CSE 576 Project Phase 1 - Dataset Creation
QA Needing Deeper Reasoning
Unanswerable Question Generation

Nickolas Dodd

September 24, 2020

Introduction

Over the last few years, there has been an explosion in the number of new datasets developed for Question Answering (QA) and Reading Comprehension (RC) systems. Most of these datasets are shallow in the sense that the majority of the questions can be answered without any real need for natural language understanding or deeper reasoning capabilities. Indeed, the majority of these datasets have focused on asking questions for which the answers were spans of text in the given context paragraph. Transformer based neural architectures such as BERT have achieved high levels of accuracy on these datasets, however recent research has shown that they are prone to overfitting to textual triggers and spurious correlations with predicted labels and perform poorly on adversarially selected inputs.

Thus, recently, several new datasets have been developed in order to introduce complexity along two main directions: diversifying the reasoning types needed to answer questions and explicitly differentiating between questions that have different levels of knowledge access. The main motivation for the first extension is a natural one: a truly intelligent agent should be capable of performing multiple types of reasoning seamlessly. However, historically, many QA datasets have been dominated by shallow questions, such as factoid-style questions, that can be solved easily by fine-tuning existing language models and don't require any notions of deeper reasoning capabilities. In addition, crowdsourced datasets were generally found to follow formulaic patterns and rely on simple heuristics such as negation or entity swapping that were easily caught by simple heuristic methods. Through diligent data collection and expertly designed crowd-sourcing tasks, several datasets now exist that contain adversarially crafted questions which pose serious challenges to the BERT-based systems.

The second extension distingushes between the level of knowledge that is needed to answer the questions, and at the broadest level can be partitioned

into the following three categories: questions answerable directly from the given textual context, questions that require external world knowledge (that isn't directly present from the context), and unanswerable questions. Not surprisingly given the theme discussed so far, most datasets focus exclusively on questions that can be answered directly from the context, in which the tasks devolve into selecting the correct span from the context paragraph. QA systems that attempt to address the second level, questions requiring some external knowledge, can take many forms; from encoding the knowledge explicitly in rule based systems, to implicit knowledge encapsulated in large neural networks such as BERT. The third level, unanswerable questions, was popularized in the release of the SQUAD 2.0 dataset, in which 50,000 adversarially written questions were added to the SQUAD 1.1 dataset. The introduction of unanswerable questions severely degraded performance of the state of the art systems trained on the SQUAD dataset at the time, however recently several systems have made significant progress and have surpassed human level performance on this dataset. As such, recent datasets such as QUAIL seek to identify and address the deficiencies in SQUAD 2.0 by introducing questions that require deeper reasoning capabilities, and define unanswerable questions in a way that incorporates uncertainty in both contextual inputs and world knowledge. The majority of datasets that include unanswerable questions, such as SQUAD 2.0 and QUAIL have been human generated by crowdsourced workers, and the literature on unanswerable question generation is relatively sparse. This report will discuss some ideas for the synthetic data generation of unanswerable questions.

1 Existing Datasets

As mentioned in the previous section, there has been a recent explosion in the number of QA datasets published in recent years. Most of these datasets end up getting solved extremely quickly, which has raised concerns about bias and data artifacts introduced during the question generation process from crowd sourced workers, rule based approaches, and methods that utilize question templates. The result is models that are incapable of deeper reasoning, but achieve high performance on these datasets by learning spurious correlations and patterns in the dataset. In this section we discuss two datasets which have been designed to address some of these issues: ARC and QUAIL.

1.1 Think you have Solved Question Answering? Try ARC, the AI2 Reasoning Challenge

The ARC question set contains grade-school science questions and is partitioned into Easy (5197 questions) and Challenge (2590 questions) sets, where the latter contains only questions that were answered incorrectly by both a retrieval-based algorithm and a word co-occurence algorithm. All of the questions are multiple choice, and most contain excactly four choices. The target student grade level ranges from 3rd to 9th grade. Predecessors of ARC include

both the SQuAD and SNLI datasets. Most of the current state of the art models are incapable of performing reasoning and only encode surface-level knowledge from the corpus, making them perform poorly on this challenge. For example, GPT-3 achieved accuracy results that were more than 25 percentage points lower than the state-of-the-art model for ARC, averaging an accuracy score in the low 50's on the ARC challenge set. In fact, the authors showed in their experiments that the leading neural models were unable to beat a randomized baseline approach on ARC at the time of release. In addition to the question set, the ARC challenge includes a corpus that is composed of 14 million science sentences. A sampled analysis suggests the corpus mentions knowledge relevant to 95% of the Challenge questions, and the authors have made the use of the corpus for the ARC challenge optional.

