, Training and Quality Control** We carefully design instructions, a training procedure, and quality control tools to minimize the amount of noise in annotations. Details on these aspects of the annotation pipeline are provided in Appendix A.

### 3.3 ASQA Dataset

By following the procedure outlined above, we annotated train, dev, and test splits of the AMBIGQA dataset. Each question in the train split was annotated by a single annotator while the dev and test splits have two annotations per question.

For 474 questions, our annotators raised concerns regarding the validity of the AMBIGQA disambiguations. Not all of these concerns necessarily indicate errors in the AMBIGQA dataset as some of them could be due to misinterpretation on the annotators' side. Nevertheless, to maintain data fidelity, we exclude the corresponding instances from the resulting dataset. Table 1 displays the final breakdown of the ASQA dataset.

Table 2 compares ASQA to other open-domain QA datasets: ELI5, MS-NLG, AMBIGQA, and NQ-OPEN. We observe that ASQA requires long answers with an average length of 64.8 (vs. 103.0 for ELI5 and 14.6 for MS-NLG), and is the only dataset that admits evaluations in terms of both ROUGE, which is typically used for long-form QA, and accuracy, which is typically used for factoid QA. This makes ASQA an appealing dataset as it enables researchers to work on long-form QA while retaining the benefits of reliable objective evaluation typical in factoid QA.

**Additional Comparison to ELI5** ELI5 is the closest existing long-form QA dataset. We now provide additional comparison of ASQA and ELI5.

*Support Documents* First, both ASQA and ELI5 supplement annotations with relevant information retrieved from Wikipedia (ASQA) or the whole Internet (ELI5). For ELI5, support documents are retrieved automatically and independently of the annotation process. The resulting documents contain, on average, 858 words. Manual analysis conducted by Fan et al. (2019) reveals that support documents are sufficient to answer 65% of the questions and have information relevant to 92% of the questions.

In ASQA, support documents are constructed as a part of the annotation process. For each annotation, the support document contains disambiguations from AMBIGQA, context paragraphs, and additional knowledge provided by the corresponding annotator (see Section 3.2 for details). On average, support documents contain 225 words, being much shorter than those for ELI5. By design of our annotation procedure, support documents should be sufficient to write long-form answers to ambiguous questions. Indeed, we observe that 92% of the annotations' tokens are present in the corresponding support documents.[2] If we exclude AMBIGQA disambiguations from the support documents, their average length reduces to 172 words, but 78% of tokens from the answers remain captured therein. These observations demonstrate that ASQA satisfies the requirement of attributability (Section 3.1).

*Inter-Annotator Agreement* Second, we compare the inter-annotator agreement in ELI5 and ASQA that we measure as the mean ROUGE-L F1 score between each pair of annotations for the same question. Our analysis reveals that ASQA has a much higher level of inter-annotator agreement: 49.6 vs. 16.9 for ELI5. Thus, ASQA admits a more well-defined notion of ground truth than ELI5.

Note that answers in ELI5 are written by Reddit users. Thus, they are inherently subjective and are not supposed to follow any predefined criteria. The diversity and subjectiveness could make human evaluation of the ELI5 answers challenging. In contrast, ASQA annotators follow common annotation guidelines and undergo a thorough training procedure, thereby aiming at generating answers that satisfy a set of well-defined criteria for human evaluation (Section 3.1).

Overall, compared to other datasets, ASQA has some novel features that may be useful for future QA research. Its benefits, however, come at the cost of a much smaller sample size than that of MS-NLG and ELI5. Thus, we believe MS-NLG and ELI5 may be useful counterparts for ASQA

[2]This statistic is computed as the ROUGE1 recall score between lowercased annotations and support documents. In this work, we use ROUGE-SCORE 0.0.4 python package for all ROUGE computations.

| QA Task | Dataset | #QAs | Dev Set Statistics | | Evaluation | |
|---|---|---|---|---|---|---|
| | | | #A per Q | #Words in A | ROUGE | Disambig-Acc |
| Short Answer | NQ-open | 91K | 1.8 | 2.2 | ✗ | ✓† |
| | AmbigQA | 14,042 | 2.8 | 2.4 | ✗ | ✓ |
| Long Form | ELI5 | 272K | 12.0 | 103.0 | ✓ | ✗ |
| | MS-NLG | 183K | 1.7 | 14.6 | ✓ | ✗ |
| | ASQA | 6,316 | 2.0 | 64.8 | ✓ | ✓ |

Table 2: Comparison of ASQA with existing open domain QA datasets. ASQA is the only QA dataset that allows for both ROUGE and accuracy evaluations. †Standard accuracy for non-ambiguous questions.

as they can be used for pre-training (that said, we leave this exploration to future work).

## 4 ASQA Metrics

In this section, we introduce metrics that we propose to evaluate performance on the ASQA task.

### 4.1 Automated Evaluation

We evaluate performance on the ASQA task along the following two aspects.

**ROUGE** Following the conventional approach for measuring the quality of generated text, we report the ROUGE-L score (Lin, 2004) in a multi-reference setup.[3] Given that each example in the development and test sets is annotated by two annotators, we compare predictions against both answers and take the *maximum* of these two scores to be the score of the prediction.

