

Fleece QA: Self-Instruct Generation for Question Answering Dataset from Large Corpus

Yueheng Zhang



Electrical Engineering and Computer Sciences
University of California, Berkeley

Technical Report No. UCB/EECS-2024-116

<http://www2.eecs.berkeley.edu/Pubs/TechRpts/2024/EECS-2024-116.html>

May 17, 2024

Copyright © 2024, by the author(s).
All rights reserved.

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. To copy otherwise, to republish, to post on servers or to redistribute to lists, requires prior specific permission.

Acknowledgement

I would like to extend my deepest gratitude to Professor Dawn Song for this great opportunity and guidance. Furthermore, I am grateful to Xiaoyuan Liu for his continuous support and strategic guidance throughout the project. Special thanks to Tianneng Shi for his invaluable assistance in debugging critical parts of our code, which was essential for the progression of our experiments.

I would like to acknowledge Berkeley RDI for providing access to the necessary GPU compute resources, without which this research would not have been possible. Additionally, we appreciate the general support and helpful discussions provided by Yu Gai, Siyuan Zhuang, Zhe Ye, and Zhun Wang, whose contributions have enriched our work.

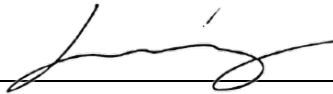
**Fleece QA: Self-Instruct Generation for Question Answering Dataset
from Large Corpus**
by Yueheng Zhang

Research Project

Submitted to the Department of Electrical Engineering and Computer Sciences,
University of California at Berkeley, in partial satisfaction of the requirements for the
degree of **Master of Science, Plan II.**

Approval for the Report and Comprehensive Examination:

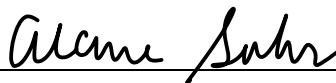
Committee:



Professor Dawn Song
Research Advisor

May 10, 2024

(Date)



Professor Alane Suhr
Second Reader

5/16/24

(Date)

Fleece QA: Self-Instruct Generation for Question Answering Dataset from Large Corpus

Yueheng Zhang

University of California, Berkeley
azicon@berkeley.edu

Abstract

This technical report presents a novel method for creating question-answering (QA) datasets from extensive unstructured textual corpora using large language models (LLMs). We produced over one million questions from the Wikitext dataset without requiring human labor, with the questions being more flexible and higher or similar quality to human annotators. A standout feature is our advanced evaluation method that uses a guided self-instruction method to assess question quality on a scale from 1 to 5 using LLM, offering a more cohesive and consistent evaluation than unguided self-instruct ratings, and matches that of human labeling. The system includes an intuitive user interface and visualization tools that facilitate easy dataset analysis and refinement. Importantly, this dataset is tailored for Retrieval-Augmented Generation (RAG) and long-context evaluation across large corpora, significantly enhancing the performance of QA systems in handling complex queries. This framework sets a new standard for future QA research, model building, and system evaluations.

1 Introduction

Question-answering (QA) systems have made significant strides in recent years, with the development of powerful models and the availability of large-scale datasets. However, the current landscape of QA datasets has been dominated by human-generated questions, often sourced through platforms like Amazon Mechanical Turk (MTurk) [8, 1]. While these datasets have proven effective in evaluating QA systems, they suffer from scalability issues and cannot be easily replicated for new corpora.

On the other hand, rule-based question-generation approaches have been proposed as an alternative to human-generated datasets [6]. These methods rely on specific data structures, such as knowledge graphs (e.g., Wikidata), to generate questions. However, this approach is limiting and may not be applicable in most scenarios where such structured data is not available.

Furthermore, the current evaluation of retrieval-augmented generation (RAG) systems, particularly those based on indexing, is inadequate due to the lack of corpus-specific, long-tailed questions that thoroughly test the capabilities of these systems [3, 7]. Existing datasets often focus on evaluating a system’s ability to search through a corpus rather than its proficiency in utilizing indexes effectively. Without suitable benchmarks, the true potential of index-based RAG systems remains untapped.

To address these challenges, we propose a novel QA dataset generation approach that leverages the power of large language models (LLMs) to create diverse, high-quality questions at an unprecedented scale. Our dataset aims to provide a comprehensive evaluation framework for indexing-based QA systems, focusing on their ability to navigate and synthesize information from indexes efficiently. By introducing new evaluation metrics and supporting self-training at scale, we endeavor to push the boundaries of QA research and enable the development of more robust and adaptable systems.