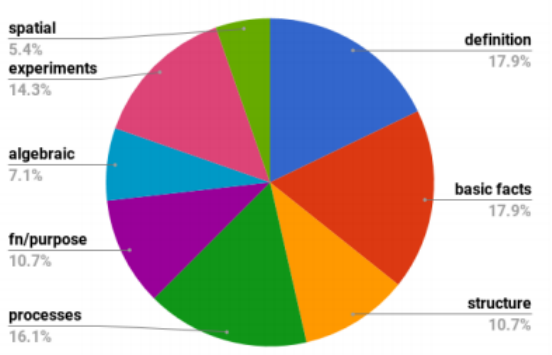


Figure 1: Relative sizes of different knowledge types suggested by the ARC Challenge Set.

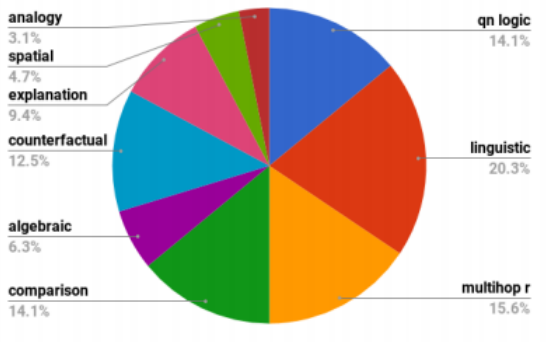


Figure 2: Relative sizes of different reasoning types suggested by the ARC Challenge Set.

Solver	Test Scores Challenge Set	Test Scores Easy Set
IR (dataset defn)	(1.02)†	(74.48)†
PMI (dataset defn)	(2.03)†	(77.82)†
IR (using ARC Corpus)	20.26	62.55
TupleInference	23.83	60.81
DecompAttn‡	24.34	58.27
Guess-all ("random")	25.02	25.02
DGEM-OpenIE‡	26.41	57.45
BiDAF‡	26.54	50.11
TableILP	26.97	36.15
DGEM	27.11	58.97

†These solvers were used to define the dataset, affecting scores.
‡Code available at https://github.com/allenai/arc-solvers

Setting	PIQA	ARC (Easy)	ARC (Challenge)	OpenBookQA
Fine-tuned SOTA	79.4	**92.0**[KKS+20]	**78.5**[KKS+20]	**87.2**[KKS+20]
GPT-3 Zero-Shot	**80.5***	68.8	51.4	57.6
GPT-3 One-Shot	**80.5***	71.2	53.2	58.8
GPT-3 Few-Shot	**82.8***	70.1	51.5	65.4

Table 3.6: GPT-3 results on three commonsense reasoning tasks, PIQA, ARC, and OpenBookQA. GPT-3 Few-Shot PIQA result is evaluated on the test server. See Section 4 for details on potential contamination issues on the PIQA test set.

1.2 QUAIL

QUAIL is a relatively new reading comprehension and question answering dataset composed of challenging multiple choice questions that span nine different reasoning tasks and four domains. The dataset is balanced, with approximately the same number of questions coming from each domain and reasoning type. The design methodology was developed in such a way as to attempt to mitigate two common pitfalls with existing QA datasets; the fact that not all texts are suitable for every question type, and the reality that crowd workers often do not write diverse questions. Major goals for the dataset were to reduce bias and data artifacts present in the majority of the existing QA datasets and to introduce diagnostic abilities by annotating question types so that suspiciously easy portions of the dataset could be identified and remedied. The questions are human generated and were produced by crowdsourced workers, prompted by relatively lengthy context paragraphs ranging bewteen 300-350 words each. These context paragraphs were drawn from a corpora consisting of 200 hand-selected texts for each domain, where each text in the domain was chosen so that it would make sense to human readers without any larger context.