**Disambiguation Metrics** A good long-form answer to an ambiguous question should contain short answers to all disambiguated questions as well as the context necessary to understand the source of ambiguity and the relationship between the short answers. However, ROUGE-L is not well suited for evaluating these aspects as it may fail to distinguish between two fluent and stylistically similar answers which provide considerably different information. Therefore, we complement ROUGE-L with two metrics that are specifically designed to capture the *completeness* and *comprehensiveness* aspects of our task:

- STR-EM (String Exact Match) The fraction of disambiguations for which the corresponding short answer is present in the long answer (exact match). The fraction is computed within each question and then averaged across all questions.
- Disambig-F1 We follow the reading comprehension literature (Rajpurkar et al., 2016, 2018) and use Roberta (Liu et al., 2019) trained on SQuADv2 to evaluate the fraction of disambiguated questions that can be answered from the predicted long answers.[4] For each disambiguation $(x_i^{(k)}, y_i^{(k)})$ in the $k$-th example, we apply the SQuADv2 model on the generated long-form answer $\hat{a}^{(k)}$ to predict short answer $\hat{y}_i^{(k)}$ to question $x_i^{(k)}$. Let $\phi$ denote a function that computes the token-level F1 score between the predicted short answer $\hat{y}_i^{(k)}$ and the ground truth short answer $y_i^{(k)}$ after normalizing answer strings in the manner done for SQuADv2 evaluations. Then the Disambig-F1 score is given by:

$$\text{Disambig-F1} = \frac{1}{N}\sum_k \frac{1}{n^{(k)}}\sum_i \phi(\hat{y}_i^{(k)}, y_i^{(k)}),$$

  where $N$ indicates the total number of instances being evaluated, and $n^{(k)}$ indicates the number of disambiguations for the $k$-th instance.

**Overall: DR Score** Both ROUGE-L and disambiguation metrics are crucial for our task. Hence, we propose an overall DR (Disambiguation-Rouge) score that combines the two metrics as follows:

$$\text{DR} = \sqrt{\text{Disambig-F1} \times \text{ROUGE-L}}.$$

We choose the geometric mean for aggregation to penalize methods that maximize one metric at a cost of the other. Note that STR-EM and Disambig-F1 aim at measuring the same aspect so we include only one of these metrics in the DR score.

### 4.2 Human Evaluation

We also design an interface for human evaluations for the ASQA task with the following metrics.

[3]We use the python `rouge-score` package. Candidate and reference summaries are lowercased and stemmed using the Porter stemmer.

[4]We use Huggingface training and evaluation scripts (Wolf et al., 2020).

- *Disambiguation Accuracy* For each long-form answer, we ask human annotators to verify whether each disambiguated question from the AMBIGQA dataset can be correctly answered using the provided information. We then report the average number of disambiguations that are captured in the long-form answers (ACC).
- *Pairwise Comparisons* We propose a pairwise evaluation scheme where annotators need to compare two long-form answers to the same question. We ask annotators to choose the better answer in terms of each of the three criteria: Comprehensiveness (COMP), Fluency (FLUE), and Human Overall impression (HO). In each pairwise comparison, an answer is given one point for victory and half for a tie. We then normalize model scores into percentages by dividing the total number of points a model receives by the number of pairwise comparisons.

## 5 Experimental Setup

We now describe the baseline models and human answers used in our experiments.

### 5.1 Models

We include the following models for comparison.

**Naïve** The naïve model (denoted as QUESTION) repeats the ambiguous question eight times.

**Retrieval-Only** The retrieval-only models retrieve a Wikipedia passage as the answer:

- DPR@1. DPR (Karpukhin et al., 2020) is a BERT-based dual encoder trained on NQ.
- JPR@1. JPR (Min et al., 2021) trains a reranker on top of DPR for questions with multiple answers in AMBIGQA. The JPR model is the state of the art retriever for AMBIGQA.

**Generative** We also evaluate T5-large based generative models (Raffel et al., 2019) in two regimes:

- T5 Closed Book (T5-C). We train T5 to answer ambiguous questions without providing any additional passages from Wikipedia. The model only relies on its pretrained knowledge to answer the question (Roberts et al., 2020).
- T5 Open Book (T5-O). The T5 model is additionally provided with context paragraphs retrieved by JPR. We vary the number of top-$K$ retrieved paragraphs used as input to T5, denoting the corresponding model as T5-O-K.

**Oracle** To investigate the headroom in retrieval systems, we experiment with an ORACLE system: T5-large provided with the gold supporting documents. The input to ORACLE includes all the disambiguations $\{(x_i, y_i)\}_{i=1}^n$ and contexts $\{C_i\}_{i=1}^n$ shown to the annotators (left half of Figure 2), as well as the additional knowledge pieces $\{e_j\}_{j=1}^m$ identified by one of the two annotators (the one with the longest answer). This system can be thought of as a generative model that has access to a perfect retriever. In evaluations, we compute ROUGE-L by comparing the answer predicted by ORACLE against the answer of the annotator whose additional knowledge pieces were *not* in the input of ORACLE (instead of the usual comparison against two references).

Appendix B provides more details on the modeling aspects of our evaluations.

### 5.2 Human Performance

We also evaluate two sets of human answers:

- Human performance with context (HP-W/-C). We use reference ASQA answers in our comparisons. Recall that the ASQA annotators were provided with context: disambiguations from AMBIGQA $\{(x_i, y_i)\}_{i=1}^n$ and context paragraphs we retrieved $\{C_i\}_{i=1}^n$. We consider performance in this setup as an upper bound on the human performance. In evaluations of ROUGE-L, we compute the score of HP-W/-C by comparing the answers from two annotators against each other (instead of the usual comparison against two references).
- Human performance without context (HP-W/O-C). To establish a conservative lower bound on human performance, we additionally annotate 200 questions from the ASQA dev set (one annotation per question) in the "no context" regime. Annotators in this regime are only given ambiguous questions as input (no disambiguations or context paragraphs) and need to search for disambiguations and the required additional information on their own.

