Model Analysis & Evaluation for Ambiguous Question Answering

Konstantinos Papakostas[*] [†]
University of Amsterdam

Irene Papadopoulou[*]
University of Amsterdam

Abstract

Ambiguous questions are a challenge for Question Answering models, as they require answers that cover multiple interpretations of the original query. To this end, these models are required to generate long-form answers that often combine conflicting pieces of information. Although recent advances in the field have shown strong capabilities in generating fluent responses, certain research questions remain unanswered. Does model/data scaling improve the answers' quality? Do automated metrics align with human judgment? To what extent do these models ground their answers in evidence? In this study, we aim to thoroughly investigate these aspects, and provide valuable insights into the limitations of the current approaches. To aid in reproducibility and further extension of our work, we open-source our code here.

1 Introduction

Question Answering (QA) has been subject to great progress in the past years, largely thanks to the representational capabilities of modern architectures like the Transformer (Vaswani et al., 2017), but also due to the curation of large, high-quality datasets that enabled the effective training of these models (Joshi et al., 2017; Kwiatkowski et al., 2019).

At the same time, the presence of *ambiguous questions* has been a challenging aspect of QA. In order to answer such questions, models are required to generate long answers with fluency and cohesion, which is often referred to in the literature as Long-Form Question Answering (LFQA). To tackle this, Min et al. (2020) curated the `AmbigQA` dataset, which contains disambiguations for various questions that were present in popular benchmarks. Extending this work, Stelmakh et al. (2022) selected a subset of the questions and crowd-sourced gold answers that cover all possible interpretations of each question, resulting in the `ASQA` dataset.

Recently, Krishna et al. (2021) performed a case study on `ELI5` (Fan et al., 2019), one of the largest collections available for LFQA, and pointed out several issues that complicate the development and evaluation of suitable models. In particular, the authors questioned whether the retrieved documents are considered by the generative models when producing an answer, and the correlation of the common evaluation metrics with human judgment.

In this work, we aim to investigate whether the baselines set on `ASQA` by Stelmakh et al. (2022) suffer from the issues pointed out by Krishna et al. (2021), but also to analyze the modeling choices that contribute to performance. Concretely, we set out to answer the following research questions:

RQ1 Does scaling the size of the generative models affect the quality of the generated answers?

RQ2 Can an intermediate round of fine-tuning on non-ambiguous LFQA collections improve performance in ambiguous QA?

RQ3 When comparing models head-to-head, does human judgment reflect the difference in the automated evaluation metrics?

RQ4 Do models base their answers on the retrieved evidence, or could they be hallucinating?

2 Methodology

We design a standard retrieval-augmented system, to identify the dimensions that contribute to disambiguating questions and generating factual answers.

2.1 The LFQA Pipeline

Evidence Retrieval The first step of the pipeline is to identify the documents that will form the basis of the generated answers. Given a question q, we employ a retrieval method R that fetches the top-k relevant documents $\{d_i\}_{i=1}^{k}$. For a document d_i to be considered relevant, it needs to cover at least

[*] Equal contributions
[†] Correspondence email: `dinos.ppk@gmail.com`

Findings of the Association for Computational Linguistics: ACL 2023, pages 4570–4580
July 9-14, 2023 ©2023 Association for Computational Linguistics

one aspect of q. To completely resolve the ambiguity, the passages[1] in the index should suffice to collectively answer all aspects of a given question.

Answer Generation Once the evidence has been collected, we feed the retrieved passages to a generative model to summarize them in a concise answer that disambiguates the question at hand. As is the standard practice in contemporary literature, we opt for a sequence-to-sequence model G that follows an encoder-decoder architecture, which first creates a dense representation of the concatenation of the question and the passages, and then produces the answer by attending to this latent representation:

$$\text{answer} = G([q; d_1; \ldots; d_k])$$

with $[\cdot\,;\cdot]$ being the concatenation of ≥ 2 passages.

2.2 Modeling Choices

We expect more sophisticated pipelines to provide better answers for ambiguous questions, and thus we make a distinction based on the complexity of the interaction between the two components.

Naive We implement the QUESTION baseline, which repeats the ambiguous question a few times in order to match the typical length of the answers in the dataset. This is a lower bound on the task, as we are not truly answering the question at hand.