Each question in QUAIL has four answer options, three answer options related to the context paragraph and a fourth "not enough information" option used to specify whether the question is unanswerable. The definition of an unanswerable questions in QUAIL is a question which cannot be answered with the given context paragraph, and for which external (world) knowledge doesn't

increase the likelihood of any of the answer options. In contrast, answerable world knowledge questions are ones in which the system needs to combine facts related to the specific given context paragraph with external knowledge, which makes one of the proposed answer options more likely than the others. The encapsulation of external knowledge in unanswerable question design is, on one hand, a natural and important requirement for enabling deeper reasoning, and on the other hand, an extremely challenging requirement to satisfy from a synthetic data generation perspective. In fact, in other existing QA datasets which contain unanswerable questions, most definitions solely require that the answer not be given in the context paragraph. This lends itself to extractive QA systems which do not explicitly incorporate external knowledge in their models. But in spite of this difficulty, unanswerable questions have been identified as one of the leading attempts to force more complex reasoning in QA systems, alongside reasoning over long texts and multiple documents. QUAIL is in fact the first QA and RC benchmark combining text-based questions, world knowledge questions, and unanswerable questions all together in the same dataset.

2 Task

The goal of our group project is to design, develop, and implement a QA system that possesses deeper reasoning capabilities. We will begin by synthetically creating a QA dataset, and then use this dataset to pretrain a model, which we will then test on QUAIL. The motivation behind the need for deeper reasoning in QA systems has been elaborated upon in the preceding sections. In QUAIL, questions are broadly categorized into three groups: text-based questions, world-knowledge questions, and unanswerable questions. It should be noted that some factoid style world-knowledge type questions may not be indicative of deeper reasoning capabilities. For instance, a majority of existing QA datasets focus on common knowledge, which is likely to have been present in the pretraining data of larger models such as BERT. In addition, unanswerable questions which are too dissimilar to their corresponding context paragraphs or questions, or for which the provided answer type doesn't match the type proposed by the new question, are easily detectable by simple heuristic based methods. Therefore, the most popular datasets to date that contain unanswerable questions have been human generated according to a pre-specified set of validation criteria. In the remainder of this report we first outline the task for our group assignment, list some examples of unanswerable questions for the QA dataset, and then discuss some potential ideas related to synthetic unanswerable question generation building off of recent work in this area.

2.1 Group Project Task Description

- **Input to System**: The inputs to our final trained model will be the question, context paragraph, and corresponding answer options. If we wish to utilize the reasoning type annotations that we include in our dataset, then an upstream network will need to first predict the question type using these inputs.
- **Output**: The output of the model will be the predicted answer choice. Matching QUAIL's answer choice structure would require three answer options related to the question and context paragraph and a fourth "not enough information" option specifying that the question is unanswerable.
- **Annotations**: Question type annotations will be provided for each question indicating the type of reasoning task needed to answer the question.
- **Evaluation Metrics**: Our main evaluation metric will be accuracy scores on the QUAIL test set.

3 Examples

Manual examples for unanswerable questions are provided in this section to give an idea of what the form of the final dataset will look like. For some of the following examples, we simply require that the question cannot be directly answerable from the text. The context paragraphs for each example are taken from paragraphs of preceding sections of this report.

1. - **Context**: First paragraph of this report, starting with "Over the last few years, . . . "
   - **Question**: How many QA and RC datasets have been developed over the last few years?
   - **Answer Choices**:
     (a) 10
     (b) 231
     (c) 42
     (d) Not enough information.
   - **Answer**: (d) Not enough information
   - **Reasoning type**: Unanswerable Question
2. - **Context**: Second paragraph of this report, starting with "Thus, recently, . . . "
   - **Question**: Why do crowd-sourced datasets generally follow formulaic patterns and rely on simple heuristics such as negation or entity swapping?
   - **Answer Choices**:

(a) Due to the instructions provided to the crowd workers.
(b) The crowd workers don't get paid enough to put in more effort.
(c) Crowd workers are known to be sleep deprived in general, negatively affecting performance.
(d) Not enough information.