In this paper, we present our LLM-based question generation methodology, the resulting dataset, and a thorough analysis of its characteristics and performance on state-of-the-art QA models. We also discuss the implications of our work for the future of QA research and its potential impact on real-world applications.

2 Dataset Generation

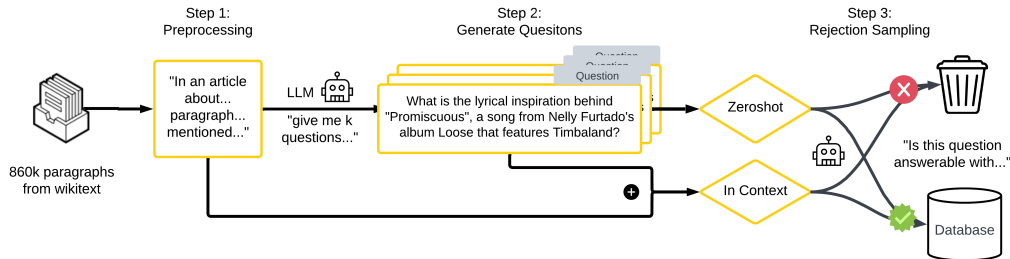


Figure 1: A high-level overview of the question generation workflow using large language models (LLMs). The process begins with 860k paragraphs from the Wikitext dataset, which are preprocessed and cleaned. The preprocessed text is then used by the LLM to generate questions. These questions undergo two types of evaluations: zero-shot and in-context. Based on these evaluations, questions are either accepted into the database if they are deemed answerable or rejected and deleted if they are not. The question shown in the figure is generated by Meta-Llama-3-70B-Instruct

The dataset generation procedure is framed within a replicable test designing paradigm where both the test designer and evaluator are human-aligned large language models (LLMs). This approach ensures that the process is systematic and scalable, incorporating human-in-the-loop (HITL) methodologies for alignment and evaluation, as detailed in the next section³.

In our case, we cleaned out 859,954 paragraphs from the wikitext [5] dataset from the [4] paper, generating over 1 million questions. We tested the question generation on two open-source LLMs: Mixtral-8x7B-v0.1 by MistralAI and Meta-Llama-3-70B-Instruct¹ by Meta; and one proprietary LLM, GPT-4-Turbo from OpenAI.

2.1 Dataset Selection

The first stage in our dataset generation involves the selection and preparation of a suitable test corpus. For our benchmark, we utilize the Wikitext dataset, available through Huggingface:

1. We selected the open-source Wikitext dataset on Huggingface as our starting point. This dataset provides a rich source of varied textual data suitable for natural language processing tasks.
2. The dataset was processed using standard data-cleaning practices. It was structured by splitting the text into hierarchical sections: article, paragraph, subparagraph, subsubparagraph, and subsubsubparagraph.
3. Each paragraph was prefixed with an optional context locator to make the text a self-contained unit of knowledge. This modification facilitates independent retrieval and comprehension, and the non-tagged version of the dataset will be reserved for testing relational retrievals later on.

2.2 Question Generation and Validation

In the second stage, a 'test designer' LLM was employed to generate and validate question-answer (QA) pairs from the cleaned dataset:

¹The bulk of the questions are generated by this model

1. For each paragraph within the articles, the LLM generated four potential questions, aiming to cover various aspects and details contained within the text.
2. Each generated question underwent a two-step validation process using a different LLM setup:
 - (a) The question was first posed in a zero-shot context to verify its formulation and relevance without accompanying text.
 - (b) It was then paired with its originating paragraph to check if the question could be accurately answered in context.
3. Questions that failed to meet the criteria in either of the validation steps were filtered out, ensuring only high-quality, answerable questions were retained.

2.3 Testing and Evaluation

The final stage involves testing and evaluating the generated QA pairs:

1. A 'test taker' LLM, Retrieval-Augmented Generation (RAG) system, or a human participant was tasked with answering the questions using the entire corpus as a reference.
2. A 'test grader' LLM then evaluated the answers by rating their correctness on a scale of 1 to 5, based on the degree of accuracy and completeness relative to the information in the corresponding paragraph.
3. Optionally, we can filter the dataset to retain only those questions exhibiting the largest discrepancy between in-context and zero-shot answers. This subset of 'hard' questions is particularly tailored to challenge the test taker and encourage learning and adaptation based on the corpus.

