Crowdsourcing Multiple Choice Science Questions

Johannes Welbl[*]
Computer Science Department
University College London
j.welbl@cs.ucl.ac.uk

Nelson F. Liu[*]
Paul G. Allen School of
Computer Science & Engineering
University of Washington
nfliu@cs.washington.edu

Matt Gardner
Allen Institute for Artificial Intelligence
mattg@allenai.org

Abstract

We present a novel method for obtaining high-quality, domain-targeted multiple choice questions from crowd workers. Generating these questions can be difficult without trading away originality, relevance or diversity in the answer options. Our method addresses these problems by leveraging a large corpus of domain-specific text and a small set of existing questions. It produces model suggestions for document selection and answer distractor choice which aid the human question generation process. With this method we have assembled *SciQ*, a dataset of 13.7K multiple choice science exam questions.[1] We demonstrate that the method produces in-domain questions by providing an analysis of this new dataset and by showing that humans cannot distinguish the crowdsourced questions from original questions. When using *SciQ* as additional training data to existing questions, we observe accuracy improvements on real science exams.

1 Introduction

The construction of large, high-quality datasets has been one of the main drivers of progress in NLP. The recent proliferation of datasets for textual entailment, reading comprehension and Question Answering (QA) (Bowman et al., 2015; Hermann et al., 2015; Rajpurkar et al., 2016; Hill et al., 2015; Hewlett et al., 2016; Nguyen et al., 2016) has allowed for advances on these tasks, particularly with neural models (Kadlec et al., 2016; Dhingra et al., 2016; Sordoni et al., 2016; Seo et al., 2016). These recent datasets cover broad and general domains, but progress on these datasets has not translated into similar improvements in more targeted domains, such as science exam QA.

Science exam QA is a high-level NLP task which requires the mastery and integration of information extraction, reading comprehension and common sense reasoning (Clark et al., 2013; Clark, 2015). Consider, for example, the question *"With which force does the moon affect tidal movements of the oceans?"*. To solve it, a model must possess an abstract understanding of natural phenomena and apply it to new questions. This transfer of general and domain-specific background knowledge into new scenarios poses a formidable challenge, one which modern statistical techniques currently struggle with. In a recent Kaggle competition addressing 8th grade science questions (Schoenick et al., 2016), the highest scoring systems achieved only 60% on a multiple choice test, with retrieval-based systems far outperforming neural systems.

A major bottleneck for applying sophisticated statistical techniques to science QA is the lack of large in-domain training sets. Creating a large, multiple choice science QA dataset is challenging, since crowd workers cannot be expected to have domain expertise, and questions can lack relevance and diversity in structure and content. Furthermore, poorly chosen answer distractors in a multiple choice setting can make questions almost trivial to solve.

The first contribution of this paper is a general method for mitigating the difficulties of crowdsourcing QA data, with a particular focus on multiple choice science questions. The method is broadly similar to other recent work (Rajpurkar et al., 2016), relying mainly on showing crowd

[*]Work done while at the Allen Institute for Artificial Intelligence.

[1]Dataset available at http://allenai.org/data.html

Proceedings of the 3rd Workshop on Noisy User-generated Text, pages 94–106
Copenhagen, Denmark, September 7, 2017.

Example 1	Example 2	Example 3	Example 4
Q: What type of organism is commonly used in preparation of foods such as cheese and yogurt?	**Q:** What phenomenon makes global winds blow northeast to southwest or the reverse in the northern hemisphere and northwest to southeast or the reverse in the southern hemisphere?	**Q:** Changes from a less-ordered state to a more-ordered state (such as a liquid to a solid) are always what?	**Q:** What is the least dangerous radioactive decay?
1) mesophilic organisms 2) protozoa 3) gymnosperms 4) viruses	1) coriolis effect 2) muon effect 3) centrifugal effect 4) tropical effect	1) exothermic 2) unbalanced 3) reactive 4) endothermic	1) alpha decay 2) beta decay 3) gamma decay 4) zeta decay
Mesophiles grow best in moderate temperature, typically between 25°C and 40°C (77°F and 104°F). Mesophiles are often found living in or on the bodies of humans or other animals. The optimal growth temperature of many pathogenic mesophiles is 37°C (98°F), the normal human body temperature. Mesophilic organisms have important uses in food preparation, including cheese, yogurt, beer and wine.	Without Coriolis Effect the global winds would blow north to south or south to north. But Coriolis makes them blow northeast to southwest or the reverse in the Northern Hemisphere. The winds blow northwest to southeast or the reverse in the southern hemisphere.	Summary Changes of state are examples of phase changes, or phase transitions. All phase changes are accompanied by changes in the energy of a system. Changes from a more-ordered state to a less-ordered state (such as a liquid to a gas) are endothermic. Changes from a less-ordered state to a more-ordered state (such as a liquid to a solid) are always exothermic. The conversion . . .	All radioactive decay is dangerous to living things, but alpha decay is the least dangerous.