Retrieval-Only In this case, we rely exclusively on a retriever to fetch the top-k passages as a response to the ambiguous question. We experiment with different values of k to evaluate whether using more passages leads to an answer that covers more of the disambiguated questions.[2]

Sequence-to-Sequence A generative language model is often used to produce a concise response that summarizes all of the disambiguating answers. We analyze three scenarios:

- Closed Book: In the most extreme approach, we assume that the model is not conditioned on the results of a retriever, but rather only on the question itself, and relies on its parametric knowledge (Roberts et al., 2020) to respond. We expect this to significantly harm performance, as the available context to provide an accurate answer is limited.
- Random Retrieval: In order to verify whether the model *grounds* its answers on the retrieved documents (Krishna et al., 2021), we design a controlled scenario where we randomly sample passages from our index as evidence.
- Open Book: In the most realistic setting, the model treats the top-k results of a retriever as context to respond appropriately. We expect stronger retrieval methods to lead to more comprehensive answers, as the generative model will be conditioned on more diverse and relevant information.

3 Experimental Setup

We aim to assess whether LFQA systems can generate concise answers that disambiguate the provided questions. In this section, we present the datasets and models used, as well as the evaluation metrics.

3.1 Datasets

We use the ASQA dataset to train and evaluate our systems. It is a subset of the AmbigQA dataset, with long-form answers and additional context for each of the selected samples. More specifically, it contains 6,316 ambiguous questions, with each one being paired to a set of disambiguated questions, the corresponding short answers, and the Wikipedia passages where the answers were found. For each question, the dataset curators crowd-sourced a long-form answer that resolves the ambiguity by summarizing all short answers. The annotators provided one reference answer for all train (4,353) samples, and two for all dev (948) and test (1,015) samples.[3]

Although ASQA is a useful resource for LFQA, when going through the dataset for preliminary analysis, we found cases where the ambiguity was to identify when "last/this year" refers to. We argue that training a model on such samples is counterintuitive, as we generally assume that the information is coming from a fixed snapshot of a knowledge base. We provide a few examples that cover similar types of questions in Appendix A.

Additionally, given the limited size of the ASQA dataset, we follow one of the proposed research directions posed by Stelmakh et al. (2022) and investigate the impact of intermediate fine-tuning on a larger LFQA collection, before training on ASQA. In

[1] We use the terms document and passage interchangeably.

[2] Employing some form of result diversification has the potential to improve the performance of the retrieval component, but we leave this direction as future work.

[3] An illustration of ambiguity can be observed in a seemingly straightforward query like "Who was the ruler in France in 1830?", which presents a challenge due to the existence of two rulers during that period.

particular, we use a processed version of the ELI5 dataset[4] that addresses some of the issues raised by Krishna et al. (2021) (226,147 train / 3,020 dev samples), as well as the NLGEN set of the MS MARCO QA dataset (153,725 train / 12,467 dev samples).

3.2 Models

The typical QA pipeline comprises a retrieval and a generative model, which can be pre-trained separately and then fine-tuned on the downstream task. Although this can be done in an end-to-end fashion, we chose to keep the retriever frozen, to avoid re-indexing of the support passages during training, and only train the generative model. We provide detailed training information in Appendix B.

Retrieval Models We experiment with both sparse (lexical) and dense (neural) methods, in order to investigate whether the type of the question encoder has an impact on the relevancy of the retrieved passages. We chose BM25 (Robertson et al., 1995) for the former, and DPR (Karpukhin et al., 2020) for the latter, as they were both supported by the Pyserini (Lin et al., 2021) toolkit and they constitute two of the most explored options in their corresponding fields. In both cases, we use the prebuilt indices of Wikipedia provided by Pyserini to have a common knowledge base that matches the one used by the dataset curators.

Generative Models We use two of the most popular Transformer-based encoder-decoder models, namely T5 (Raffel et al., 2019) and BART (Lewis et al., 2020). More specifically, we experiment with three variants of these models, in increasing parameter count: BART-base (140M params), T5-base (220M params), and BART-large (400M params). By doing so, we aim to verify whether increasing the capacity of the generative model corresponds to an increase in answer quality.