3 HITL and User Interface

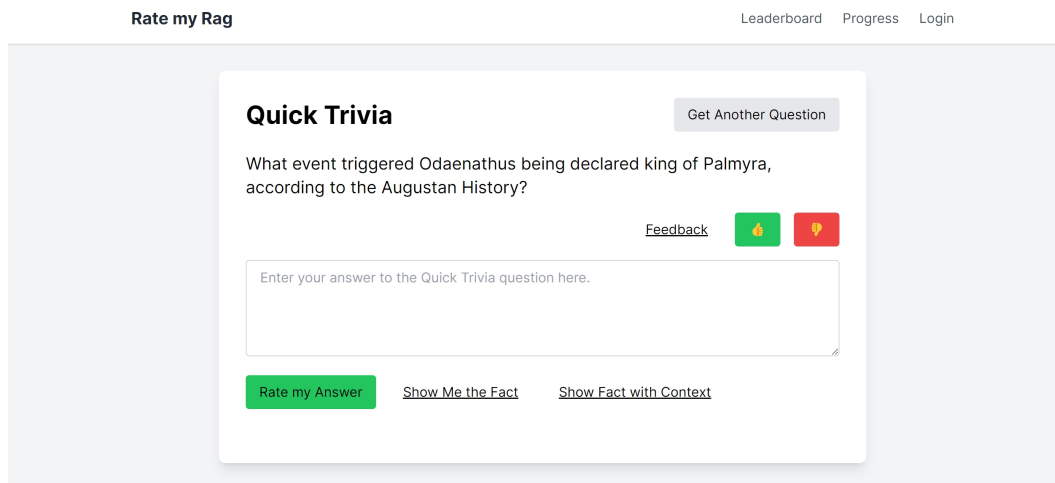


Figure 2: The Fleece QA Trivia Interface. The question shown in the image is generated by Meta-Llama-3-70B-Instruct

3.1 Interactive Trivia Game

Validating the well-formedness and effectiveness of the questions generated by our LLM can be a labor-intensive process that requires significant human effort. To address this challenge while engaging participants more effectively, we developed an interactive trivia game. This platform not only serves to validate the questions but also enhances user engagement by reframing the task into a game rather than a straightforward evaluation, which could be perceived as mundane.

The trivia game allows users to answer questions that are randomly presented. Each question is accompanied by a set of documents retrieved by a simple elastic search, which provides context to help answer the questions. Participants are encouraged to use external search tools, such as Google Search, placing human solvers on equal footing with Retrieval-Augmented Generation (RAG) systems.

Participants have the opportunity to flag questions through two reporting mechanisms:

1. A simple thumbs-up/thumbs-down button for immediate feedback.
2. A more detailed textual report submission option for specific feedback.

Figure 2 illustrates the trivia game interface, where users can interact with the question and reporting tools.

3.2 Competition Screen - Chatbot Arena Style Challenges

Building on the trivia game, we are developing a competitive platform akin to Chatbot Arena [2]. This platform will host weekly competitions where different test designers (models) are tasked with creating tests based on various unstructured text corpora. Participants will have access to a training set and can submit their solutions via a set of APIs, which is a feature planned for future implementation.

3.2.1 Proposed Features

- Allow users to download the training set and upload their solutions for testing.
- Optionally host existing datasets to broaden the scope of competitions and challenges.
- Feature a unified leaderboard that ranks both human users and RAG systems, fostering a competitive environment where human intelligence and artificial intelligence can be directly compared.

This competitive approach not only motivates participants but also provides a rich dataset of user interactions and solution effectiveness, which can be used to further refine the LLMs and RAG systems.

4 Experiments

4.1 Question Ratings

To gauge the quality of the questions, we randomly picked 10k questions from our dataset and ran them against our rating scheme. We employ a two-fold evaluation method where questions are first rated in a zero-shot setting to test generality and well-formedness, followed by an in-context setting to assess relevance. Ratings are based on correctness and completeness, with scores from 1 to 5, where 5 indicates a perfect match with the expected answer.