Figure 1: The first four *SciQ* training set examples. An instance consists of a question and 4 answer options (the correct one in green). Most instances come with the document used to formulate the question.

workers a passage of text and having them ask a question about it. However, unlike previous dataset construction tasks, we (1) need domain-relevant passages and questions, and (2) seek to create multiple choice questions, not direct-answer questions.

We use a two-step process to solve these problems, first using a noisy classifier to find relevant passages and showing several options to workers to select from when generating a question. Second, we use a model trained on real science exam questions to predict good answer distractors given a question and a correct answer. We use these predictions to aid crowd workers in transforming the question produced from the first step into a multiple choice question. Thus, with our methodology we leverage existing study texts and science questions to obtain new, relevant questions and plausible answer distractors. Consequently, the human intelligence task is shifted away from a purely *generative* task (which is slow, difficult, expensive and can lack diversity in the outcomes when repeated) and reframed in terms of a *selection*, *modification* and *validation* task (being faster, easier, cheaper and with content variability induced by the suggestions provided).

The second contribution of this paper is a dataset constructed by following this methodology. With a total budget of $10,415, we collected 13,679 multiple choice science questions, which we call *SciQ*. Figure 1 shows the first four training examples in *SciQ*. This dataset has a multiple choice version, where the task is to select the correct answer using whatever background information a system can find given a question and several answer options, and a direct answer version, where given a passage and a question a system must predict the span within the passage that answers the question. With experiments using recent state-of-the-art reading comprehension methods, we show that this is a useful dataset for further research. Interestingly, neural models do not beat simple information retrieval baselines on the multiple choice version of this dataset, leaving room for research on applying neural models in settings where training examples number in the tens of thousands, instead of hundreds of thousands. We also show that using *SciQ* as an additional source of training data improves performance on real 4^{th} and 8^{th} grade exam questions, proving that our method successfully produces useful in-domain training data.

2 Related Work

Dataset Construction. A lot of recent work has focused on constructing large datasets suitable for training neural models. QA datasets have been assembled based on Freebase (Berant et al., 2013; Bordes et al., 2015), Wikipedia articles (Yang et al., 2015; Rajpurkar et al., 2016; Hewlett et al.,

2016) and web search user queries (Nguyen et al., 2016); for reading comprehension (RC) based on news (Hermann et al., 2015; Onishi et al., 2016), children books (Hill et al., 2015) and novels (Paperno et al., 2016), and for recognizing textual entailment based on image captions (Bowman et al., 2015). We continue this line of work and construct a dataset for science exam QA. Our dataset differs from some of the aforementioned datasets in that it consists of natural language questions produced by people, instead of cloze-style questions. It also differs from prior work in that we aim at the narrower domain of science exams and in that we produce multiple choice questions, which are more difficult to generate.

Science Exam Question Answering. Existing models for multiple-choice science exam QA vary in their reasoning framework and training methodology. A set of sub-problems and solution strategies are outlined in Clark et al. (2013). The method described by Li and Clark (2015) evaluates the coherence of a scene constructed from the question enriched with background KB information, while Sachan et al. (2016) train an entailment model that derives the correct answer from background knowledge aligned with a max-margin ranker. Probabilistic reasoning approaches include Markov logic networks (Khot et al., 2015) and an integer linear program-based model that assembles proof chains over structured knowledge (Khashabi et al., 2016). The Aristo ensemble (Clark et al., 2016) combines multiple reasoning strategies with shallow statistical methods based on lexical co-occurrence and IR, which by themselves provide surprisingly strong baselines. There has not been much work applying neural networks to this task, likely because of the paucity of training data; this paper is an attempt to address this issue by constructing a much larger dataset than was previously available, and we present results of experiments using state-of-the-art reading comprehension techniques on our datasets.

Automatic Question Generation. Transforming text into questions has been tackled before, mostly for didactic purposes. Some approaches rely on syntactic transformation templates (Mitkov and Ha, 2003; Heilman and Smith, 2010), while most others generate cloze-style questions. Our first attempts at constructing a science question dataset followed these techniques. We found the methods did not produce high-quality science questions, as there were problems with selecting relevant text, generating reasonable distractors, and formulating coherent questions.

Several similarity measures have been employed for selecting answer distractors (Mitkov et al., 2009), including measures derived from WordNet (Mitkov and Ha, 2003), thesauri (Sumita et al., 2005) and distributional context (Pino et al., 2008; Aldabe and Maritxalar, 2010). Domain-specific ontologies (Papasalouros et al., 2008), phonetic or morphological similarity (Pino and Esknazi, 2009; Correia et al., 2010), probability scores for the question context (Mostow and Jang, 2012) and context-sensitive lexical inference (Zesch and Melamud, 2014) have also been used. In contrast to the aforementioned similarity-based selection strategies, our method uses a feature-based ranker to learn plausible distractors from original questions. Several of the above heuristics are used as features in this ranking model. Feature-based distractor generation models (Sakaguchi et al., 2013) have been used in the past by Agarwal and Mannem (2011) for creating biology questions. Our model uses a random forest to rank candidates; it is agnostic towards taking cloze or humanly-generated questions, and it is learned specifically to generate distractors that resemble those in real science exam questions.