- **Answer**: (d) Not enough information
- **Reasoning type**: Unanswerable Question

3.
- **Context**: Third paragraph of this report, starting with "The second extension distinguishes . . . "
- **Question**: By what year will the majority of datasets that contain unanswerable questions no longer be human generated by crowdsourced workers?
- **Answer Choices**:
  (a) 2021
  (b) 2045
  (c) Stardate 2258.42
  (d) Not enough information.
- **Answer**: (d) Not enough information
- **Reasoning type**: Unanswerable Question

4.
- **Context**: Fourth paragraph of this report, starting with "As mentioned in the previous . . . "
- **Question**: Which dataset generation techniques are least likely to introduce bias and data artifacts?
- **Answer Choices**:
  (a) Crowd sourced workers
  (b) Rule-based question generation
  (c) Methods utilizing question templates
  (d) Not enough information.
- **Answer**: (d) Not enough information
- **Reasoning type**: Unanswerable Question

5.
- **Context**: Fifth paragraph of this report, starting with "The ARC question set contains . . . "
- **Question**: What would happen if we increased the target student grade level ranges from 3rd to 9th grade to 10th to 12th grade?
- **Answer Choices**:
  (a) The model variance would increase.
  (b) The model prediction uncertainty would increase.

(c) The number of training examples would decrease.

(d) Not enough information.

- **Answer**: (d) Not enough information
- **Reasoning type**: Unanswerable Question

4 Synthetic Dataset Generation

4.1 Learning to Ask Unanswerable Questions

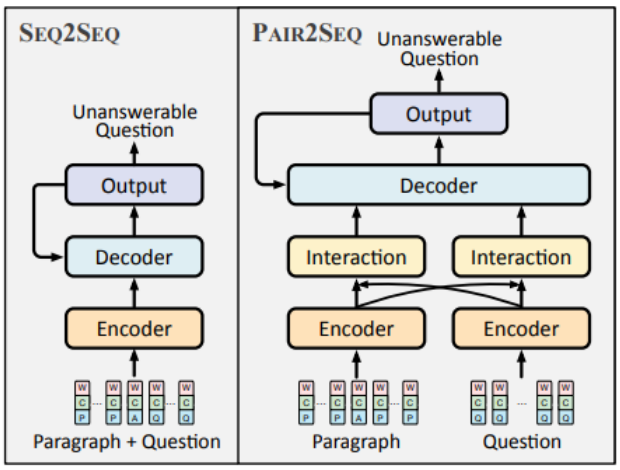
This proposed approach will extend work in the paper, Learning to Ask Unanswerable Questions for Machine Reading Comprehension (Zhu et. al, 2019) by attempting to incorporate external world knowledge into the unanswerable question generation process. In the original formulation in the paper, we are given both an answerable question q_a and context paragraph p that contains answer a, and our goal is to generate unanswerable question q_u. There are three criteria that q_u should satisfy:

1. q_u cannot be answered by p
2. q_u must be relevant to both p and q_a
3. The type of answer elicited by q_u must match a's type

The authors proposes the use of two different encoder-decoder style neural architectures and factorize the probability of generating unanswerable question q_u accordingly as

$$P(q_u \mid q_a, p, a) = \prod_{t=1}^{|q_u|} P(q_u^{(t)} \mid q_u^{(<t)}, q_a, p, a)$$

where $q_u^{(<t)} = q_u^{(1)} \dots q_u^{(t-1)}$. In both approaches there are three different types of embeddings used: word, character, and token type embeddings, where token type embeddings differentiate between question, answer, and paragraph type tokens. The first model uses LSTM based neural networks for both encoding and decoding, with attention mechanism applied on the encoding to form context vectors on each time step. The unanswerable question tokens are ultimately generated using a learnable convex combination of tokens taken via copy mechanism from the input layers, or sampled from a vocabulary distribution using the hidden decoder and context states. The second approach uses a pair-to-sequence model to better capture the interactions between input questions and context paragraphs, by encoding them separately, and then using attention-based matching to make them aware of each other.