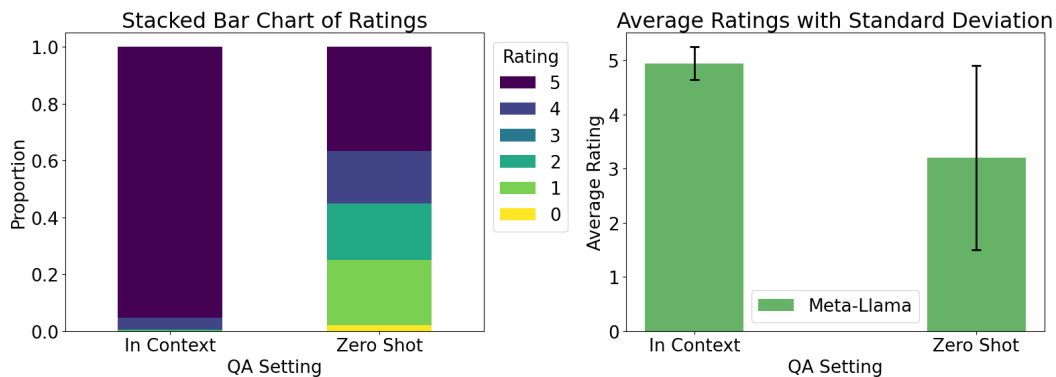


Figure 3: Questions are consistently answerable in context while being challenging in zero-shot

As seen in figure 3, the questions generated by Meta-Llama-3-70B-Instruct receive a consistent high score (4-5) with no instance of ratings below 4 for in-context answering of the questions generated. This may show that the questions well-formed with respect to the facts, meaning that the questions are answerable with the fact provided. We also see that the zero-shot answering is highly varied, as seen in both the stacked chart and the mean-variance bar graph, demonstrating our methods is able to generate questions with varying levels of difficulty. This shows that the questions generated with our methods can be suitable for RAG-based LLM systems. We can also see a number of scores 0's in our sample, proving that the questions captured and preserved the long-tailed facts and are challenging for even the state-of-the-art models.

4.2 Different Models

To see if this pattern persists across different models, especially vetting out LLM’s bias towards its own answers, we randomly sampled 50 of the questions generated and conducted analyses on two different models in a comparison study, one proprietary and one open source: GPT-4-Turbo and Meta-Llama-3-70B-Instruct. Both answers are rated by Latter, giving us a relatively just playing field.

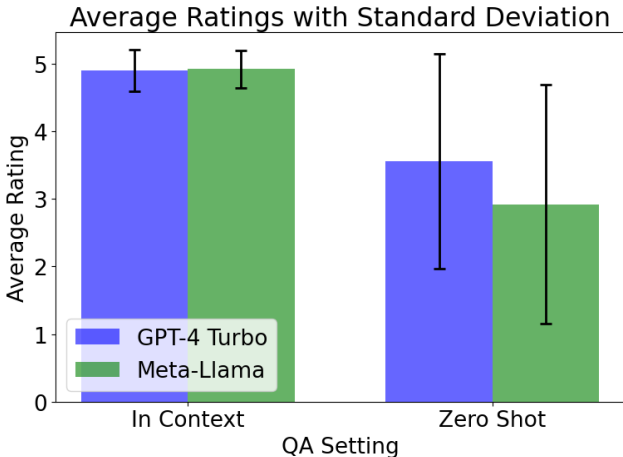


Figure 4: Average ratings with variance as error bars for GPT-4-Turbo and Meta-Llama-3-70B-Instruct.

In figure 4, we see that the average ratings are close to 5 for both models in the in-context setting, with Meta-Llama-3-70B-Instruct receiving a slight edge possibly due to its bias when rating itself. In the zero-shot setting, we see that both models has a significantly lower average rating as well as a larger variance. We also note that the zero-shot performance of GPT-4-Turbo is higher than Meta-Llama-3-70B-Instruct, possibly due to the former being a stronger model and able to remember more detailed information.