3 Creating a science exam QA dataset

In this section we present our method for crowdsourcing science exam questions. The method is a two-step process: first we present a set of candidate passages to a crowd worker, letting the worker choose one of the passages and ask a question about it. Second, another worker takes the question and answer generated in the first step and produces three distractors, aided by a model trained to predict good answer distractors. The end result is a multiple choice science question, consisting of a question q, a passage p, a correct answer a^*, and a set of distractors, or incorrect answer options, $\{a'\}$. Some example questions are shown in Figure 1. The remainder of this section elaborates on the two steps in our question generation process.

3.1 First task: producing in-domain questions

Conceiving an original question from scratch in a specialized domain is surprisingly difficult; performing the task repeatedly involves the danger of

falling into specific lexical and structural patterns. To enforce diversity in question content and lexical expression, and to inspire relevant in-domain questions, we rely on a corpus of in-domain text about which crowd workers ask questions. However, not all text in a large in-domain corpus, such as a textbook, is suitable for generating questions. We use a simple filter to narrow down the selection to paragraphs likely to produce reasonable questions.

Base Corpus. Choosing a relevant, in-domain base corpus to inspire the questions is of crucial importance for the overall characteristics of the dataset. For science questions, the corpus should consist of topics covered in school exams, but not be too linguistically complex, specific, or loaded with technical detail (e.g., scientific papers). We observed that articles retrieved from web searches for science exam keywords (e.g. "animal" and "food") yield a significant proportion of commercial or otherwise irrelevant documents and did not consider this further. Articles from science-related categories in Simple Wikipedia are more targeted and factual, but often state highly specific knowledge (e.g., "Hoatzin can reach 25 inches in length and 1.78 pounds of weight.").

We chose science study textbooks as our base corpus because they are directly relevant and linguistically tailored towards a student audience. They contain verbal descriptions of general natural principles instead of highly specific example features of particular species. While the number of resources is limited, we compiled a list of 28 books from various online learning resources, including CK-12[2] and OpenStax[3], who share this material under a Creative Commons License. The books are about biology, chemistry, earth science and physics and span elementary level to college introductory material. A full list of the books we used can be found in the appendix.

Document Filter. We designed a rule-based document filter model into which individual paragraphs of the base corpus are fed. The system classifies individual sentences and accepts a paragraph if a minimum number of sentences is accepted. With a small manually annotated dataset of sentences labelled as either relevant or irrelevant, the filter was designed iteratively by adding filter rules to first improve precision and then recall on a held-out validation set. The final filter included lexical, grammatical, pragmatical and complexity based rules. Specifically, sentences were filtered out if they *i)* were a question or exclamation *ii)* had no verb phrase *iii)* contained modal verbs *iv)* contained imperative phrases *v)* contained demonstrative pronouns *vi)* contained personal pronouns other than third-person *vii)* began with a pronoun *viii)* contained first names *ix)* had less than 6 or more than 18 tokens or more than 2 commas *x)* contained special characters other than punctuation *xi)* had more than three tokens beginning uppercase *xii)* mentioned a graph, table or web link *xiii)* began with a discourse marker (e.g. *'Nonetheless'*) *xiv)* contained absoulute wording (e.g. *'never', 'nothing', 'definitely'*) *xv)* contained instructional vocabulary (*'teacher'*, *'worksheet'*, . . .).

Besides the last, these rules are all generally applicable in other domains to identify simple declarative statements in a corpus.

Question Formulation Task. To actually generate in-domain QA pairs, we presented the filtered, in-domain text to crowd workers and had them ask a question that could be answered by the presented passage. Although most undesirable paragraphs had been filtered out beforehand, a non-negligible proportion of irrelevant documents remained. To circumvent this problem, we showed each worker *three* textbook paragraphs and gave them the freedom to choose one or to reject all of them if irrelevant. Once a paragraph had been chosen, it was not reused to formulate more questions about it. We further specified desirable characteristics of science exam questions: no *yes/no* questions, not requiring further context, querying general principles rather than highly specific facts, question length between 6-30 words, answer length up to 3 words (preferring shorter), no ambiguous questions, answers clear from paragraph chosen. Examples for both desirable and undesirable questions were given, with explanations for why they were good or bad examples. Furthermore we encouraged workers to give feedback, and a contact email was provided to address upcoming questions directly; multiple crowdworkers made use of this opportunity. The task was advertised on *Amazon Mechanical Turk*, requiring *Master's* status for the crowdworkers, and paying a compensation of 0.30$ per HIT. A total of 175 workers participated in the whole crowdsourcing

[2] www.ck12.org
[3] www.openstax.org

project.