3.3 Metrics

Automated Evaluation Evaluating the performance of generative models is one of the most challenging aspects for LFQA. For recall-oriented QA systems, the most common metric used is ROUGE-L (Lin, 2004), which identifies the longest common sub-sequence between the generated answer and a reference (gold) answer. Sequences are penalized proportionally to their length to prevent generating a longer output to artificially increase the overlap with the reference text. However, a recent study by Krishna et al. (2021) revealed that ROUGE-L did not always correlate with human judgment, pointing to the need for a more diverse evaluation setup. Specifically for ambiguous LFQA, Stelmakh et al. (2022) proposed two metrics to quantify the disambiguation ability of the model:

1. **STR-EM** (String Exact Match): the fraction of disambiguated answers that the model includes verbatim in its output.

2. **DISAMBIG-F1**: the fraction of answers that can be deduced with a text comprehension model, using the predicted long answer and the disambiguated question:

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

where N is the number of evaluation samples, $n^{(k)}$ the number of disambiguations for the k-th question, $y_i^{(k)}$ its i-th ground-truth short answer, $\hat{y}_i^{(k)}$ the predicted short answer[5], and ϕ a function that computes the token-level F1 score between them.

Finally, they define another metric, namely **DR**, which is the geometric mean of DISAMBIG-F1 and ROUGE-L, as an overall estimate of the performance in both disambiguation and answer overlap.

Human Evaluation To better align the model's performance with the overall satisfaction of the human users that interact with the system, we follow Stelmakh et al. (2022) in creating an anthropocentric evaluation pipeline. Our approach differs in that it performs *head-to-head comparisons* between different models, aiming to draw conclusions about the design choices that lead to a well-performing LFQA system. For a pair of answers, we compare their *comprehensiveness* (COMP); whether the answer suffices to understand both the source of ambiguity in the question, and the relation between the individual answers, their *fluency* (FLUE); whether the answer is coherent and fluent from a human reading stance, and the *overall human impression* (OVER); which of the answers is prefered overall.

4 Results

Automated Evaluation We evaluate our models on the development set of ASQA using the au-

[4] Available on HuggingFace: lfqa & lfqa_support_docs.

[5] Generated from a RoBERTa (Liu et al., 2019) model trained on SQUAD2.0 (Rajpurkar et al., 2018), with the predicted long answer and the disambiguated question as input.

	ANSWER LENGTH	ROUGE-L	STR-EM	DISAMBIG-F1	DR
QUESTION	71.6	15.3	1.2	0.1	1.4
T5-base CLOSED BOOK	38.1	30.7	3.7	2.7	9.1
BART-base CLOSED BOOK	44.5	31.5	3.9	2.8	9.3
BART-large CLOSED BOOK	50.2	33.4	7.1	4.5	12.2
BM25@1,3,5	103.7 / 310.9 / 518.0	28.6 / 20.3 / 15.2	18.6 / 30.0 / 36.4	10.6 / 14.7 / 17.4	**17.4** / 17.3 / 16.2
DPR@1,3,5	103.5 / 310.4 / 517.3	31.4 / 22.1 / 16.4	29.3 / 44.0 / 50.8	17.5 / 22.8 / 25.7	**23.5** / 22.5 / 20.5
T5-base DPR@1,3,5	51.4 / 57.8 / 57.8	31.6 / 33.7 / 33.9	23.9 / 26.4 / 26.2	17.6 / 17.8 / 17.9	23.6 / 24.5 / **24.6**
T5-base NLGEN DPR@1,3,5	48.5 / 56.0 / 56.0	31.2 / 33.7 / 33.7	24.6 / 26.0 / 26.7	16.8 / 18.1 / 17.8	22.9 / **24.7** / 24.5
BART-base DPR@1,3,5	52.4 / 57.1 / 56.7	33.1 / 33.9 / 33.9	24.2 / 25.1 / 24.5	16.1 / 16.5 / 16.4	23.1 / **23.7** / 23.5
BART-large DPR@1,3,5	54.7 / 62.5 / 63.3	34.2 / 36.4 / 36.6	26.0 / 30.0 / 29.8	18.1 / 20.8 / 20.5	24.9 / **27.5** / 27.4
BART-large ELI5 DPR@1,3,5	55.2 / 59.7 / 59.9	35.1 / 36.6 / 37.0	26.7 / 30.3 / 29.7	19.0 / 21.0 / 21.0	25.8 / **27.7** / 27.6

