AmazonQA: A Review-Based Question Answering Task

Mansi Gupta[*] , **Nitish Kulkarni**[*] , **Raghuveer Chanda**[*] ,
Anirudha Rayasam and **Zachary C Lipton**

Carnegie Mellon University

{mansig1, nitishkk, rchanda, arayasam, zlipton}@cs.cmu.edu

Abstract

Every day, thousands of customers post questions on Amazon product pages. After some time, if they are fortunate, a knowledgeable customer might answer their question. Observing that many questions can be answered based upon the available product reviews, we propose the task of review-based QA. Given a corpus of reviews and a question, the QA system synthesizes an answer. To this end, we introduce a new dataset and propose a method that combines information retrieval techniques for selecting relevant reviews (given a question) and "reading comprehension" models for synthesizing an answer (given a question and review). Our dataset consists of 923k questions, 3.6M answers and 14M reviews across 156k products. Building on the well-known Amazon dataset, we collect additional annotations, marking each question as either answerable or unanswerable based on the available reviews. A deployed system could first classify a question as answerable and then attempt to generate an answer. Notably, unlike many popular QA datasets, here the questions, passages, and answers are all extracted from real human interactions. We evaluate numerous models for answer generation and propose strong baselines, demonstrating the challenging nature of this new task.

1 Introduction

E-commerce customers at websites like Amazon post thousands of product-specific questions per day. From our analysis of a large-scale dataset crawled from Amazon's website [McAuley and Yang, 2016], comprising question-answering data and product reviews, we observe that: (i) most questions have a response time of several days, with an average of around 2 days per question (excluding those questions which remain unanswered indefinitely); (ii) the product reviews are comparatively elaborate and informative as judged against the answers posted to specific questions; and (iii) following human-in-the-loop experiments, we discovered that more
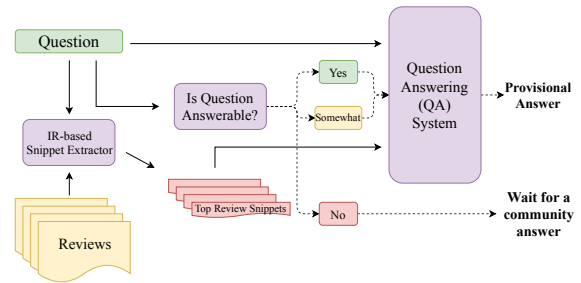
---

[*] contributed equally to this work

Figure 1: A community question-answering platform that provides provisional answers to questions using reviews.

than half of the questions can (at least partially) be answered using existing reviews (§4).

Figure 1 depicts our conception of a system that can provide on-demand, immediate answers to the user questions in a community question answering platform, by leveraging existing user reviews. Motivated by the possibility of designing such systems, we introduce the *review-based community question answering* task:

> Given a set of product reviews and a question concerning a specific product, generate an informative natural language answer.

We build upon the *question-answering* (QA) and product review dataset due to [McAuley and Yang, 2016], incorporating additional curation and annotations to create a new resource for automatic community question answering. Our resource, denoted *AmazonQA*, offers the following distinctive qualities: (i) it is extracted entirely from existing, real-world data; and (ii) it may be the largest public QA dataset with descriptive answers. To facilitate the training of complex ML-based QA models on the dataset, we provide rich pre-processing, extracting top review snippets for each question based on information retrieval (IR) techniques, filtering outliers and building an *answerability* classifier to allow for training of QA models on only the answerable questions. Further, we implement a number of heuristic-based and neural QA models for benchmarking the performance of some of the best-performing existing QA models on this dataset. We also provide human evaluation by experts as well as the users from the community question answering platform.