In 12.1% of the cases all three documents were rejected, much fewer than if a single document had been presented (assuming the same proportion of relevant documents). Thus, besides being more economical, proposing several documents reduces the risk of generating irrelevant questions and in the best case helps match a crowdworker's individual preferences.

3.2 Second task: selecting distractors

Generating convincing answer distractors is of great importance, since bad distractors can make a question trivial to solve. When writing science questions ourselves, we found that finding reasonable distractors was the most time-consuming part overall. Thus, we support the process in our crowdsourcing task with model-generated answer distractor suggestions. This primed the workers with relevant examples, and we allowed them to use the suggested distractors directly if they were good enough. We next discuss characteristics of good answer distractors, propose and evaluate a model for suggesting such distractors, and describe the crowdsourcing task that uses them.

Distractor Characteristics. Multiple choice science questions with nonsensical incorrect answer options are not interesting as a task to study, nor are they useful for training a model to do well on real science exams, as the model would not need to do any kind of science reasoning to answer the training questions correctly. The difficulty in generating a good multiple choice question, then, lies not in identifying expressions which are false answers to q, but in generating expressions which are *plausible* false answers. Concretely, besides being false answers, good distractors should thus:

- be grammatically consistent: for the question "When animals use energy, what is always produced?" a noun phrase is expected.
- be consistent with respect to abstract properties: if the correct answer belongs to a certain category (e.g., chemical elements) good distractors likely should as well.
- be consistent with the semantic context of the question: a question about animals and energy should not have *newspaper* or *bingo* as distractors.

Distractor Model Overview. We now introduce a model which generates plausible answer distrators and takes into account the above criteria. On a basic level, it ranks candidates from a large collection C of possible distractors and selects the highest scoring items. Its ranking function

$$r : (q, a^*, a') \mapsto s_{a'} \in [0, 1] \quad (1)$$

produces a confidence score $s_{a'}$ for whether $a' \in C$ is a good distractor in the context of question q and correct answer a^*. For r we use the scoring function $s_{a'} = P(a' \text{ is good} \mid q, a^*)$ of a binary classifier which distinguishes plausible (good) distractors from random (bad) distractors based on features $\phi(q, a^*, a')$. For classification, we train r on actual in-domain questions with observed false answers as the plausible (good) distractors, and random expressions as negative examples, sampled in equal proportion from C. As classifier we chose a random forest (Breiman, 2001), because of its robust performance in small and mid-sized data settings and its power to incorporate nonlinear feature interactions, in contrast, e.g., to logistic regression.

Distractor Model Features. This section describes the features $\phi(q, a^*, a')$ used by the distractor ranking model. With these features, the distractor model can learn characteristics of real distractors from original questions and will suggest those distractors that it deems the most realistic for a question. The following features of question q, correct answer a^* and a tentative distractor expression a' were used:

- bags of $GloVe$ embeddings for q, a^* and a';
- an indicator for POS-tag consistency of a^* and a';
- singular/plural consistency of a^* and a';
- log. avg. word frequency in a^* and a';
- Levenshtein string edit distance between a^* and a';
- suffix consistency of a^* and a' (firing e.g. for (*regeneration, exhaustion*));
- token overlap indicators for q, a^* and a';
- token and character length for a^* and a' and similarity therein;
- indicators for numerical content in q, a^* and a' consistency therein;

- indicators for units of measure in q, a^* and a', and for co-occurrence of the same unit;
- WordNet-based hypernymy indicators between tokens in q, a^* and a', in both directions and potentially via two steps;
- indicators for 2-step connections between entities in a^* and a' via a KB based on OpenIE triples (Mausam et al., 2012) extracted from pages in Simple Wikipedia about anatomical structures;
- indicators for shared Wordnet-hyponymy of a^* and a' to one of the concepts most frequently generalising all three question distractors in the training set (e.g. *element*, *organ*, *organism*).

The intuition for the knowledge-base link and hypernymy indicator features is that they can reveal sibling structures of a^* and a' with respect to a shared property or hypernym. For example, if the correct answer a^* is *heart*, then a plausible distractor a' like *liver* would share with a^* the hyponymy relation to *organ* in WordNet.

Model Training. We first constructed a large candidate distractor set C whose items were to be ranked by the model. C contained 488,819 expressions, consisting of (1) the 400K items in the GloVe vocabulary (Pennington et al., 2014); (2) answer distractors observed in training questions; (3) a list of noun phrases from Simple Wikipedia articles about body parts; (4) a noun vocabulary of ∼6000 expressions extracted from primary school science texts. In examples where a^* consisted of multiple tokens, we added to C any expression that could be obtained by exchanging one unigram in a^* with another unigram from C.