## 6 Results

We evaluate all models introduced above in the automated evaluations. Additionally, we conduct a small-scale human study involving a subset of models to provide some verification of the automated evaluation results. Specifically, our human study

| | LEN (WRDS) | ROUGE-L | STR-EM | DISAMBIG-F1 | DR |
|---|---|---|---|---|---|
| QUESTION | 71.6 | 15.3 | 1.2 | 0.2 | 1.5 |
| DPR@1 | 99.9 | 31.1 | 30.1 | 16.7 | 22.8 |
| JPR@1 | 196.8 | 27.9 | 45.0 | 25.8 | 26.9 |
| T5 CLOSED BOOK (T5-C) | 62.5 | 31.0 | 10.3 | 7.4 | 15.1 |
| T5 OPEN BOOK 1 PASSAGE (T5-O-1) | 63.0 | 36.5 | 33.6 | 21.2 | 27.9 |
| T5 OPEN BOOK 3 PASSAGES (T5-O-3) | 71.1 | 38.8 | 39.9 | 25.1 | 31.2 |
| T5 OPEN BOOK 5 PASSAGES (T5-O-5) | 71.6 | 39.2 | 41.0 | 26.4 | 32.1 |
| T5 OPEN W/ ORACLE CONTEXT (ORACLE) | 82.6 | 46.6* | 88.7 | 59.2 | 52.5* |
| HUMAN W/O CONTEXT (HP-W/O-C) | 73.5 | 42.2 | 51.8 | 39.0 | 40.6 |
| HUMAN W/ CONTEXT (HP-W/-C) | 64.8 | 49.4* | 98.4 | 77.4 | 61.8* |

Table 3: Evaluation of baselines on the dev set of the ASQA task. T5 models with passages retrieved by JPR are the best models, but there is a large gap between human performance and model performance on all metrics. *As explained in Section 5, for ORACLE and HP-W/-C we only use one of the references to compute ROUGE-L.

involves four model outputs (JPR@1, T5-C, T5-O-1, T5-O-5) and two sets of human-generated answers (HP-W/O-C, HP-W/-C) that are juxtaposed on a subset of 45 randomly chosen questions from the development set of ASQA. For each of the questions, six target answers are split into three pairs and pairwise comparisons are conducted by authors of this paper in a blind manner.

**Importance of Retrieval** Models that take the output of a retrieval system (T5-O-1/3/5) perform much stronger than the closed-book model (T5-C) on both automated metrics and the human evaluation. T5-O-1 outperforms T5-C by 20.0 points on human evaluation (HO) and by 12.8 points on DR. T5-O-5 outperforms T5-C by 15.6 points on HO and by 17.0 points on DR.

Following Krishna et al. (2021), we also experimented with a *random retrieval* baseline where, during inference, the model was provided randomly selected retrieved passages from the training set. This baseline gets a DR of only 7.8, further confirming that, different from ELI5, retrieval is very important for ASQA.

**Importance of Summarization** Retrieval is very important for ASQA, but just using the top retrieved passage from a strong system (JPR@1) is not sufficient. Even though the STR-EM and Disambig-F1 metrics of JPR@1 are considerably higher than these of T5-O-1 (by 11.4 and 4.6 points, respectively), the human overall impression score HO and the DR score are similar across these models. This discrepancy is observed because the disambiguation metrics do not evaluate the conciseness of the answers, and the advantage of JPR@1 on these metrics is gained at the cost of the increased answer length (196.8 words). In contrast, T5 models tend to generate shorter answers whose length is much closer to the average length of human references (65 words). Hence, in addition to including the correct information, answers in ASQA must be concise which highlights the importance of summarization.

| | ACC | COMP | FLUE | HO |
|---|---|---|---|---|
| JPR@1 | 36.1 | 44.4 | 42.2 | 37.8 |
| T5 C | 8.4 | 35.6 | 32.2 | 21.1 |
| T5 O-1 | 25.7 | 36.7 | 38.9 | 41.1 |
| T5 O-5 | 28.0 | 36.7 | 37.8 | 36.7 |
| HP-W/O-C | 52.7 | 60.0 | 66.7 | 74.4 |
| HP-W/-C | 94.3 | 86.7 | 82.2 | 88.9 |

Table 4: Results of human evaluations executed on a set of 45 questions from the development set of ASQA. The scores are in percentage and larger values are better. All metrics are specified in Section 4.2.

**Correlation with Human Judgments** Table 5 reports Pearson correlations between different automated metrics and the human judgments, enabling us to study the validity of the automated metrics.

First, we observe that Disambig-F1 is better correlated with human evaluations than ROUGE-L. That said, we note that ROUGE-L is an important metric as it enforces concise answers.

Second, observe that Disambig-F1 scores (Table 3) underestimate the human evaluations of ACC (Table 4). This discrepancy is likely due to: (i) a distribution shift between ASQA and SQUADV2; and (ii) the presence of distracting answers from the other disambiguated questions in the long answers, which are known to degrade QA models' accuracy (Jia and Liang, 2017). However, almost perfect correlation between Disambig-F1 and ACC

| | ROUGE-L | DISAMBIG-F1 | DR |
|---|---|---|---|
| ACC | 81.1 | 99.3 | 97.9 |
| COMP | 79.3 | 96.4 | 93.7 |
| FLUE | 83.4 | 94.4 | 94.4 |
| HO | 86.4 | 92.9 | 95.0 |

Table 5: Correlation between human and automated metrics. DR has the highest correlation with the overall human score HO among all automated metrics.

(99.3) implies that this discrepancy does not impact the ordering of the different systems, thereby enabling us to meaningfully evaluate the relative differences in performance. Additionally, the presence of strong distractors ensures that the Disambig-F1 metric cannot be easily gamed by mentioning all the short answers without appropriate context.

Finally, we note that the DR score has the highest correlation with the overall human judgment HO among all automated metrics. While the difference with Disambig-F1 is not statistically significant, this observation hints at the importance of combining ROUGE-L and Disambig-F1 in the overall metric to take a holistic view on the model performance.