2 Related Work

Open-World and Closed-World Question Answering An open-world question-answering dataset constitutes a set of question-answer pairs accompanied by a knowledge database, with no explicit link between the question-answer pairs and knowledge database entries. SimpleQA [Bordes *et al.*, 2015] is a representative example of such a dataset. It requires simple reasoning over the Freebase knowledge database to answer questions. In closed-world question answering, the associated snippets are sufficient to answer all corresponding questions. Despite using open-world snippets in AmazonQA, we pose the final question-answering in a closed-world setting. We ensure that for any answerable question, the associated snippets contain all the required supporting information. Below, we highlight key distinguishing features of the recent popular closed-world QA datasets, comparing and contrasting them with AmazonQA. Basic statistics for related dataset are shown in Table 1.

Span-based Answers SQuAD [Rajpurkar *et al.*, 2016; Rajpurkar *et al.*, 2018] is a single-document dataset. The answers are multi-word spans from the context. To address the challenge of developing QA systems that can handle longer contexts, SearchQA [Dunn *et al.*, 2017] presents contexts consisting of more than one document. Here, the questions are not guaranteed to require reasoning across multiple documents as the supporting documents are collected through information retrieval after the (question, answer) pairs are determined. We follow a similar information retrieval scheme to curate AmazonQA, but unlike [Dunn *et al.*, 2017] whose answers are spans in the passage, ours are free-form.

Free-form Answers Some recent datasets, including [Nguyen *et al.*, 2016; He *et al.*, 2018] have free-form answer generation. MS MARCO [Nguyen *et al.*, 2016] contains user queries from Bing Search with human generated answers. Systems generate free-form answers and are evaluated by automatic metrics such as ROUGE-L and BLEU-1. Another variant with human generated answers is DuReader [He *et al.*, 2018] for which the questions and documents are based on user queries from Baidu Search and Baidu Zhidao.

Community/Opinion Question Answering The idea of question-answering using reviews and product information has been previously explored. [McAuley and Yang, 2016] address subjective queries using the relevance of reviews. [Wan and McAuley, 2016] extend this work by incorporating aspects of personalization and ambiguity. [Yu *et al.*, 2012] employ SVM classifiers for identifying the question aspects, question types, and classifying responses as opinions or not, optimizing salience, coherence and diversity to generate an answer. However, to the best of our knowledge, no prior work answers user queries from the corresponding reviews. Our baseline models are inspired by machine comprehension models like Bi-Directional Attention Flow (BiDAF) which predict the start and end positions of spans in the context.

3 Dataset

We build upon the dataset of [McAuley and Yang, 2016], who collected reviews, questions, and answers by scraping the product pages of Amazon.com, for the period of May 1996 to July 2014, spanning 17 categories of products including *Electronics*, *Video Games*, *Home and Kitchen*, etc. First, we preprocess and expand (via new annotations) this raw dataset to suit the QA task, detailing each step, and characterizing dataset statistics in the following sections.

3.1 Data Processing

As an artifact of web-crawling, many questions and reviews contain chunks of duplicate text that we identify and remove. We find that a few of the reviews, questions, and answers, are significantly longer as compared to their median lengths. To mitigate this, we remove the outliers from the dataset. The distributions of questions, answers and reviews on the basis of length are shown in Figure 2.

Along with the raw product reviews, we also provide query-relevant review-snippets for each question. We extract the snippets by first tokenizing the reviews, chunking the review text into snippets and ranking the snippets based on the TF-IDF metric. For tokenization, we remove all the capitalization except words that are fully capitalized as they might indicate abbreviations like 'IBM'. We consider the punctuation marks as individual tokens (the only exception is apostrophe (') which is commonly used in words such as don't, I'll etc., that should not be separated). We then chunk the review text into snippets of length 100, or to the end of a sentence boundary, whichever is greater. These candidate snippets are then ranked on the basis of relevance between question and the review-snippet using BM25 score [Robertson and Zaragoza, 2009]. The set of 10 most relevant snippets for each question are provided in the dataset.

3.2 Data Statistics

We obtain roughly 923k questions with 3.6M answers on 156k Amazon products having 14M unique reviews. The average length of questions, answers and reviews is 14.8, 31.2 and 72.0, respectively.