The model was then trained on a set of 3705 science exam questions (4^{th} and 8^{th} grade) , separated into 80% training questions and 20% validation questions. Each question came with four answer options, providing three good distractor examples. We used `scikit-learn`'s implementation of random forests with default parameters. We used 500 trees and enforced at least 4 samples per tree leaf.

Distractor Model Evaluation. Our model achieved $99,4\%$ training and $94,2\%$ validation accuracy overall. Example predictions of the distractor model are shown in Table 1. Qualitatively, the predictions appear acceptable in most cases, though the quality is not high enough to use them directly without additional filtering by crowd workers. In many cases the distractor is semantically related, but does not have the correct type (e.g., in column 1, "nutrient" and "soil" are not elements). Some predictions are misaligned in their level of specificity (e.g. "frogs" in column 3), and multiword expressions were more likely to be unrelated or ungrammatical despite the inclusion of part of speech features. Even where the predicted distractors are not fully coherent, showing them to a crowd worker still has a positive priming effect, helping the worker generate good distractors either by providing nearly-good-enough candidates, or by forcing the worker to think why a suggestion is not a good distractor for the question.

Distractor Selection Task. To actually generate a multiple choice science question, we show the result of the first task, a (q, a^*) pair, to a crowd worker, along with the top six distractors suggested from the previously described model. The goal of this task is two-fold: (1) quality control (validating a previously generated (q, a^*) pair), and (2) validating the predicted distractors or writing new ones if necessary.

The first instruction was to judge whether the question could appear in a school science exam; questions could be marked as ungrammatical, having a false answer, being unrelated to science or requiring very specific background knowledge. The total proportion of questions passing was 92.8%.

The second instruction was to select up to two of the six suggested distractors, and to write at least one distractor by themselves such that there is a total of three. The requirement for the worker to generate one of their own distractors, instead of being allowed to select three predicted distractors, was added after an initial pilot of the task, as we found that it forced workers to engage more with the task and resulted in higher quality distractors.

We gave examples of desirable and undesirable distractors and the opportunity to provide feedback, as before. We advertised the task on *Amazon Mechanical Turk*, paying 0.2$ per HIT, again requiring AMT *Master's* status. On average, crowd workers found the predicted distractors good enough to include in the final question around half of the time, resulting in 36.1% of the distractors in the final dataset being generated by the model (because workers were only allowed to pick two predicted distractors, the theoretical maximum is 66%). Acceptance rates were higher in

Q: Compounds containing an atom of what element, bonded in a hydrocarbon framework, are classified as amines?	**Q:** Elements have orbitals that are filled with what?	**Q:** Many species use their body shape and coloration to avoid being detected by what?	**Q:** The small amount of energy input necessary for all chemical reactions to occur is called what?
A: nitrogen	**A:** electrons	**A:** predators	**A:** activation energy
oxygen (0.982)	**ions** (0.975)	**viruses** (0.912)	conversely energy (0.987)
hydrogen (0.962)	atoms (0.959)	ecosystems (0.896)	**decomposition energy** (0.984)
nutrient (0.942)	crystals (0.952)	frogs (0.896)	membrane energy (0.982)
calcium (0.938)	protons (0.951)	distances (0.8952)	motion energy (0.982)
silicon (0.938)	neutrons (0.946)	**males** (0.877)	context energy (0.981)
soil (0.9365)	**photons** (0.912)	crocodiles (0.869)	**distinct energy** (0.980)

Table 1: Selected distractor prediction model outputs. For each QA pair, the top six predictions are listed in row 3 (ranking score in parentheses). Boldfaced candidates were accepted by crowd workers.

the case of short answers, with almost none accepted for the few cases with very long answers.

The remainder of this paper will investigate properties of *SciQ*, the dataset we generated by following the methodology described in this section. We present system and human performance, and we show that *SciQ* can be used as additional training data to improve model performance on real science exams.

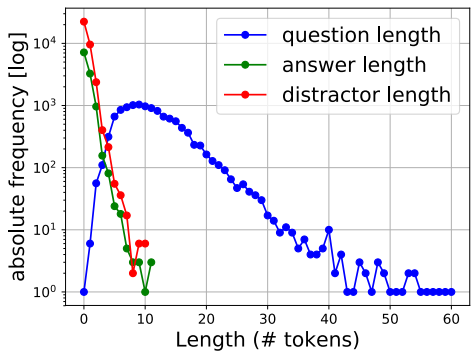

Figure 2: Total counts of question, answer and distractor length, measured in number of tokens, calculated across the training set.

Model	**Accuracy**
Aristo	77.4
Lucene	80.0
TableILP	31.8
AS Reader	74.1
GA Reader	73.8
Humans	87.8 ± 0.045

Table 2: Test set accuracy of existing models on the multiple choice version of *SciQ*.