Table 1: Performance comparison on the development set of ASQA, using the metrics described in Section 3.3. For retrieval-augmented models, @1, 3, 5 indicates using $1/3/5$ retrieved passages as evidence. **Bold** scores indicate best result among different number of retrieved passages, underlined scores indicate best result in each setup.

tomated metrics introduced in Section 3.3. Table 1 displays the results for the naive baseline, the retrieval-only experiments using both BM25 and DPR, and the closed/open book experiments with DPR. In cases where we use a retriever, we fetch the top-k relevant documents, for $k \in \{1, 3, 5\}$. As the level of information contained in the retrieved passages has a direct impact on the answer quality, we perform a short study on the upper bound of retrieval in Appendix C.

Naturally, the naive QUESTION baseline performs the worst, with a ROUGE-L score of 15.3 and a DISAMBIG-F1 score of 0.1, leading to an overall DR score of 1.4. Focusing on the retrieval-only experiments, we observe that DPR consistently outperforms BM25 in all metrics, which is anticipated as semantic matching allows us to retrieve passages that answer different aspects of the question. Remarkably, although the closed-book variants surpass the retrieval-only methods in terms of ROUGE-L, we see the opposite trend for all other metrics. The open-book variants outperform the rest, confirming our hypothesis that augmenting the generative model with a retriever is crucial for performance. We also notice that the increase in performance closely follows the growth in parameter count, implying that **[RQ1]** model scaling improves the quality of the answer.[6] Surprisingly, the impact of fine-tuning on larger collections on performance is marginal, as the disambiguation metrics only narrowly improve. For completeness, we report the intermediate fine-tuning results for the closed-book experiments in Appendix D. In general, we deem that **[RQ2]** naively fine-tuning on non-ambiguous LFQA datasets has limited added value, as the models are not inclined to identify any uncertainty in the meaning of the questions asked.

	COMP	FLUE	OVER
T5-base vs **BART-base***	50%	38%	50%
BART-large vs **BART-base***	61%	61%	61%
BART-large ELI5 vs **BART-large***	61%	50%	61%
BART-large: DPR@3 vs **DPR@1**	75%	50%	75%
DPR@1 vs **BART-base DPR@1**	61%	38%	50%
BART-large DPR@1 vs **DPR@1**	38%	50%	75%

Table 2: Head-to-head comparison using human evaluation metrics. Pairs with * use DPR@3 for retrieval. Bars indicate the annotators' preference for either model.

Human Evaluation We perform a head-to-head comparison between a selection of approaches to determine which modeling aspects contribute more/less to the performance. For each pair, we asked 2 assessors to compare the models *blindly* using the human evaluation metrics defined in Section 3.3. We report our results in Table 2, where we observe that most comparisons follow the trends we described previously using the automated metrics. Noticeably, when comparing T5-base DPR@3 vs BART-base DPR@3 and DPR@1 vs BART-base DPR@1, human annotators seem to equally prefer both models overall. However, the difference in DR for these pairs is merely a few decimal points, which justifies the inability to distinguish between them. Hence, we confirm that **[RQ3]** the *overall human impression* aligns with automated metrics.

Random Retrieval Finally, we evaluate whether the generated answers are grounded on the retrieved passages. Table 3 showcases the performance for the models used in our open-book experiments, and we generally see a *decrease* in performance when compared to fetching relevant documents. Interestingly, BART-base with random re-

[6] To gain further insight into some of the common errors, we showcase a few example model outputs in Appendix E.

	ROUGE-L	STR-EM	DIS-F1	DR
T5-base	20.5	2.0	0.7	3.7
T5-base NLGEN	20.8	1.6	0.8	4.1
BART-base	29.8	5.5	3.5	10.3
BART-large	24.3	4.4	2.2	7.2
BART-large ELI5	29.6	4.8	3.0	9.5

Table 3: Random Retrieval performance. We use $k = 3$ random passages from the DPR index as evidence.

trieval performs better compared to its closed-book variant. As this is the smallest model we tested, we speculate that for LFQA, under-parameterized architectures may be relying more on the question and not the corresponding evidence. Despite this, **[RQ4]** models for ambiguous LFQA appear to be well grounded in the retrieved evidence overall.