Answerability Annotation We classify each question-context pair as *answerable* or *non-answerable* based on whether the answer to the question is at least partially contained in reviews. To train this *Answerability classifier*, we obtain the training data through crowdsourcing using Amazon Mechanical Turk. The details of the Answerability classifier and the MTurk experiments are provided in §4. Our classifier marks roughly $570K$ pairs as *answerable* out of total of $923K$ question with 72% precision. Note that the Fig 2 shows similar shape of length distributions for both *All* and *Answerable* pairs, indicating that the annotations cannot simply be predicted by the question/reviews lengths. Further, there's no visible relation between the annotations and the answer lengths.

Question Type Annotation [McAuley and Yang, 2016] classify the questions as *descriptive* (open-ended) or *yes/no* (binary). They use an regular expression based approach proposed by [He and Dai, 2011] Category-wise statistics of each class are shown in Figure 5. We also extract first three keywords of each question and provide an analysis of 50 most frequent such 3-grams in Figure 4. About 30% of the questions are covered by these 3-grams.

| Dataset	Question Source	Document Source	Answer Type	# Qs	# Documents
NewsQA [Trischler *et al.*, 2017]	Crowd-Sourced	CNN	Span of words	100K	10K
SearchQA [Dunn *et al.*, 2017]	Generated	WebDoc	Span of words	140K	6.9M passages
SQuAD [Rajpurkar *et al.*, 2016]	Crowd-Sourced	Wiki	Span of words	100K	536
RACE [Lai *et al.*, 2017]	Crowd-Sourced	English Exams	Multiple choice	97l	28K
ARC [Clark *et al.*, 2018]	Generated	WebDoc	Multiple choice	7787	14M sentences,
DuReader [He *et al.*, 2018]	Crowd-Sourced	WebDoc/CQA	Manual summary	200K	1M
NarrativeQA [Kočiský *et al.*, 2018]	Crowd-Sourced	Books & Movies	Manual summary	46.7K	1,572 stories
MS MARCO [Nguyen *et al.*, 2016]	User Logs	WebDoc	Manual summary	1M	8.8M passages, 3.2M docs

Table 1: Existing QA Datasets
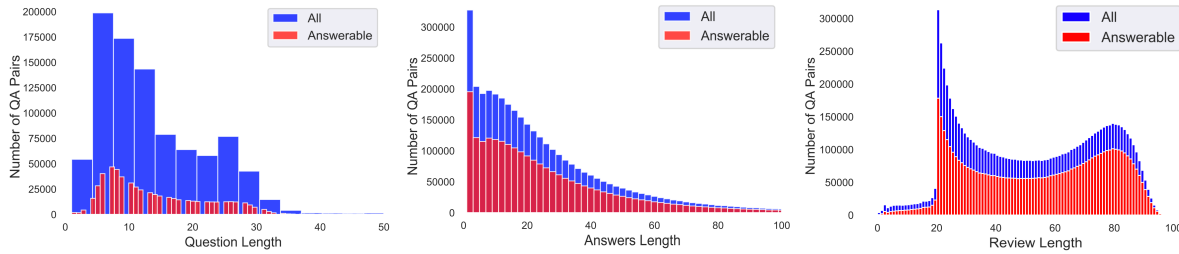
Figure 2: Left to Right: Question length distribution, Answer length distribution and Review length distribution for Answerable Questions and All (Answerable + Non-Answerable) Questions. Note that the shape of distributions of *All* questions and *Answerable* questions is similar

```
"answers": [
    {
        "helpful": [2, 2],
        "answerType": "NA",
        "answerText": "hmmm...I imagine so, but I used mine on s
    },
    {
        "helpful": [1, 1],
        "answerType": "NA",
        "answerText": "I have not used it on tall boots that are
    }
],
"questionText": "Does it work for tall boots that are too tight
"questionType": "descriptive",
"category": "Clothing_Shoes_and_Jewelry",
"asin": "B0010TN0UW",
"review_snippets": [
    "I ordered a pricey pair of fitted ...",
    "For some odd reason my left boot seems tighter then my righ
    "I took off the boots, put on some socks, and lo and behold,
    "..my darn big calves made the last two inches of zipping th
    "..they were very tight across the instep. I thought I ..",
    "I could not even put my feet inside my JS shoes .."
]
```