3.3 Dataset properties

SciQ has a total of 13,679 multiple choice questions. We randomly shuffled this dataset and split it into training, validation and test portions, with 1000 questions in each of the validation and test portions, and the remainder in train. In Figure 2 we show the distribution of question and answer lengths in the data. For the most part, questions and answers in the dataset are relatively short, though there are some longer questions.

Each question also has an associated passage used when generating the question. Because the multiple choice question is trivial to answer when given the correct passage, the multiple choice version of *SciQ* does not include the passage; systems must retrieve their own background knowledge when answering the question. Because we have the associated passage, we additionally created a direct-answer version of *SciQ*, which has the passage and the question, but no answer options. A small percentage of the passages were obtained from unreleasable texts, so the direct answer version of *SciQ* is slightly smaller, with 10481 questions in train, 887 in dev, and 884 in test.

Qualitative Evaluation. We created a crowdsourcing task with the following setup: A person was presented with an original science exam question and a crowdsourced question. The instructions were to choose which of the two questions was more likely to be the real exam question. We randomly drew 100 original questions and 100 instances from the *SciQ* training set and presented the two options in random order. People identified the science exam question in 55% of the cases, which falls below the significance level of p=0.05 under a null hypothesis of a random guess[4].

[4] using normal approximation

4 SciQ Experiments

4.1 System performance

We evaluated several state-of-the-art science QA systems, reading comprehension models, and human performance on *SciQ*.

Multiple Choice Setting. We used the Aristo ensemble (Clark et al., 2016), and two of its individual components: a simple information retrieval baseline (Lucene), and a table-based integer linear programming model (TableILP), to evaluate *SciQ*. We also evaluate two competitive neural reading comprehension models: the Attention Sum Reader (AS Reader, a GRU with a pointer-attention mechanism; Kadlec et al. (2016)) and the Gated Attention Reader (GA Reader, an AS Reader with additional gated attention layers; Dhingra et al. (2016)). These reading comprehension methods require a supporting text passage to answer a question. We use the same corpus as Aristo's Lucene component to retrieve a text passage, by formulating five queries based on the question and answer[5] and then concatenating the top three results from each query into a passage. We train the reading comprehension models on the training set with hyperparameters recommended by prior work ((Onishi et al., 2016) for the AS Reader and (Dhingra et al., 2016) for the GA Reader), with early stopping on the validation data[6]. Human accuracy is estimated using a sampled subset of 650 questions, with 13 different people each answering 50 questions. When answering the questions, people were allowed to query the web, just as the systems were.

Table 2 shows the results of this evaluation. Aristo performance is slightly better on this set than on real science exams (where Aristo achieves 71.3% accuracy (Clark et al., 2016)).[7] Because TableILP uses a hand-collected set of background knowledge that does not cover the topics in *SciQ*, its performance is substantially worse here than on its original test set. Neural models perform reasonably well on this dataset, though, interestingly, they are not able to outperform a very simple information retrieval baseline, even when using exactly the same background information. This suggests that *SciQ* is a useful dataset for studying reading comprehension models in medium-data settings.

[5]The question text itself, plus each of the four answer options appended to the question text.

[6]For training and hyperparameter details, see Appendix

[7]We did not retrain the Aristo ensemble for *SciQ*; it might overly rely on TableILP, which does not perform well here.

Dataset	AS Reader	GA Reader
4^{th} grade	40.7%	37.6%
4^{th} grade + SciQ	45.0%	45.4%
Difference	+4.3%	+7.8%
8^{th} grade	41.2%	41.0%
8^{th} grade + SciQ	43.0%	44.3%
Difference	+1.8%	+3.3%

Table 3: Model accuracies on real science questions validation set when trained on 4^{th} / 8^{th} grade exam questions alone, and when adding *SciQ*.

Direct Answer Setting. We additionally present a baseline on the direct answer version of *SciQ*. We use the Bidirectional Attention Flow model (BiDAF; Seo et al. (2016)), which recently achieved state-of-the-art results on SQuAD (Rajpurkar et al., 2016). We trained BiDAF on the training portion of *SciQ* and evaluated on the test set. BiDAF achieves a 66.7% exact match and 75.7 F1 score, which is 1.3% and 1.6% below the model's performance on SQuAD.

4.2 Using *SciQ* to answer exam questions

Our last experiment with *SciQ* shows its usefulness as training data for models that answer real science questions. We collected a corpus of 4^{th} and 8^{th} grade science exam questions and used the AS Reader and GA Reader to answer these questions.[8] Table 3 shows model performances when only using real science questions as training data, and when augmenting the training data with *SciQ*. By adding *SciQ*, performance for both the AS Reader and the GA Reader improves on both grade levels, in a few cases substantially. This contrasts with our earlier attempts using purely synthetic data, where we saw models overfit the synthetic data and an overall performance decrease. Our successful transfer of information from *SciQ* to real science exam questions shows that the question distribution is similar to that of real science questions.