**Remaining Headroom** Both the upper bound (61.8 DR and 88.9 HO) and the lower bound (40.6 DR and 74.4 HO) on human performance significantly exceed the best model performance (T5-O-5 with 32.1 DR and 36.7 HO). Hence, there is a lot of headroom for the community to explore in ASQA. We report some additional insights that may be helpful for future work in Section 7.

## 7 Analysis

We now conduct additional analysis that provides insights on the ASQA task.

**Headroom in Summarization** As shown in Figure 3, the Disambig-F1 score of retrieval-based methods increases considerably as the number of retrieved passages increases. However, there is a big gap between T5 and JPR, even though T5 takes the output passages from JPR as an input. This indicates that T5 tends to either lose information while summarizing the passages or produce outputs that are inconsistent with its input. Moreover, the Disambig-F1 of JPR@5 already exceeds the lower bound on human performance. Thus, progress in summarization alone may be sufficient to raise the overall level of performance on ASQA to this lower bound.

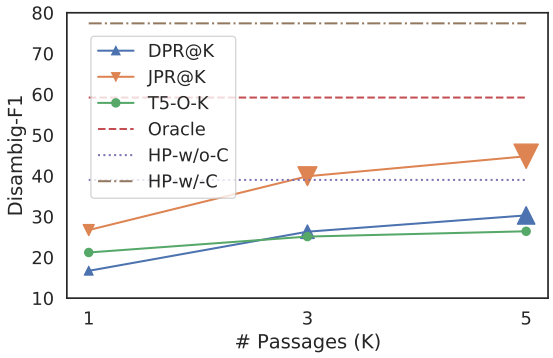


Figure 3: Disambig-F1 of different methods with a varying number of retrieved passages. Marker sizes are proportional to the answer lengths. The T5-O-K score increases with $K$ but there is also an increasing gap between T5-O-K and JPR@K. Passages from the latter are used as input for the former.

To provide further insight into the summarization aspect of our task, we conduct a manual analysis of the answers generated by the open-book T5-O-5 model. Our analysis identifies several characteristic mistakes (hallucination, questions misunderstanding, and repetitions) that need to be addressed to improve performance on the ASQA task. More details on this evaluation are provided in Appendix C.

**Headroom in Retrieval** Figure 3 compares models by Disambig-F1 and the higher score means that the passage generated by a model provides answers to more disambiguated questions. We observe that the best-performing retrieval system, JPR@5, lags behind the output of the ORACLE model by 14.4 and the human upper bound by 32.6. Hence, improving the retrieval step for ASQA is also important.

## 8 Conclusion

In contrast to existing datasets for long-form QA, ASQA admits a clear notion of correctness that we use to define an overall metric of performance (DR). Our empirical evaluations demonstrate that DR correlates well with the human judgment; and there is a large gap between human performance and the strong baselines. Thus, we believe that ASQA is an appealing task for the QA community. Our analysis suggests that strong performance on ASQA is contingent upon both high-quality retrieval and summarization. These aspects constitute important directions for future work on ASQA.

## 9 Limitations

We now make two remarks that we urge the reader to consider when interpreting the results of this work.

**Inter-Annotator Agreement** In Section 3.3, we observed that inter-annotator agreement in ASQA is higher than in ELI5. We note, however, that the high inter-annotator agreement in ASQA is contingent upon the high inter-annotator agreement in the AMBIGQA dataset. Indeed, AMBIGQA disambiguations serve as a shared source of information between the two ASQA annotators working on the same instance, potentially inflating the level of agreement.

That said, Min et al. (2020) observe that human annotators have a decent level of agreement in constructing the disambiguations in AMBIGQA, thereby supporting the observation that ASQA is more objective than ELI5.

**Evaluation Metrics** Second, we caveat that our accuracy metrics (STR-EM and Disambig-F1) only measure the *recall* of the required information in the long answers. In cases where the long answer hallucinates incorrect disambiguations or facts, the accuracy metrics may still be high as long as the correct disambiguations are included. We note, however, that this unnecessary extra information may still be penalized by the ROUGE-L metric. Moreover, in the presence of distractors, we also expect the accuracy of the Roberta model used for reading comprehension to degrade, thereby effectively penalizing a low precision.

On a separate note, the Disambig-F1 metric requires a high-accuracy QA system. Hence, for domains that are significantly different from Wikipedia, fine-tuning the Roberta SQUADV2 model on the task might be important to ensure the effectiveness of the Disambig-F1 metric.

## Acknowledgements

We thank Kristina Toutanova, Kenton Lee, Shashi Narayan for their valuable insights and feedback. We want to specially thank Sewon Min for discussions, as well as for sharing the implementation details on JPR. We also thank anonymous reviewers for providing detailed and insightful comments on our work.

## References

Abdalghani Abujabal, Rishiraj Saha Roy, Mohamed Yahya, and Gerhard Weikum. 2019. ComQA: A community-sourced dataset for complex factoid question answering with paraphrase clusters. 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 307–317, Minneapolis, Minnesota. Association for Computational Linguistics.

Tom Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared D Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, Sandhini Agarwal, Ariel Herbert-Voss, Gretchen Krueger, Tom Henighan, Rewon Child, Aditya Ramesh, Daniel Ziegler, Jeffrey Wu, Clemens Winter, Chris Hesse, Mark Chen, Eric Sigler, Mateusz Litwin, Scott Gray, Benjamin Chess, Jack Clark, Christopher Berner, Sam McCandlish, Alec Radford, Ilya Sutskever, and Dario Amodei. 2020. Language models are few-shot learners. In *Advances in Neural Information Processing Systems*, volume 33, pages 1877–1901. Curran Associates, Inc.