Figure 3: A sample instance from the *AmazonQA* dataset

3.3 Training, Development and Test Sets

We split the data into train, development and test sets on the basis of products rather than on the basis of questions or answers. It means that all the QA pairs for a particular product would be present in exactly one of the three sets. This is to ensure that the model learns to formulate answers from just the provided reviews and not from other information about the product. After filtering the answerable questions, the dataset contains 570,132 question-answer-review instances. We make a random 80-10-10 (train-development-test) split. Each split uniformly consists of 85% of *descriptive* and 15% *yes/no* question types.

4 Answerability

Since the QA pairs have been obtained independently from the context (reviews), there is no natural relationship between them. In order to ascertain that some part of the context actually answers the question, our system would require a classifier to determine whether a question-context pair is answerable or not. To do this, we conducted a crowdsourcing study on the *Amazon Mechanical Turk* platform to obtain the labels on the question-context pairs which we use as the training data.

4.1 MTurk Experiments

We provide workers with questions and a list of corresponding review snippets, asking them to label whether the associated reviews are sufficient to answer each question. We experimented with multiple design variations on a sample set containing a couple hundred expert tagged examples before rolling out the full study and try to optimize for cost and accuracy. A single MTurk *hit* is a web page shown to the workers, consisting of $N = 5$ questions, out of which one of them is a *'decoy'* question. *Decoy* questions are a set of expert annotated, relatively easier samples which have 100% inter-annotator agreement. The responses to the decoy questions allow us to compute a lower bound of a worker's performance. With this guiding metric, we detail our most efficient design template below. An example question with corresponding evidence in review text is provided in Table 2.

- In addition to the *answerable* (Y) and *not-answerable* (N) labels, we also provided a *somewhat* (S) label to the workers. *Somewhat* indicates that the reviews contain

Figure 4: Distribution of most frequent first 3 words in the questions of *AmazonQA* dataset, with examples

Figure 5: Question Type Distribution

information that partially answers the question. This is helpful to provide a provisional answer with limited insight, rather than a complete answer.

- Workers are required to mark snippets that (partially/completely) answer the question. Although we do not at present leverage this information, we observe that this inclusion leads workers to peruse the reviews, consequently minimizing mislabeling.

- Each hit has a total of 5 questions, including a decoy question. To discard careless workers, we estimate *Worker Error* as follows $\sum abs(worker\_label - true\_label)/2.0$. The label values are mapped as $\{Y : 1, S : 0, N : -1\}$

Examples of labels with the corresponding evidence are shown in Figure 6.

Q: What is the dimension of the product?	
Span	**Is Ans?**
"...the size of the bag is about 11in x 12in..."	Yes
"...this bag is big enough to fit a macbook..."	Somewhat
"...the size of the front pocket is very small..."	No

Table 2: An example question (Q) with evidence in review text (*Span*) and expected label (*Is Ans?*)

Finally, we rolled out the study on a total of 6000 (non-

Will this work on a Canon EOS 50D?

Mark all snippets that contain the answer (**either complete or partial**) to the above question

☐ The difference between this and an ordinary Canon lens will be much greater.As a side note - if you already own either or both the teleconverters (1.4x and 2x)......

☐ The next level is.... white Canon EF 200mm f/2L IS USM.F/2 and IS (Image Stabilization) versus f/2.8 and no image stabilization.Sharpness level is the same. Any.....

☐ It is at times frustrating if you can't work around. It is a bit of an issue, but I wouldn't consider it serious enough that you should avoid this lens for this.....

☐ If it doesn't suit you, don't buy it. While primes have much better IQ than zooms, I understand that is not the case when it comes to Canon L tele zooms such as......

☐ With so many choices in the Canon line up this lens gets overlooked quite a bit. The reason... It does not have image stabilization and lacks the versatility.......