5 Conclusion

In conclusion, by exploring the LFQA field in answering ambiguous questions, we find that the ASQA dataset is a good foundation to develop and evaluate models. We notice that larger generative models produce better answers, with semantic matching for retrieval having a positive impact. To compensate for the small size of the dataset, we experiment with intermediate fine-tuning on larger collections and find that doing so only marginally improves the results. By comparing the performance when using relevant versus random documents, we show the models' dependency on the provided context. Our human evaluation confirms the general trends we observed using the automated metrics, giving credit to their disambiguating ability.

Limitations

We see two main limitations in our study. Primarily, given the clear trend of larger generative models producing higher quality answers, an obvious question is to investigate whether this continues to be the case indefinitely, or whether it saturates after a critical amount of parameters. Despite this, due to hardware restrictions, we were unable to experiment with models larger than BART-large. Additionally, considering that the field of ambiguous QA inherently requires complementary pieces of evidence, there is no doubt that diversification methods are bound to yield better results in terms of disambiguation quality. In this work, however, we limited ourselves to using a typical neural retriever, shifting our focus toward the factuality and the fluency of the generated answers.

Ethics Statement

As we use a publicly available Wikipedia index as our knowledge base, it is possible that the generated answers may contain some form of bias that reflects the information submitted by the anonymous editors of the website. To prevent this, follow-up work could examine how to detect misinformation or hate speech in indexed passages, before using them as evidence for the generative models.

Acknowledgements

This work was carried out on the Dutch national e-infrastructure with the support of the SURF Cooperative. We would also like to thank Evangelos Kanoulas for his valuable feedback and guidance during this project.

References

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.

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.

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.

Mike Lewis, Yinhan Liu, Naman Goyal, Marjan Ghazvininejad, Abdelrahman Mohamed, Omer Levy, Veselin Stoyanov, and Luke Zettlemoyer. 2020. BART: Denoising sequence-to-sequence pre-training for natural language generation, translation, and comprehension. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 7871–7880, Online. Association for Computational Linguistics.

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, Xueguang Ma, Sheng-Chieh Lin, Jheng-Hong Yang, Ronak Pradeep, and Rodrigo Nogueira. 2021. Pyserini: A Python Toolkit for Reproducible Information Retrieval Research with Sparse and Dense Representations. In *Proceedings of the 44th International ACM SIGIR Conference on Research and Development in Information Retrieval*, SIGIR '21, page 2356–2362, New York, NY, USA. Association for Computing Machinery.

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. *arXiv preprint arXiv:1907.11692*.

Ilya Loshchilov and Frank Hutter. 2017. Decoupled Weight Decay Regularization. *arXiv preprint arXiv:1711.05101*.

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.

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.

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.

Stephen Robertson, S. Walker, S. Jones, M. M. Hancock-Beaulieu, and M. Gatford. 1995. Okapi at TREC-3. In *Overview of the Third Text REtrieval Conference (TREC-3)*, pages 109–126. Gaithersburg, MD: NIST.

Ivan Stelmakh, Yi Luan, Bhuwan Dhingra, and Ming-Wei Chang. 2022. ASQA: Factoid Questions Meet Long-Form Answers. In *Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing*, pages 8273–8288, Abu Dhabi, United Arab Emirates. Association for Computational Linguistics.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Ł ukasz Kaiser, and Illia Polosukhin. 2017. Attention is All you Need. In *Advances in Neural Information Processing Systems*, volume 30. Curran Associates, Inc.

A Potential Issues with ASQA

During our experiments, we notice certain issues regarding the ASQA dataset. In particular, we observed that in some cases the ambiguity in the question is pedantic. For instance, there are questions where a date is not specified (e.g., "What kind of car won the Daytona 500 this year?"), and in a typical scenario, the system would assume that the question refers to the current year. Instead, in ASQA this type of question is considered ambiguous, and the short answers resolve the ambiguity by reformulating the question for different years (e.g., 2017, 2016, 2015, etc.). In addition, we argue that some of the disambiguated questions and their short answers are too specific. This results in the model being penalized for not generating the exact correct terms, even though its answer semantically covers some part of that interpretation (e.g., the last example in Table 6).