Eunsol Choi, He He, Mohit Iyyer, Mark Yatskar, Wentau Yih, Yejin Choi, Percy Liang, and Luke Zettlemoyer. 2018. QuAC: Question answering in context. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 2174–2184, Brussels, Belgium. Association for Computational Linguistics.

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

Esin Durmus, He He, and Mona Diab. 2020. FEQA: A question answering evaluation framework for faithfulness assessment in abstractive summarization. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 5055–5070, Online. Association for Computational Linguistics.

Alexander Fabbri, Irene Li, Tianwei She, Suyi Li, and Dragomir Radev. 2019. Multi-news: A large-scale multi-document summarization dataset and abstractive hierarchical model. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 1074–1084, Florence, Italy. Association for Computational Linguistics.

Angela Fan, Yacine Jernite, Ethan Perez, David Grangier, Jason Weston, and Michael Auli. 2019. ELI5: Long form question answering. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 3558–3567, Florence, Italy. Association for Computational Linguistics.

Kelvin Guu, Kenton Lee, Zora Tung, Panupong Pasupat, and Ming-Wei Chang. 2020. REALM: retrieval-

augmented language model pre-training. *CoRR*, abs/2002.08909.

Or Honovich, Leshem Choshen, Roee Aharoni, Ella Neeman, Idan Szpektor, and Omri Abend. 2021. $q^2$: Evaluating factual consistency in knowledge-grounded dialogues via question generation and question answering. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pages 7856–7870, Online and Punta Cana, Dominican Republic. Association for Computational Linguistics.

Gautier Izacard and Edouard Grave. 2021. Leveraging passage retrieval with generative models for open domain question answering. In *Proceedings of the 16th Conference of the European Chapter of the Association for Computational Linguistics: Main Volume*, pages 874–880, Online. Association for Computational Linguistics.

Robin Jia and Percy Liang. 2017. Adversarial examples for evaluating reading comprehension systems. In *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing*, pages 2021–2031, Copenhagen, Denmark. Association for Computational Linguistics.

Mandar Joshi, Eunsol Choi, Daniel Weld, and Luke Zettlemoyer. 2017. TriviaQA: A large scale distantly supervised challenge dataset for reading comprehension. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1601–1611, Vancouver, Canada. Association for Computational Linguistics.

Vladimir Karpukhin, Barlas Oguz, Sewon Min, Patrick Lewis, Ledell Wu, Sergey Edunov, Danqi Chen, and Wen-tau Yih. 2020. Dense passage retrieval for open-domain question answering. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 6769–6781, Online. Association for Computational Linguistics.

Tomáš Kočiský, Jonathan Schwarz, Phil Blunsom, Chris Dyer, Karl Moritz Hermann, Gábor Melis, and Edward Grefenstette. 2018. The NarrativeQA reading comprehension challenge. *Transactions of the Association for Computational Linguistics*, 6:317–328.

Kalpesh Krishna, Aurko Roy, and Mohit Iyyer. 2021. Hurdles to progress in long-form question answering. In *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 4940–4957, Online. Association for Computational Linguistics.

Tom Kwiatkowski, Jennimaria Palomaki, Olivia Redfield, Michael Collins, Ankur Parikh, Chris Alberti, Danielle Epstein, Illia Polosukhin, Jacob Devlin, Kenton Lee, Kristina Toutanova, Llion Jones, Matthew Kelcey, Ming-Wei Chang, Andrew M. Dai, Jakob Uszkoreit, Quoc Le, and Slav Petrov. 2019. Natural questions: A benchmark for question answering research. *Transactions of the Association for Computational Linguistics*, 7:452–466.

Kenton Lee, Ming-Wei Chang, and Kristina Toutanova. 2019. Latent retrieval for weakly supervised open domain question answering. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 6086–6096, Florence, Italy. Association for Computational Linguistics.

Patrick S. H. Lewis, Ethan Perez, Aleksandra Piktus, Fabio Petroni, Vladimir Karpukhin, Naman Goyal, Heinrich Küttler, Mike Lewis, Wen-tau Yih, Tim Rocktäschel, Sebastian Riedel, and Douwe Kiela. 2020. Retrieval-augmented generation for knowledge-intensive NLP tasks. *CoRR*, abs/2005.11401.

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.

Peter J Liu, Mohammad Saleh, Etienne Pot, Ben Goodrich, Ryan Sepassi, Lukasz Kaiser, and Noam Shazeer. 2018. Generating wikipedia by summarizing long sequences. *arXiv preprint arXiv:1801.10198*.

Yinhan Liu, Myle Ott, Naman Goyal, Jingfei Du, Mandar Joshi, Danqi Chen, Omer Levy, Mike Lewis, Luke Zettlemoyer, and Veselin Stoyanov. 2019. Roberta: A robustly optimized BERT pretraining approach. *CoRR*, abs/1907.11692.

Sewon Min, Kenton Lee, Ming-Wei Chang, Kristina Toutanova, and Hannaneh Hajishirzi. 2021. Joint passage ranking for diverse multi-answer retrieval. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pages 6997–7008, Online and Punta Cana, Dominican Republic. Association for Computational Linguistics.

Sewon Min, Julian Michael, Hannaneh Hajishirzi, and Luke Zettlemoyer. 2020. AmbigQA: Answering ambiguous open-domain questions. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 5783–5797, Online. Association for Computational Linguistics.

Reiichiro Nakano, Jacob Hilton, Suchir Balaji, Jeff Wu, Long Ouyang, Christina Kim, Christopher Hesse, Shantanu Jain, Vineet Kosaraju, William Saunders, et al. 2021. Webgpt: Browser-assisted question-answering with human feedback. *arXiv preprint arXiv:2112.09332*.