Is the above question **answerable** using any of the snippets?

○ Yes
○ Somewhat
○ No

Figure 6: Example of an AMT page with a question, corresponding review snippets, and the label to be marked

decoy) questions, and eliminated the workers that performed poorly on the decoy questions. We retained 3,297 labeled questions after filtering. On an average it took ~ 8.6 minutes for workers to complete one hit.

4.2 Answerability Classifier

We process the data and extract query-relevant review-snippets as per §3.1. We use a sentence tokenizer on the top 5 review snippets to generate a list of sentences used as context to ascertain whether the question is answerable. To featurize the question-context pairs, we obtain two kinds of representations for the sentences, a corpus based tf-idf vectorizer and a tf-idf weighted Glove embeddings [Pennington *et al.*, 2014] to capture both statistical and semantic similarity between question and snippets.

As features, we use the number of question tokens, number of context tokens, absolute number and the fraction of question tokens that are also present in the context. We also use the cosine similarity between mean and max pooled representations of sentences with that of the question as features, thus

	Precision	Recall	F1-Score
Expert	0.92	0.81	0.86
Worker	0.73	0.73	0.73
Classifier	0.67	0.83	0.74

Table 3: Answerability precision, recall and F1-score by the expert, worker and the Logistic Regression classifier (C=1, threshold=0.6)

getting a total of 8 features for each item.

We split the dataset into train, validation and test sets with 2967, 330 and 137 question-context instances, respectively. We train a binary classifier on the data collected from MTurk, where the instances with labels as *Yes* and *Somewhat* are labeled as positive instances, and *No* are labeled as negative instances. The test set is annotated by 4 experts and we use it to measure the performance of MTurk workers, the binary classifier as well as the inter-expert agreement. The results are shown in the Table 3. We use this model to classify answerability for the whole dataset.

5 Baseline Models

To benchmark the generation of natural language answers, we implement three language models. We also implement reading-comprehension (RC) models that perform well on exiting span-based QA datasets to assess their performance on this task.

5.1 Language Models

To evaluate our ability to generate answers given reviews, we train a set of models for both answer generation (language modeling) and conditional language modeling (sequence to sequence transduction). If a is an answer, q is the corresponding question, and R is a set of reviews for the product, we train models to approximate the conditional distributions: $P(a)$, $P(a \mid q)$ and $P(a \mid q, R)$.

We train a simple language model that estimates the probability of an answer, $P(a)$, via the chain rule. The other two models estimate the probability of an answer conditioned on (i) just the question and (ii) both the question and the reviews. By evaluating models for all three tasks, we can ensure that the models are truly making use of each piece of additional information. Such ablation tests are necessary, especially in light of recent studies showing several NLP tasks to be easier than advertised owing to some components being unnecessary [Kaushik and Lipton, 2018; Gururangan *et al.*, 2018].

The three language models not only provide us an insight into the difficulty of predicting an answer using the question and reviews but also act as natural baselines for generative models trained on this dataset. To implement these language models, we use a generalized encoder-decoder based sequence-to-sequence architecture (Figure 7). The reviews are encoded using an LSTM-based encoder, and the encoded representations are averaged to form an aggregate review representation. The question representation (also by an LSTM-based encoder) and the aggregated review representation are concatenated and used to initialize an LSTM-decoder that generates the tokens of the answer at each step.
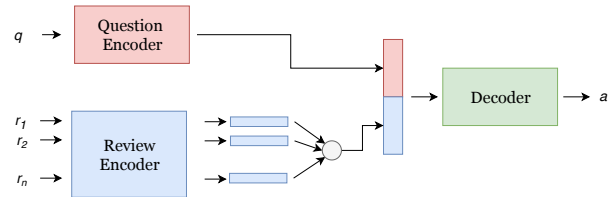

Figure 7: A schematic overview of the model $P(a \mid q, R)$. The $P(a \mid q)$ and $P(a)$ models are special cases where the review representation or both review and question are absent. All encoders and decoders are LSTM-based.