B Implementation Details

We use the official T5-base, BART-base, and BART-large implementations from HuggingFace. We train each model for 20 epochs on the ASQA[7] dataset with the AdamW optimizer (Loshchilov and Hutter, 2017), using a weight decay of 0.01, and a learning rate of 10^{-5} for T5 and $5 \cdot 10^{-6}$ for BART. Training is stopped early if the validation loss doesn't decrease after 5 epochs. We use a train batch size of 8 for open-book experiments, and 16 for closed-book, with an evaluation batch size of 8 in both cases. We train our models on one NVIDIA Titan RTX GPU, and use 16-bit mixed precision to accelerate training. All of the models converged within ~30 minutes of training.

For our intermediate fine-tuning experiments, we first train T5-base on NLGEN subset[8] of the MS MARCO dataset for 1 epoch with a learning rate of 10^{-4} and then continue training on ASQA as described above. For BART, we use a publicly available instance from HuggingFace that has been pretrained on ELI5[9], and continue training on ASQA.

Finally, we use beam decoding with 5 beams and a max sequence length of 100 tokens. We force the model to not repeat the same trigram in its output by using the option no_repeat_ngram_size=3 when generating answers.

C Upper Bound for Retrieval

To investigate the efficiency of our retrieval models, we perform an analysis for the upper bound of the retrieved passages' relevancy to the corresponding gold answers. In particular, for each of the systems used, we count the number of relevant short answers retrieved using an exact string match with the dataset's gold answers. In Table 4, we notice that even the best retrieval setup at our disposal, namely **DPR@5**, only fetches $\sim 44\%$ of the relevant answers on average or up to 58% in cases where it identifies at least one relevant passage. This emphasizes the need for a multi-hop retrieval system like JPR (Min et al., 2021) in order to fully utilize the power of the first component of the pipeline. It is evident that a higher upper bound for retrieval will increase the overall performance of the systems that tackle ambiguous LFQA, making it a high priority for follow-up work.

	Avg # of short answers retrieved	
	In all results	In results with ≥ 1 correct answer
BM25@1	0.52 (14.71%)	1.61 (45.44%)
BM25@3	0.89 (24.10%)	1.83 (49.35%)
BM25@5	1.11 (29.81%)	1.96 (52.53%)
DPR@1	0.89 (24.70%)	1.74 (48.38%)
DPR@3	1.39 (37.73%)	2.00 (54.61%)
DPR@5	1.62 (43.79%)	2.15 (58.15%)

Table 4: Number of relevant chunks of evidence identified using different retrieval systems. This constitutes an upper bound to the generative model's ability to answer the various disambiguated questions.

D Effect of Intermediate Fine-Tuning for the Closed Book Experiments

Table 5 displays the impact of the intermediate fine-tuning on a larger LFQA collection in the closed-book setting, using automated evaluation metrics. Similarly to the open-book setting, **BART-large ELI5** performs the best, which confirms the hypothesis that larger models are able to benefit the most when given more training data.

	ROUGE-L	STR-EM	DIS-F1	DR
T5-base	30.7	3.7	2.7	9.1
T5-base NLGEN	30.0	2.9	2.8	9.2
BART-base	31.5	3.9	2.8	9.3
BART-large	33.4	7.1	4.5	12.2
BART-large ELI5	34.1	7.7	4.8	12.8

Table 5: Closed-book performance, showcasing the effect of intermediate fine-tuning on a large LFQA dataset.

[7] https://huggingface.co/datasets/din0s/asqa

[8] https://huggingface.co/datasets/din0s/msmarco-nlgen

[9] https://huggingface.co/vblagoje/bart_lfqa

E Answer Demonstration

We demonstrate a brief selection of answers generated by our best-performing model (**BART-large** `ELI5`) in Table 6. We observe that the main reason for our model's inability to disambiguate all parts of the question is that the passages retrieved with DPR simply do not cover all possible interpretations. In most cases, the first passage retrieved suffices to answer at least one of the disambiguated questions, and the rest passages are either not relevant or answer the same question again. Contrary to the results of Stelmakh et al. (2022), our model does not hallucinate or repeat itself. We hypothesize that since their model (**T5-large**) is almost twice as big as **BART-large** in terms of parameter count, it can better utilize its context but at the same time be overconfident in its answers.