As these approaches imply, data augmentation for synthetic generation of unanswerable questions relies on a repository of answerable questions. Thus, we propose to augment unanswerable questions on the set of answerable questions generated by other members of the team, in addition to answerable questions in related datasets such as SQUAD and ARC. For the proportion of this answerable base dataset which is composed of text-based questions, the method as described in the paper can be used as is to generate the augmented unanswerable questions. Synthetic generation of unanswerable questions for questions requiring external knowledge is unexplored in the literature. We propose extensions to the models in (Zhu et. al, 2019) as a first attempt towards this goal.

A survey of this type of QA reasoning task is given in Natural Language QA Approaches using Reasoning with External Knowledge (Baral et. al, 2020). In Section 3.1 of this survey, an overview of recent methods for extracting external knowledge is detailed. Extracting external knowledge based on the context paragraph and plausible answer will enable the generation of unanswerable questions that take related external knowledge into account that isn't present directly in the context paragraph. Since the unanswerable question generation procedure detailed above depends on a base of answerable questions to augment from, the exact external knowledge extraction method to be used should in principle depend on the reasoning type and source of knowledge used for each of the other question types generated by others on the team for this project. Since these question type annotations will be available for each question, ideally we could select specific tailored knowledge extraction methods based on the approach taken for each respective reasoning type based on the type of entities being reasoned over, and whether unstructured vs. structured knowledge is more appropriate. If this specialized approach turns out to be inpractical for any reason due to the constraints of the project, general approaches that extract structured knowledge from language models appear promising.

Let us refer to the knowledge extraction mechanism to be used generically as, $\mathcal{K}(p, q)$. We propose utilizing this external knowledge to generate unanswerable questions using three generic strategies of varying complexity:

1. Concatenate $\mathcal{K}(p, q)$ directly to the given context, and feed through the existing seq-to-seq or pair-to-seq models. Care must be taken in this approach to constrain $|q_u|$ generated to be around the same length as questions in the rest of the dataset.

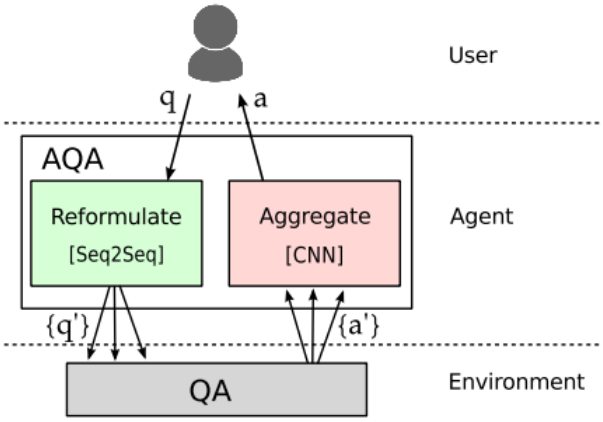
2. Augment the system with a fourth token type embedding, e, which denotes external knowledge tokens. Use the seq-to-seq model by packing the relevant retrieved external knowledge into the question+paragraph sequence using special separator tokens in between. Aside from making the necessary adjustments to account for the additional fourth token type, the rest of the encoder-decoder architecture is essentially then the same, and would be used as is from that point on.

3. The third approach would be to add the fourth token type as mentioned in approach two, as well as introducing a third encoder-interaction pair to the pair-to-seq model to directly model interactions between question, context, and external knowledge and use the attention mechanism described previously to make all three aware of each other.

The proposed extensions are listed in order of increasing complexity and the validation approach would be to run quick experiments for the first approach before moving on to the other approaches. Ablation studies on these approaches using different external knowledge mechanisms would be desirable by the end of project deadline if any of these techniques are used. In any case, the model without any of the proposed extensions can be used to generate unanswerable questions for questions that do not require external knowledge.

4.2 Active QA with Reinforcement Learning

In this second proposed approach, a modification of the ActiveQA model first described in Ask the Right Questions: Active Question Reformulation using Reinforcement Learning (Buck et. al, 2017) will be given that attempts to extend the architecture to generate unanswerable questions. In the original architecture, the authors envision an RL agent that sits between a user and the QA environment, which they model as a black-box, that learns how to reformulate questions submitted by the user in order to improve answers retrieved from the QA system. We propose to extend this architecture to generate unanswerable questions by reformulating the reward signal and introducing external knowledge bases in the black-box QA module.