In the stacked graph 5, we can more clearly see the difference in the quality distribution of the generated questions. We observe that like the larger scale experiment done in the previous section 4, both models exhibit a clear stratified question quality in the zero-shot setting while maintaining a consistent high score for in-context answering. Notably, we see Meta-Llama-3-70B-Instruct having 0's in its evaluation while GPT-4-Turbo does not in the randomly drawn sample, potentially also pointing to stronger model performance.

Coefficient	Value	95% CI
Constant	3.56	[3.24, 3.88]
Setting Index (IC vs. ZS)	1.34	[0.89, 1.80]

Table 1: Regression results for GPT-4 Turbo with $R^2 = 0.258$ and adjusted $R^2 = 0.251$.

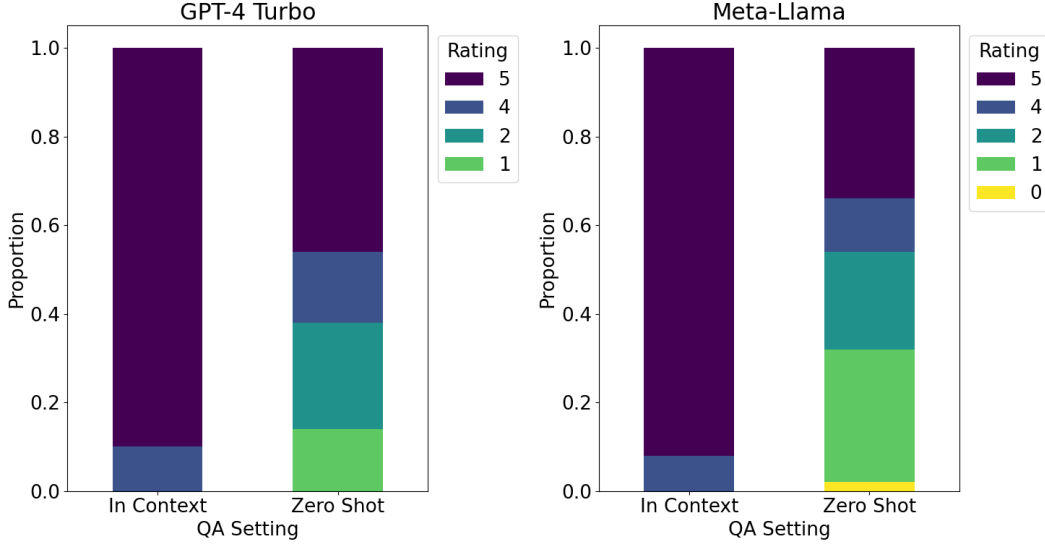


Figure 5: Questions are consistently answerable in context while being challenging in zero-shot

Coefficient	Value	95% CI
Constant	2.92	[2.56, 3.28]
Setting Index (IC vs. ZS)	2.00	[1.50, 2.50]

Table 2: Regression results for Meta-Llama-3-70B-Instruct with $R^2 = 0.389$ and adjusted $R^2 = 0.382$.

The regression analysis (as seen in table 1 and table 2) suggests that the questions we generated are both well-formed (as suggested by the consistently high in-context ratings), and sufficiently challenging (as shown by the high variance in the zero-shot setting with only 50 samples).

As a bonus, we also discovered that Meta-Llama-3-70B-Instruct generally provides higher and more consistent answer ratings compared to GPT-4 Turbo across both IC and ZS settings, as indicated by the higher R^2 value and overall rating levels. This suggests either a better generalization ability in varying interaction contexts for the Meta-Llama model, or a tendency to highly rate the model’s own generations.

5 Conclusion

This study has presented a comprehensive framework for generating, validating, and enhancing a dataset of questions designed to benchmark language models and human question-answering capabilities. By integrating human-aligned LLMs in both the test design and evaluation phases, we have established a replicable and scalable methodology that leverages advanced LLM capabilities. The inclusion of an interactive trivia game and a competitive platform akin to Chatbot Arena enriches user engagement and allows for the practical application of these models in a controlled, yet dynamic environment.

Our approach not only facilitates the effective validation of question well-formedness and relevancy but also enhances user interaction through an easy-to-use user interface, which inherently improves data collection quality and model feedback loops. The dual reporting system for question validation—allowing users to flag questions as either valid or invalid, and to provide detailed feedback—ensures that only high-quality data is used in further training and evaluation cycles.