Question, provided disambiguations, and answers	Generated model output: BART-large `ELI5` DPR@3	DPR@3 retrieved passages	Answer analysis
Q: Who was the ruler of France in 1830? *Q1*: Who was the ruler of France until 2 August 1830? *A1a*: Charles Philippe *A1b*: Charles X *Q2*: Who was the ruler of France after 9 August 1830? *A2*: Louis-Philippe I	the french revolution of 1830, also known as the july revolution or in french, the second french revolution, led to the overthrow of king charles x, the french bourbon monarch, and the ascent of his cousin louis philippe, duke of orléans, who himself, after 18 precarious years on the throne, would be overthrown in 1848.	***July Revolution*** The French Revolution of 1830, also known as the July Revolution ("révolution de Juillet"), Second French Revolution or in French ("Three Glorious [Days]"), led to the overthrow of King Charles X, the French Bourbon monarch, and the ascent of his cousin Louis Philippe, Duke of Orléans, who himself, after 18 precarious years on the throne, would be overthrown in 1848.. ***July Monarchy*** 1840, to general indifference. The publicś attention was concentrated on the trial of Marie Lafarge.. ***July Revolution*** ..led to the overthrow of King Charles X, the French Bourbon monarch, and the ascent of his cousin Louis Philippe, Duke of Orléans, who himself..	We do not notice any hallucination or repetition in this example. The model manages to correctly answer the ambiguous question with fluency and coherency. In addition, through reading the answer we understand why the question asked was ambiguous.
Q: The repetition of phrases or sentences with similar structures or meanings is called? *Q1*:The repetition of phrases or sentences with similar structures is called? *A1a*: parallel construction *A1b*: parallelism *A1c*: Parallel syntax *Q2*: The repetition of phrases or sentences with similar meanings is called? *A2*: Schesis onomaton	parallelism, also known as parallel structure or parallel construction, is a balance within one or more sentences of similar phrases or clauses that have the same grammatical structure. the application of parallelism affects readability and may make texts easier to process.	***Parallelism (grammar)*** In grammar, parallelism, also known as parallel structure or parallel construction, is a balance within one or more sentences of similar phrases or clauses that have the same grammatical structure.. ***Metathesis (linguistics)*** Metathesis (; from Greek , from "I put in a different order"; Latin: "trānspositiō") is the transposition.. ***Resultative*** Resultative In linguistics, a resultative (abbreviated) is a form that expresses that something or someone has undergone a change in state as the result of the completion of an event..	In this example, the model only manages to get one of the two answers that disambiguate the question. We do not observe any hallucinations or repetition.
Q: When does callie come in grey's anatomy? *Q1*: What date does Callie Torres first appear in Grey's Anatomy? *A1*: February 26, 2006 *Q2*: Which episode does Callie Torres first appear in Grey's Anatomy? *A2*: "What Have I Done to Deserve This?" (2.19) *Q3*: What date does Callie Torres start to appear in Grey's Anatomy as a series regular? *A3*: September 21, 2006 *Q4*: What episode Callie Torres start to appear in Grey's Anatomy as a series regular? *A4*: "Time Has Come Today" (3.01)	grey's anatomy is an american medical drama television series created by shonda rhimes, which airs on the abc in the united states. the character callie torres was introduced in season two, as a senior orthopedic resident as a love interest for intern george o'malley (t.r. knight). she was originally contracted to appear on a recurring basis, but received star billing in the third season. at the end of the show's twelfth season, ramirez	***Callie Torres*** ..."Callie" Torres, M.D. is a fictional character from the medical drama television series "Greyś Anatomy"... She was introduced in season two, as a senior orthopedic resident, as a love interest for intern George OḾalley (T.R. Knight). Eventually becoming an attending orthopedic surgeon, the character was originally contracted to appear on a recurring basis, but received star billing.. ***Callie Torres*** ..Eventually becoming an attending orthopedic surgeon, the character was originally contracted to appear on a recurring basis, but received star billing in the third season..' ***Callie Torres*** Ramirez was nominated for several awards for her portrayal of Torres, including the..	Here the model is not able to get any of the answers correctly. However, one could argue that two of the four disambiguated questions were partially answered since the model outputs that the character was introduced in season two and became a series regular in the third season. We also do not notice any hallucinations or repetition.