Preksha Nema, Mitesh M. Khapra, Anirban Laha, and Balaraman Ravindran. 2017. Diversity driven attention model for query-based abstractive summarization. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1063–1072, Vancouver, Canada. Association for Computational Linguistics.

Tri Nguyen, Mir Rosenberg, Xia Song, Jianfeng Gao, Saurabh Tiwary, Rangan Majumder, and Li Deng. 2016. MS MARCO: A human generated MAchine Reading COmprehension dataset. *CoRR*, abs/1611.09268.

Colin Raffel, Noam Shazeer, Adam Roberts, Katherine Lee, Sharan Narang, Michael Matena, Yanqi Zhou, Wei Li, and Peter J. Liu. 2019. Exploring the limits of transfer learning with a unified text-to-text transformer. *CoRR*, abs/1910.10683.

Pranav Rajpurkar, Robin Jia, and Percy Liang. 2018. 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: Short Papers)*, pages 784–789, Melbourne, Australia. Association for Computational Linguistics.

Pranav Rajpurkar, Jian Zhang, Konstantin Lopyrev, and Percy Liang. 2016. SQuAD: 100,000+ questions for machine comprehension of text. In *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing*, pages 2383–2392, Austin, Texas. Association for Computational Linguistics.

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.

Adam Roberts, Colin Raffel, and Noam Shazeer. 2020. How much knowledge can you pack into the parameters of a language model? In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 5418–5426, Online. Association for Computational Linguistics.

Claude Sammut and Geoffrey I. Webb, editors. 2010. *TF–IDF*, pages 986–987. Springer US, Boston, MA.

Nisan Stiennon, Long Ouyang, Jeffrey Wu, Daniel Ziegler, Ryan Lowe, Chelsea Voss, Alec Radford, Dario Amodei, and Paul F Christiano. 2020. Learning to summarize with human feedback. *Advances in Neural Information Processing Systems*, 33:3008–3021.

Haitian Sun, William W Cohen, and Ruslan Salakhutdinov. 2021. Conditionalqa: A complex reading comprehension dataset with conditional answers. *arXiv preprint arXiv:2110.06884*.

Adam Trischler, Tong Wang, Xingdi Yuan, Justin Harris, Alessandro Sordoni, Philip Bachman, and Kaheer Suleman. 2017. NewsQA: A machine comprehension dataset. In *Proceedings of the 2nd Workshop on Representation Learning for NLP*, pages 191–200, Vancouver, Canada. Association for Computational Linguistics.

Ellen M. Voorhees and Dawn M. Tice. 2000. Building a question answering test collection. In *Proceedings of the 23rd Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, SIGIR '00, page 200–207, New York, NY, USA. Association for Computing Machinery.

Alex Wang, Kyunghyun Cho, and Mike Lewis. 2020. Asking and answering questions to evaluate the factual consistency of summaries. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 5008–5020, Online. Association for Computational Linguistics.

Thomas Wolf, Lysandre Debut, Victor Sanh, Julien Chaumond, Clement Delangue, Anthony Moi, Pierric Cistac, Tim Rault, Remi Louf, Morgan Funtowicz, Joe Davison, Sam Shleifer, Patrick von Platen, Clara Ma, Yacine Jernite, Julien Plu, Canwen Xu, Teven Le Scao, Sylvain Gugger, Mariama Drame, Quentin Lhoest, and Alexander Rush. 2020. Transformers: State-of-the-art natural language processing. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing: System Demonstrations*, pages 38–45, Online. Association for Computational Linguistics.

Zhilin Yang, Peng Qi, Saizheng Zhang, Yoshua Bengio, William Cohen, Ruslan Salakhutdinov, and Christopher D. Manning. 2018. HotpotQA: A dataset for diverse, explainable multi-hop question answering. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 2369–2380, Brussels, Belgium. Association for Computational Linguistics.

Ming Zhong, Da Yin, Tao Yu, Ahmad Zaidi, Mutethia Mutuma, Rahul Jha, Ahmed Hassan Awadallah, Asli Celikyilmaz, Yang Liu, Xipeng Qiu, and Dragomir Radev. 2021. QMSum: A new benchmark for query-based multi-domain meeting summarization. In *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 5905–5921, Online. Association for Computational Linguistics.

# Appendix

We now provide additional discussion of several aspects of this work.

## A Additional Details on the Annotation Procedure

We begin with an additional discussion of the annotation procedure.

**Construction of Context Paragraphs** As discussed in Section 3, in our annotation task, we supplement each disambiguation $(x_i, y_i)$ from AMBIGQA with a context passage $C_i$. Let us now describe the procedure used to construct these context passages.

For each disambiguation $(x_i, y_i)$, we execute the following three-stage procedure:

1. Among all paragraphs from Wikipedia pages $W$ visited by AMBIGQA annotators, we select those that contain $y_i$.
2. We compute TF-IDF similarity (Sammut and Webb, 2010) between the selected paragraphs and $x_i$.
3. If the highest similarity exceeds a certain empirically selected threshold, we use the corresponding paragraph as an additional context $C_i$ provided to annotators. Otherwise, we do not provide context for that disambiguation ($C_i = \emptyset$). The threshold was selected by the manual analysis of a subset of questions-context pairs. Our criteria was to avoid confusing (e.g., irrelevant) context paragraphs and we qualitatively selected the threshold according to this criteria.

Following this procedure, we were able to provide non-empty additional context passages for 45% of all disambiguations used in our annotation procedure.