5.1 Future Work

Looking ahead, several enhancements and expansions are planned to build on the foundation laid by this initial framework:

1. **Adversarial Question Generation:** Implementing techniques for generating questions that specifically target the weaknesses of current models. This would not only test the models' robustness but also help in identifying blind spots and areas requiring further improvement.
2. **Question Difficulty Rating:** Developing a system to automatically rate the difficulty of questions based on their complexity and the models' performance. This would allow for more nuanced training and benchmarking, where questions can be matched to model capabilities.
3. **Consistent Metrics for Self-Ask, Answer, and Grade:** Establishing a set of consistent metrics to evaluate the quality of questions and answers. This iterative process would enable continual refinement of questions, filtering out those that are irrelevant or poorly formed.
4. **Diversity in Question Quality:** Conducting extensive tests on question quality by training classifiers to recognize different types of questions. This would help in understanding question diversity and refining the question generation algorithms to produce a balanced and comprehensive set of questions.
5. **Fine-Tuning Corpus-Based Models and RAG Systems:** Utilizing the refined questions to fine-tune corpus-based models and RAG systems. This application promises to enhance the accuracy and reliability of these systems, ensuring they are well-calibrated to handle a variety of real-world informational needs.

These future directions underscore our commitment to advancing the field of NLP by developing tools and methodologies that not only challenge but also advance the capabilities of both human and machine learning systems in understanding and processing natural language.

Acknowledgments and Disclosure of Funding

I would like to extend my deepest gratitude to Professor Dawn Song for this great opportunity and guidance. Furthermore, I am grateful to Xiaoyuan Liu for his continuous support and strategic guidance throughout the project. Special thanks to Tianneng Shi for his invaluable assistance in debugging critical parts of our code, which was essential for the progression of our experiments.

I would like to acknowledge Berkeley RDI for providing access to the necessary GPU compute resources, without which this research would not have been possible. Additionally, we appreciate the general support and helpful discussions provided by Yu Gai, Siyuan Zhuang, Zhe Ye, and Zhun Wang, whose contributions have enriched our work.

A Appendix

A.1 Trivia Interface

The Trivia user interface is now publicly available under the following URL:

<http://gpublaze.ist.berkeley.edu:3000>

A.2 Code

The dataset creation process and backend are programmed under a single API. Code is made public at the following GitHub repository:

<https://github.com/CoLearn-Dev/FleeceKMBackend>

References

- [1] Shulin Cao, Jiaxin Shi, Liangming Pan, Lunyiu Nie, Yutong Xiang, Lei Hou, Juanzi Li, Bin He, and Hanwang Zhang. Kqa pro: A dataset with explicit compositional programs for complex question answering over knowledge base, 2022.
- [2] Wei-Lin Chiang, Lianmin Zheng, Ying Sheng, Anastasios Nikolas Angelopoulos, Tianle Li, Dacheng Li, Hao Zhang, Banghua Zhu, Michael Jordan, Joseph E. Gonzalez, and Ion Stoica. Chatbot arena: An open platform for evaluating llms by human preference, 2024.

- [3] Patrick 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. Retrieval-augmented generation for knowledge-intensive nlp tasks, 2021.
- [4] Stephen Merity, Caiming Xiong, James Bradbury, and Richard Socher. Pointer sentinel mixture models, 2016.
- [5] Stephen Merity, Caiming Xiong, James Bradbury, and Richard Socher. Wikitext. Hugging Face Dataset, 2016.
- [6] Iulian Vlad Serban, Alberto García-Durán, Caglar Gulcehre, Sungjin Ahn, Sarath Chandar, Aaron Courville, and Yoshua Bengio. Generating factoid questions with recurrent neural networks: The 30m factoid question-answer corpus, 2016.
- [7] Kurt Shuster, Mojtaba Komeili, Leonard Adolphs, Stephen Roller, Arthur Szlam, and Jason Weston. Language models that seek for knowledge: Modular search generation for dialogue and prompt completion, 2022.
- [8] Zhilin Yang, Peng Qi, Saizheng Zhang, Yoshua Bengio, William W. Cohen, Ruslan Salakhutdinov, and Christopher D. Manning. Hotpotqa: A dataset for diverse, explainable multi-hop question answering, 2018.