5.2 Span-based QA Model

To assess the performance of span-based RC models on our dataset, we convert our dataset into a span-based format. To do that, we use the descriptive answers from users to heuristically create a span (sequences of words) from the reviews that best answer the question. We then train a span based model, R-Net [Group, 2017], that uses a gated self-attention mechanism and pointer networks to find the location of the answers in the reviews.

Span Heuristics To create spans for supervision, we first create a set of candidate spans by either considering (i) all n-grams, where $n \in \{10, 20\}$, or (ii) all sentences in the reviews. We then rank the spans based on heuristics such as (i) BLEU-2 [Papineni *et al.*, 2002] or ROUGE [Lin, 2004] with the actual answers, and (ii) BM25 based IR score match with the question. To evaluate span quality, we annotate a test sample of 100 questions and corresponding spans from each heuristic as *answerable* and *not-answerable* using the spans. From this evaluation (Table 4), we identify IR- and BLEU-4-based sentences as the most effective span heuristics. Even though the fraction of questions *answerable* using the spans is relatively small, the spans serve as noisy supervision for the models owing to the large size of the training data. In table 5, we show performance of R-Net when trained on spans generated by two of these span heuristics.

Span Heuristic	fraction of *answerable* questions	
	sentences	n-grams
IR	0.58	0.24
BLEU-2	0.44	0.21
BLEU-4	0.45	0.26

Table 4: Expert evaluation of span heuristics on test set.

6 Results and Evaluation

Language Models For the language models, we compare the validation perplexities for all the three models (§5) on the test dataset of *AmazonQA*. Since perplexity is the exponential of the negative log likelihood per each word in the corpus, a lower perplexity indicates that the model deems the answers as more likely. Naturally, we expect the perplexities of the model $P(a \mid q, R)$ to be lower than those of $P(a \mid q)$, which should, in turn, be lower than perplexities from $P(a)$. We empirically verify this on the test set of AmazonQA (table 6).

	Bleu-1	Bleu-2	Bleu-3	Bleu-4	Rouge	$P(a \mid q)$ / $P(a \mid q, R)$
Heuristic Baselines						
A sentence from reviews as answer						
Random sentence	78.56	63.95	44.37	29.87	49.12	97.27 / 88.51
Top-1 using IR	89.49	74.80	56.76	43.52	61.48	128.32 / 115.66
Top-1 Using BLEU	**92.74**	**78.43**	**60.91**	**48.08**	**62.68**	-
Entire review as answer						
Top-1 (helpfulness score)	20.66	19.78	16.39	12.54	37.01	146.00 / 267.86
Top-1 (Wilson score)	20.74	19.84	16.44	12.58	37.26	144.44 / 126.07
Neural Baseline						
R-Net						
BLEU Heuristic	47.04	40.32	31.48	23.92	40.22	-
Human Answers						
Amazon User Community	80.88	68.86	54.36	42.01	62.18	-
Expert (Spans)	68.33	57.79	44.61	34.43	51.09	-
Expert (Descriptive)	53.67	46.56	37.81	30.76	53.31	-

Table 7: The performances and *perplexities* of various methods on the *AmazonQA* test set. The neural baseline, R-Net, is trained using spans created from BLEU Heuristic as explained in 5.2

Model	Test Perplexity
$P(a)$	97.01
$P(a \mid q)$	70.13
$P(a \mid q, R)$	65.40

Table 6: Perplexities of the language models on the test set

Span Heuristic	Exact Match	F1
BLEU_2	5.71	33.97
ROUGE	5.94	30.35

Table 5: The test-set performance of R-Net using different span generating heuristics for supervision

Heuristic-based answers For comparison with our baseline models, we consider the following trivial heuristic-based experiments to predict the answer: (i) **A sentence from the reviews as answer**: We consider the top-ranked sentence based on IR and BLEU-2 scores as an answer to the question. We also experiment with using a random sentence as an answer; (ii) **An entire review as the answer**: We use the top review based on (i) *helpfulness* of the reviews, as indicated by Amazon users; and (ii) *Wilson Score*[1] [Agresti and Coull, 1998] derived from *helpfulness* and *unhelpfulness* of the reviews, as the answer.