**Instructions and Training** The instructions for our task are written along the lines of the four criteria we discussed in Section 3.1 and are provided in supplementary materials. In addition to the detailed instructions, we carefully design the training procedure to minimize the amount of noise in the annotations. In that, before being accepted to the main task, annotators go through the following three-step training procedure:

1. *Self-study session* First, we give annotators a short version of the instructions. They study them on their own and then annotate three sample questions.
2. *In-person session* Following the self-study session, we have an online video session in which we walk annotators through the full version of the instructions and discuss mistakes made in the self-study annotations.
3. *Exam session* Finally, annotators complete a five-question exam. We manually evaluate all the exam answers and share personal feedback with annotators.

In total, 27 annotators went through our training procedure and all of them were eventually accepted to work full-time on the main task. We note that the quality of answers in the self-study session was very diverse with some annotators making critical mistakes (e.g., not covering some of the disambiguations). However, the in-person session proved to be efficient in helping annotators to understand the requirements, leading to exam answers of consistently high quality.

**Quality Control and Feedback** Next, we discuss additional steps we took to help annotators in writing answers that satisfy the objectives formulated in Section 3.1. First, we added an automated check to our interface that warns annotators if any of the short answers $\{y_i\}_{i=1}^n$ is missing from their long-form answer. Annotators were able to override the warning if they believe that an equivalent formulation of the missing short answer is already included. For example, given two disambiguations with short answers "four seasons" and "4 seasons", annotators were instructed to use any of these two equivalent options.

Second, in addition to the carefully designed training procedure, we were also continuously monitoring the annotators' performance as they were going through the task. In that, we were giving regular constructive feedback that highlighted areas of improvement and pointed out mistakes identified in annotators' past answers. While we did not observe any significant decay in quality between the exam session and the main task annotation, we believe that continuous monitoring is crucial to avoid creating an incentive for annotators to reduce the amount of effort they put into the task.

Finally, to ensure that annotators did not have to guess when they met some situation not explained in the instructions, we maintained an FAQ document in which annotators could ask their questions

and receive an answer within a day. To support this mechanism, we allowed annotators to "park" an annotation task they were unsure about and return to it after they have their concerns resolved.

**Annotators' Well-Being** For this study, we recruited annotators who were fully dedicated to our task (8 hours a day for 5 days a week). To reduce the pressure on annotators and allow them to work at a comfortable pace, we gave annotators one hour to answer each question and recommended answering ten or more questions per day. On average, it took annotators 15 minutes to answer each question with the time consumption slightly decreasing as annotators get familiar with the task. The compensation rate for the task was set to be $17.8/hour which is higher than the minimum hourly wage in the US.

## B Additional Details on Modeling

In this section, we provide additional details on the modeling aspect of our evaluations.

**Input Format** Figures 4 and 5 provide schematic representations of inputs to the T5-O-K and ORACLE models, respectively. Bold black text represents tags that separate conceptually different parts of the input, text in blue is replaced with the instance-specific content in the actual training and evaluation data.

The input to T5-O-K is simpler and consists of two parts separated by the `context` tag: an ambiguous question and $K$ retrieved passages. Each retrieved passage consists of the `info` field that contains the retrieved passage and the `wikipage` field that displays the title of the source Wikipedia page. Retrieved passages are separated with the pipe symbol "|".

The input to the ORACLE model is more complex and has five parts:

- An ambiguous question $q$
- Short answers $\{y_i\}_{i=1}^{n}$ (`answers`)
- Disambiguated questions $\{x_i\}_{i=1}^{n}$ (`disambiguations`)
- Context paragraphs $\{C_i\}_{i=1}^{n}$ (`context1`)

$ambiguous_question **context: info:** $retrieved_passage_1 **wikipage:** $source_of_passage_1 **| ... | info:** $retrieved_passage_K **wikipage:** $source_of_passage_K

Figure 4: Input to the T5-O-K model.

- Additional knowledge pieces provided by the annotator $\{e_j\}_{j=1}^{m}$ (`context2`)

Similarly to the T5-O model, context paragraphs and additional knowledge pieces have `info` and `wikipage` fields, and the pipe symbol "|" is used to separate elements in the list.

$ambiguous_question **answers:** $short_answer_1 **| ... |** $short_answer_n **disambiguations:** $disambiguated_question_1 **| ... |** $disambiguated_question_n **context1: info:** $context_paragraph_1 **wikipage:** $source_of_context_1 **| ... | info:** $context_paragraph_n **wikipage:** $source_of_context_n **context2: info:** $additional_knowledge_1 **wikipage:** $source_of_knowledge_1 **| ... | info:** $additional_knowledge_m **wikipage:** $source_of_knowledge_m

Figure 5: Input to the ORACLE model.

**Parameter Choice** We use the context length of 512, 1024, and 2048 for the T5-O-1, T5-O-3, and T5-O-5 models, respectively. We use batch size of 8 across the three models. For T5-C, we use a batch size of 16 with a context length of 128.

## C Qualitative Analysis

To provide further insight into the importance of the generation aspect of our task, we conduct a manual analysis of the answers generated by the T5 open-book model. Our main observation is that even if the knowledge necessary to answer an ambiguous question *is successfully retrieved*, T5 often struggles to provide a high-quality answer. Table 6 demonstrates several characteristic mistakes that we identify.

**Hallucination** The first two rows of Table 6 demonstrate examples of hallucination in the T5-generated answers. In the first example, T5 hallucinates facts about the *2016 elections* (there were no elections in 2016) and about *the winner of the 2017 elections* (Rick Baker did not win the elections). In the second example, T5 starts with a wrong disambiguation (dragons do not marry people) and then mixes up facts about two characters from different books (*Daenerys Targaryen* and *Elizabeth/Liz Pennykettle*) into one.