5 Conclusion

We have presented a method for crowdsourcing the creation of multiple choice QA data, with

[8]There are approx. 3200 8^{th} grade training questions and 1200 4^{th} grade training questions. Some of the questions come from `www.allenai.org/data`, some are proprietary.

a particular focus on science questions. Using this methodology, we have constructed a dataset of 13.7K science questions, called *SciQ*, which we release for future research. We have shown through baseline evaluations that this dataset is a useful research resource, both to investigate neural model performance in medium-sized data settings, and to augment training data for answering real science exam questions.

There are multiple strands for possible future work. One direction is a systematic exploration of multitask settings to best exploit this new dataset. Possible extensions for the direction of generating answer distractors could lie in the adaptation of this idea in negative sampling, e.g. in KB population. Another direction is to further bootstrap the data we obtained to improve automatic document selection, question generation and distractor prediction to generate questions fully automatically.

References

Manish Agarwal and Prashanth Mannem. 2011. Automatic gap-fill question generation from text books. In *Proceedings of the 6th Workshop on Innovative Use of NLP for Building Educational Applications*. Association for Computational Linguistics, Stroudsburg, PA, USA, IUNLPBEA '11, pages 56–64. http://dl.acm.org/citation.cfm?id=2043132.2043139.

Itziar Aldabe and Montse Maritxalar. 2010. *Automatic Distractor Generation for Domain Specific Texts*, Springer Berlin Heidelberg, Berlin, Heidelberg, pages 27–38.

Jonathan Berant, Andrew Chou, Roy Frostig, and Percy Liang. 2013. Semantic parsing on freebase from question-answer pairs. In *Proceedings of the 2013 Conference on Empirical Methods in Natural Language Processing, EMNLP 2013, 18-21 October 2013, Grand Hyatt Seattle, Seattle, Washington, USA, A meeting of SIGDAT, a Special Interest Group of the ACL*. pages 1533–1544. http://aclweb.org/anthology/D/D13/D13-1160.pdf.

Antoine Bordes, Nicolas Usunier, Sumit Chopra, and Jason Weston. 2015. Large-scale simple question answering with memory networks. *CoRR* abs/1506.02075. http://arxiv.org/abs/1506.02075.

Samuel R. Bowman, Gabor Angeli, Christopher Potts, and Christopher D. Manning. 2015. A large annotated corpus for learning natural language inference. In *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing (EMNLP)*. Association for Computational Linguistics.

Leo Breiman. 2001. Random forests. *Machine Learning* 45(1):5–32.

Peter Clark. 2015. Elementary school science and math tests as a driver for ai: Take the aristo challenge! In *Proceedings of the Twenty-Ninth AAAI Conference on Artificial Intelligence*. AAAI Press, AAAI'15, pages 4019–4021. http://dl.acm.org/citation.cfm?id=2888116.2888274.

Peter Clark, Oren Etzioni, Tushar Khot, Ashish Sabharwal, Oyvind Tafjord, Peter Turney, and Daniel Khashabi. 2016. Combining retrieval, statistics, and inference to answer elementary science questions. In *Proceedings of the Thirtieth AAAI Conference on Artificial Intelligence*. AAAI Press, AAAI'16, pages 2580–2586. http://dl.acm.org/citation.cfm?id=3016100.3016262.

Peter Clark, Philip Harrison, and Niranjan Balasubramanian. 2013. A study of the knowledge base requirements for passing an elementary science test. In *Proceedings of the 2013 Workshop on Automated Knowledge Base Construction*. ACM, New York, NY, USA, AKBC '13, pages 37–42. https://doi.org/10.1145/2509558.2509565.

Rui Correia, Jorge Baptista, Nuno Mamede, Isabel Trancoso, and Maxine Eskenazi. 2010. Automatic generation of cloze question distractors. In *Proceedings of the Interspeech 2010 Satellite Workshop on Second Language Studies: Acquisition, Learning, Education and Technology, Waseda University, Tokyo, Japan*.

Bhuwan Dhingra, Hanxiao Liu, William W. Cohen, and Ruslan Salakhutdinov. 2016. Gated-attention readers for text comprehension. *CoRR* abs/1606.01549. http://arxiv.org/abs/1606.01549.

Michael Heilman and Noah A. Smith. 2010. Good question! statistical ranking for question generation. In *Human Language Technologies: The 2010 Annual Conference of the North American Chapter of the Association for Computational Linguistics*. Association for Computational Linguistics, Stroudsburg, PA, USA, HLT '10, pages 609–617. http://dl.acm.org/citation.cfm?id=1857999.1858085.