While these heuristics are largely similar to the span heuristics described in 5.2, the difference is that here, each of these heuristics is treated as a "QA model" in isolation for comparison with more complex models while the span heuristics are used to generate noisy supervision for training a span-based neural model.

Metrics As the answers to both *yes/no* and *descriptive* questions are descriptive in nature, we use BLEU and ROUGE scores as the evaluation metrics. We use the answers provided by the Amazon users as the reference answer.

Human Evaluation We compare to human performance for both experts and Amazon users. For expert evaluation, we annotate a test set of 100 questions for both span-based and descriptive answers. To compute the performance of Amazon users, for each question, we evaluate an answer using the other answers to the same question as reference answers.

Table 7 shows the performances of the different baseline models on our dataset. We note that the scores of the answers from Amazon users as well as the sentence-based heuristic baselines are higher than the those of the span-based model (R-net). This indicates a large scope of improvement for the QA models on this dataset.

Note that a random sentence from the review performs nearly as well as answers provided by Amazon users. This might be due to the fact that the Amazon users can see the existing answers and would be more inclined to write a new answer to add additional information. This also partially explains the high scores for IR-based top-sentence heuristic. Since many of the questions are of the type *yes/no*, the conventional descriptive answer evaluation metrics such as BLEU and ROUGE, that are based on token similarity may not be the best metrics for evaluation. A better way to evaluate the system would be to have a combined set of metrics such as accuracy for *yes/no* questions, BLEU and ROUGE for descriptive and perplexities from the language models.

7 Conclusion

We present a large Question Answering dataset, that is interesting for the following reasons: i) the answers are provided by users in a real-world scenario; ii) the questions are frequently only partially answerable based on the reviews, leaving the challenge to provide the best answer under partial information; and iii) the dataset is large, comprising several categories and domains thus possibly useful for learning to answer out-of-domain questions. To promote research in this direction, we publicly release[2] the dataset and our implementations of all baselines.

---

[1]Lower bound of Wilson score 95% CI for a Bernoulli parameter

[2]https://github.com/amazonqa/amazonqa

References

[Agresti and Coull, 1998] Alan Agresti and Brent A Coull. Approximate is better than "exact" for interval estimation of binomial proportions. *The American Statistician*, 52(2):119–126, 1998.

[Bordes *et al.*, 2015] Antoine Bordes, Nicolas Usunier, Sumit Chopra, and Jason Weston. Large-scale simple question answering with memory networks. *CoRR*, abs/1506.02075, 2015.

[Clark *et al.*, 2018] Peter Clark, Isaac Cowhey, Oren Etzioni, Tushar Khot, Ashish Sabharwal, Carissa Schoenick, and Oyvind Tafjord. Think you have solved question answering? try arc, the ai2 reasoning challenge. *arXiv preprint arXiv:1803.05457*, 2018.

[Dunn *et al.*, 2017] Matthew Dunn, Levent Sagun, Mike Higgins, V Ugur Guney, Volkan Cirik, and Kyunghyun Cho. Searchqa: A new q&a dataset augmented with context from a search engine. *arXiv preprint arXiv:1704.05179*, 2017.

[Group, 2017] Natural Language Computing Group. R-net: Machine reading comprehension with self-matching networks. May 2017.

[Gururangan *et al.*, 2018] Suchin Gururangan, Swabha Swayamdipta, Omer Levy, Roy Schwartz, Samuel Bowman, and Noah A Smith. Annotation artifacts in natural language inference data. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 2 (Short Papers)*, volume 2, pages 107–112, 2018.

[He and Dai, 2011] Jing He and Decheng Dai. Summarization of yes/no questions using a feature function model. In *Asian Conference on Machine Learning*, pages 351–366, 2011.