For the reformulation model, the policy is implemented by a sequence-to-sequence model that assigns probability

$$\pi_\theta(q \mid q_0) = \prod_{t=1}^{T} p(w_t \mid w_1, \ldots, w_{t-1}, q_0)$$

for generated question $q = w_1, \ldots, w_T$ where T is the lgneth of q with tokens $w_t \in V$ for vocabulary V, and q_0 is the original given question. Skipping details given in the paper, policy gradient methods are used to formulate a final objective of

$$\mathbb{E}_{q \sim \pi_\theta(\cdot \mid q_0)}[R(f(q)) - B(q_0)] + \lambda H[\pi(q \mid q_0)]$$

where

- $H[\pi_\theta(q \mid q_0)]$ is an entropy regularization term to avoid collapse onto sub-optimal deterministic policies
- $B(q_0) = \mathbb{E}_{q \sim \pi_\theta(\cdot \mid q_0)}[R(f(q))]$ is the baseline reward subtracted off to reduce variance of the estimator for the gradients
- R is the reward function, which we use to select the best possible answer $a^* = \arg\max_a R(a \mid q_0)$
- $a = f(q)$ represents the unknown function of a question q computed by the environment

There are two main aspects of this system which make it promising towards the given goal of generating synthetic unanswerable questions. First, the QA environment is assumed to be a black-box to the rest of the environment, and the agent doesn't have access to its parameters, activations, or gradients, and, as mentioned by the authors in the paper, it allows for the possibility for interaction to be incorporated with other sources of information such as structured data from knowledge bases. In the paper, the authors use a BiDAF model

for their QA environment. Second, the structure of the objective is written in such a way that the function encapsulating the reward signal is modular. In the original experiments carried out by the authors, token-level F1 score is used for the reward.

To enable unanswerable question generation using this framework, we first propose augmenting the QA environment with external knowledge sources and extraction techniques as outlined in the (Baral et. al, 2020). Second, we recast the reward signal such that

1. Each $q \in \{q'\}$ is relevant to both the original question q_0 and the context paragraph p, where $\{q'\}$ is the set of reformulated questions generated by the model.
2. For each $a \in \{a'\}$, the type for a matches the answer type implied by q_0.
3. Subject to these two constraints, we seek to maximize the entropy/uncertainty of the generated answers.

4.3 Deep Paraphrasing

The third augmentation approach is to use the method and model detailed in Paraphrase Generation with Deep Reinforcement Learning (Li et. al, 2018) for question, context, and answer choice paraphrase generation. This approach differs from the previous two in that it is a general data augmentation method for question generation and is not specialized towards unanswerable questions specifically. The motivation for using this technique is taken from the key takeaways in QUAIL, in which they found that adversarial paraphrasing is an effective strategy for generating challenging questions that degrade performance even for leading models such as BERT. For more information on paraphrasing approach taken in QUAIL, please refer to https://github.com/text-machine-lab/quail/blob/master/quail_challenge_set/Paraphrasing.md.

The model in (Li et. al, 2018) uses a generator-evaluator approach to paraphrase generic sentences, where the generator uses a seq-to-seq model with attention and copy mechanisms and the evaluator utilizes a decomposable attention-based deep matching model. Since we propose no extensions or changes to this model, the reader is referred to the reference paper for details. At a high level, given an input sentence $X = [x_1, \ldots, x_S]$ a paraphrased sentence $Y = [y_1, \ldots, y_T]$ is generated, and the evaluation network then ensures that Y is indeed a paraphrasing of X. This technique could be used not only to generate new questions based on a given answerable or unanswerable question, but also to paraphrase context paragraphs (or portions thereof) and answer choices. The generic applicability of this approach makes it suitable as a post-process data augmentation technique for each set of questions generated by the members of our group, but extensions that include this method in the respective

question generation models for each reasoning type should be reasonably easy to adapt as well. For instance, we can imagine using paraphrasing to augment answerable question/answer pairs before training the extensions to the models of (Zhu et. al, 2019), or in conjunction with the active question reformulation method proposed in the previous section. Using an automated paraphrasing approach would allow us to go easily go beyond the 2/1 paraphrasing strategy specified in QUAIL's paraphrasing experiments to create a powerful adversarially augmented dataset.