**Question Misunderstanding** Another issue we observe in the T5-generated answers is that sometimes the answers provide a coherent story that is relevant to the question but does not answer it. This problem is illustrated in the third row of Table 6 where the question "*When was «under God» added to the Pledge of Allegiance?*" is answered with a

| Question and disambiguations | T5 output | JPR@5 relevant segments | Error analysis |
|---|---|---|---|
| *Q*: Who won the mayor race in st petersburg florida?<br><br>*Disambig-Q1*: Who won the 2017 mayor race in st petersburg florida?<br>Kriseman \| Rick Kriseman<br>*Disambig-Q2*: Who won the 2013 mayor race in st petersburg florida?<br>Kriseman \| Rick Kriseman<br>*Disambig-Q3*: Who won the 2009 mayor race in st petersburg florida?<br>Foster \| Bill Foster | the city of st. petersburg, florida has had several mayoral elections. democrat rick kriseman won the 2016 mayoral election, while republican former mayor rick baker did so in the 2017 mayoral election. kriseman defeated baker in the 2017 general election with 51.62% of the vote. | *2017 St. Petersburg, Florida mayoral election*: . . . Incumbent mayor Rick Kriseman, a Democrat, opted to seek re-election. His opponent in the runoff was Republican former mayor Rick Baker . . . Incumbent Rick Kriseman won the re-election. . .<br>*Rick Kriseman*: On November 7, 2017, Kriseman was re-elected for a second term, winning the general election with 51.62% of the vote. He defeated former Republican Mayor Rick Baker. . . | T5 hallucinates (i) the 2016 mayoral elections (there were no elections in 2016) and (ii) the winner of the 2017 mayoral elections (Kriseman won elections, but the model claims Baker to be the winner). |
| *Q*: Who was the mother of dragons married to?<br><br>*Disambig-Q1*: Who was the mother of dragons first married to?<br>Khal Drogo \| Drogo<br>*Disambig-Q2*: Who was the mother of dragons married to second?<br>Hizdahr zo Loraq<br>*Disambig-Q3*: What leader of Dothraki is the mother of dragons married to?<br>Khal Drogo<br>*Disambig-Q4*: What Ghiscari noble is the mother of dragons married to?<br>Hizdahr zo Loraq | dragons are often married to multiple people in a song of ice and fire storyline. in a song of ice and fire, the mother of dragons is known as elizabeth/liz pennykettle, a woman probably in her thirties who makes dragons out of clay and sells them at pottery fairs. she is the mother of lucy pennykettle, wife of the blinded ex-monk arthur, and landlady of david rain. elizabeth/liz pennykettle - a woman probably in her thirties who makes dragons out of clay and sells them at pottery fairs. she is the mother of lucy pennykettle, wife of the blinded ex-monk arthur, and landlady of david rain. | *A Song of Ice and Fire*: . . . Daenerys is married off by her elder brother Viserys Targaryen to . . .<br>*Daenerys Targaryen*: Viserys marries Daenerys to the powerful Dothraki warlord Khal Drogo in exchange for his military support . . .<br>*The Last Dragon Chronicles*: Elizabeth/Liz Pennykettle – A woman probably in her thirties who makes dragons out of clay and sells them at pottery fairs. | In the first sentence of the answer, T5 hallucinates that dragons are married to humans in the *A Song of Ice and Fire* storyline. In the second sentence, T5 suddenly jumps from a discussion of *A Song of Ice and Fire* to *The Last Dragon Chronicles*, thereby misleading the reader.<br><br>The blue text highlights repeated content. |
| *Q*: When was under god added to the pledge of alligence?<br><br>*Disambig-Q1*: When was under god officially added to the pledge of alligence?<br>June 14, 1954<br>*Disambig-Q2*: When did Bowman popularly introduce the pledge of alligence with under god added to a society meeting in Chicago?<br>February 12, 1948<br>*Disambig-Q3*: When was the bill signed into law that under god was added to the pledge of Allegiance?<br>Flag Day \| June 14, 1954<br>*Disambig-Q4*: When was under god added to the pledge of Allegiance for the first time?<br>February 12, 1948<br>*Disambig-Q5*: When did the board of directors of the Knights of Columbus adopt a resolution to add under god to the pledge of Allegiance?<br>April 30, 1951 | the pledge of allegiance of the united states is an expression of allegiance to the flag of the united states and the republic of the united states of america. it was originally composed by captain george thatcher balch, a union army officer during the civil war and later a teacher of patriotism in new york city schools. the form of the pledge used today was largely devised by francis bellamy in 1892, and formally adopted by congress as the pledge in 1942. the most recent alteration of its wording came on flag day, june 14, 1954. [Missing mention of "under god" words] | *Pledge of Allegiance* Louis Albert Bowman, an attorney from Illinois, was the first to suggest the addition of "under God" to the pledge . . . At a meeting on February 12, 1948, he led the society in reciting . . .<br>*Pledge of Allegiance* In 1951, the Knights of Columbus, the world's largest Catholic fraternal service organization, also began including the words "under God" in the Pledge of Allegiance.<br>*Pledge of Allegiance* Congress passed the necessary legislation and Eisenhower signed the bill into law on Flag Day, June 14, 1954. Eisenhower said: The phrase "under God" was incorporated into the Pledge of Allegiance on June 14, 1954, by a Joint Resolution of Congress amending § 4 of the Flag Code enacted in 1942. | The T5 output introduces the Pledge of Allegiance and mentions some of the right dates (June 14, 1954), but does not mention that alteration on June 14, 1954, included the words "under god" to the Pledge. |

Table 6: Error analysis for T5-O-5. The colored text highlights problematic parts of the T5 output.

history of the Pledge of Allegiance but does not mention the target phrase («under God»).

**Repetitions** Finally, we observe a somewhat technical issue of repetitions in the generated answers, as shown in the second row of Table 6.