Table 6: Qualitative analysis. Green/orange text highlights correct/partially-correct parts of the answer.

ACL 2023 Responsible NLP Checklist

A For every submission:

☑ A1. Did you describe the limitations of your work?
Yes, after the conclusion (Section 5)

☑ A2. Did you discuss any potential risks of your work?
Yes, in the ethics statement

☑ A3. Do the abstract and introduction summarize the paper's main claims?
Yes, in Section 1

☒ A4. Have you used AI writing assistants when working on this paper?
Left blank.

B ☑ Did you use or create scientific artifacts?

Yes, Sections 3.1 and 3.2

☑ B1. Did you cite the creators of artifacts you used?
Yes, Sections 3.1 and 3.2

☒ B2. Did you discuss the license or terms for use and / or distribution of any artifacts?
We assume models and datasets on HuggingFace have a permissive license to use for research

☑ B3. Did you discuss if your use of existing artifact(s) was consistent with their intended use, provided that it was specified? For the artifacts you create, do you specify intended use and whether that is compatible with the original access conditions (in particular, derivatives of data accessed for research purposes should not be used outside of research contexts)?
We discuss possible misuses of our models in the Ethics section, and we use artifacts purely for research purposes, without discussing it however in the paper's text

☒ B4. Did you discuss the steps taken to check whether the data that was collected / used contains any information that names or uniquely identifies individual people or offensive content, and the steps taken to protect / anonymize it?
No, but we mention this as a follow-up work in the Ethics statement

☒ B5. Did you provide documentation of the artifacts, e.g., coverage of domains, languages, and linguistic phenomena, demographic groups represented, etc.?
We use English-only datasets, but the original authors of these works have not specified such information

☑ B6. Did you report relevant statistics like the number of examples, details of train / test / dev splits, etc. for the data that you used / created? Even for commonly-used benchmark datasets, include the number of examples in train / validation / test splits, as these provide necessary context for a reader to understand experimental results. For example, small differences in accuracy on large test sets may be significant, while on small test sets they may not be.
Yes, Section 3.1

C ☑ Did you run computational experiments?

Section 4

☑ C1. Did you report the number of parameters in the models used, the total computational budget (e.g., GPU hours), and computing infrastructure used?
Yes, Appendix B

The Responsible NLP Checklist used at ACL 2023 is adopted from NAACL 2022, with the addition of a question on AI writing assistance.

☑ C2. Did you discuss the experimental setup, including hyperparameter search and best-found hyperparameter values?
Yes, Appendix B

☒ C3. Did you report descriptive statistics about your results (e.g., error bars around results, summary statistics from sets of experiments), and is it transparent whether you are reporting the max, mean, etc. or just a single run?
No, we ran our experiments for a single seed due to computational budget limitations

☒ C4. If you used existing packages (e.g., for preprocessing, for normalization, or for evaluation), did you report the implementation, model, and parameter settings used (e.g., NLTK, Spacy, ROUGE, etc.)?
No, but we use the default settings for the packages used

D ☑ Did you use human annotators (e.g., crowdworkers) or research with human participants?

Yes, Section 4

☒ D1. Did you report the full text of instructions given to participants, including e.g., screenshots, disclaimers of any risks to participants or annotators, etc.?
No, the instructions were given orally as it was a small cohort

☒ D2. Did you report information about how you recruited (e.g., crowdsourcing platform, students) and paid participants, and discuss if such payment is adequate given the participants' demographic (e.g., country of residence)?
No, but the assessors were fellow classmates who volunteered to do it

☒ D3. Did you discuss whether and how consent was obtained from people whose data you're using/curating? For example, if you collected data via crowdsourcing, did your instructions to crowdworkers explain how the data would be used?
No, but they were fully aware of the purposes of this research

□ D4. Was the data collection protocol approved (or determined exempt) by an ethics review board?
Not applicable. Left blank.

☒ D5. Did you report the basic demographic and geographic characteristics of the annotator population that is the source of the data?
No, they are Master's students from Europe, both male and female