[He *et al.*, 2018] Wei He, Kai Liu, Jing Liu, Yajuan Lyu, Shiqi Zhao, Xinyan Xiao, Yuan Liu, Yizhong Wang, Hua Wu, Qiaoqiao She, et al. Dureader: a chinese machine reading comprehension dataset from real-world applications. In *Proceedings of the Workshop on Machine Reading for Question Answering*, pages 37–46, 2018.

[Kaushik and Lipton, 2018] Divyansh Kaushik and Zachary C Lipton. How much reading does reading comprehension require? a critical investigation of popular benchmarks. In *Empirical Methods in Natural Language Processing (EMNLP)*, 2018.

[Kočiský *et al.*, 2018] Tomáš Kočiský, Jonathan Schwarz, Phil Blunsom, Chris Dyer, Karl Moritz Hermann, Gáabor Melis, and Edward Grefenstette. The narrativeqa reading comprehension challenge. *Transactions of the Association of Computational Linguistics*, 6:317–328, 2018.

[Lai *et al.*, 2017] Guokun Lai, Qizhe Xie, Hanxiao Liu, Yiming Yang, and Eduard Hovy. Race: Large-scale reading comprehension dataset from examinations. In *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing*, pages 785–794, 2017.

[Lin, 2004] Chin-Yew Lin. Rouge: A package for automatic evaluation of summaries. *Text Summarization Branches Out*, 2004.

[McAuley and Yang, 2016] Julian McAuley and Alex Yang. Addressing complex and subjective product-related queries with customer reviews. In *Proceedings of the 25th International Conference on World Wide Web*, pages 625–635. International World Wide Web Conferences Steering Committee, 2016.

[Nguyen *et al.*, 2016] Tri Nguyen, Mir Rosenberg, Xia Song, Jianfeng Gao, Saurabh Tiwary, Rangan Majumder, and Li Deng. Ms marco: A human generated machine reading comprehension dataset. *choice*, 2640:660, 2016.

[Papineni *et al.*, 2002] Kishore Papineni, Salim Roukos, Todd Ward, and Wei-Jing Zhu. Bleu: a method for automatic evaluation of machine translation. In *Proceedings of the 40th annual meeting on association for computational linguistics*, pages 311–318. Association for Computational Linguistics, 2002.

[Pennington *et al.*, 2014] Jeffrey Pennington, Richard Socher, and Christopher Manning. Glove: Global vectors for word representation. In *Proceedings of the 2014 conference on empirical methods in natural language processing (EMNLP)*, pages 1532–1543, 2014.

[Rajpurkar *et al.*, 2016] Pranav Rajpurkar, Jian Zhang, Konstantin Lopyrev, and Percy Liang. 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, 2016.

[Rajpurkar *et al.*, 2018] Pranav Rajpurkar, Robin Jia, and Percy Liang. Know what you don't know: Unanswerable questions for squad. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, volume 2, pages 784–789, 2018.

[Robertson and Zaragoza, 2009] Stephen Robertson and Hugo Zaragoza. The probabilistic relevance framework: Bm25 and beyond. *Found. Trends Inf. Retr.*, 3(4):333–389, April 2009.

[Trischler *et al.*, 2017] Adam Trischler, Tong Wang, Xingdi Yuan, Justin Harris, Alessandro Sordoni, Philip Bachman, and Kaheer Suleman. Newsqa: A machine comprehension dataset. *ACL 2017*, page 191, 2017.

[Wan and McAuley, 2016] Mengting Wan and Julian McAuley. Modeling ambiguity, subjectivity, and diverging viewpoints in opinion question answering systems. In *Data Mining (ICDM), 2016 IEEE 16th International Conference on*, pages 489–498. IEEE, 2016.

[Yu *et al.*, 2012] Jianxing Yu, Zheng-Jun Zha, and Tat-Seng Chua. Answering opinion questions on products by exploiting hierarchical organization of consumer reviews. In *Proceedings of the 2012 Joint Conference on Empirical Methods in Natural Language Processing and Computational Natural Language Learning*, pages 391–401. Association for Computational Linguistics, 2012.