Karl Moritz Hermann, Tomáš Kočiský, Edward Grefenstette, Lasse Espeholt, Will Kay, Mustafa Suleyman, and Phil Blunsom. 2015. Teaching machines to read and comprehend. In *Advances in Neural Information Processing Systems (NIPS)*. http://arxiv.org/abs/1506.03340.

Daniel Hewlett, Alexandre Lacoste, Llion Jones, Illia Polosukhin, Andrew Fandrianto, Jay Han, Matthew Kelcey, and David Berthelot. 2016. Wikireading: A novel large-scale language understanding task over wikipedia. *CoRR* abs/1608.03542. http://arxiv.org/abs/1608.03542.

Felix Hill, Antoine Bordes, Sumit Chopra, and Jason Weston. 2015. The goldilocks principle: Reading children's books with explicit memory representations. *CoRR* abs/1511.02301. http://arxiv.org/abs/1511.02301.

Rudolf Kadlec, Martin Schmid, Ondrej Bajgar, and Jan Kleindienst. 2016. Text understanding with the attention sum reader network. *CoRR* abs/1603.01547. http://arxiv.org/abs/1603.01547.

Daniel Khashabi, Tushar Khot, Ashish Sabharwal, Peter Clark, Oren Etzioni, and Dan Roth. 2016. Question answering via integer programming over semi-structured knowledge. In *Proceedings of the Twenty-Fifth International Joint Conference on Artificial Intelligence, IJCAI 2016, New York, NY, USA, 9-15 July 2016*. pages 1145–1152. http://www.ijcai.org/Abstract/16/166.

Tushar Khot, Niranjan Balasubramanian, Eric Gribkoff, Ashish Sabharwal, Peter Clark, and Oren Etzioni. 2015. Exploring markov logic networks for question answering. In *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing, EMNLP 2015, Lisbon, Portugal, September 17-21, 2015*. pages 685–694. http://aclweb.org/anthology/D/D15/D15-1080.pdf.

Yang Li and Peter Clark. 2015. Answering elementary science questions by constructing coherent scenes using background knowledge. In *EMNLP*. pages 2007–2012.

Mausam, Michael Schmitz, Robert Bart, Stephen Soderland, and Oren Etzioni. 2012. Open language learning for information extraction. In *Proceedings of the 2012 Joint Conference on Empirical Methods in Natural Language Processing and Computational Natural Language Learning*. Association for Computational Linguistics, Stroudsburg, PA, USA, EMNLP-CoNLL '12, pages 523–534. http://dl.acm.org/citation.cfm?id=2390948.2391009.

Ruslan Mitkov and Le An Ha. 2003. Computer-aided generation of multiple-choice tests. In *Proceedings of the HLT-NAACL 03 Workshop on Building Educational Applications Using Natural Language Processing - Volume 2*. Association for Computational Linguistics, Stroudsburg, PA, USA, HLT-NAACL-EDUC '03, pages 17–22. https://doi.org/10.3115/1118894.1118897.

Ruslan Mitkov, Le An Ha, Andrea Varga, and Luz Rello. 2009. Semantic similarity of distractors in multiple-choice tests: Extrinsic evaluation. In *Proceedings of the Workshop on Geometrical Models of Natural Language Semantics*. Association for Computational Linguistics, Stroudsburg, PA, USA, GEMS '09, pages 49–56. http://dl.acm.org/citation.cfm?id=1705415.1705422.

Jack Mostow and Hyeju Jang. 2012. Generating diagnostic multiple choice comprehension cloze questions. In *Proceedings of the Seventh Workshop on Building Educational Applications Using NLP*. Association for Computational Linguistics, Stroudsburg, PA, USA, pages 136–146. http://dl.acm.org/citation.cfm?id=2390384.2390401.

Tri Nguyen, Mir Rosenberg, Xia Song, Jianfeng Gao, Saurabh Tiwary, Rangan Majumder, and Li Deng. 2016. MS MARCO: A human generated machine reading comprehension dataset. *CoRR* abs/1611.09268. http://arxiv.org/abs/1611.09268.

Takeshi Onishi, Hai Wang, Mohit Bansal, Kevin Gimpel, and David A. McAllester. 2016. Who did what: A large-scale person-centered cloze dataset. In *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing, EMNLP 2016, Austin, Texas, USA, November 1-4, 2016*. pages 2230–2235. http://aclweb.org/anthology/D/D16/D16-1241.pdf.

Andreas Papasalouros, Konstantinos Kanaris, and Konstantinos Kotis. 2008. Automatic generation of multiple choice questions from domain ontologies. In Miguel Baptista Nunes and Maggie McPherson, editors, *e-Learning*. IADIS, pages 427–434.

Denis Paperno, Germán Kruszewski, Angeliki Lazaridou, Quan Ngoc Pham, Raffaella Bernardi, Sandro Pezzelle, Marco Baroni, Gemma Boleda, and Raquel Fernández. 2016. The lambada dataset: Word prediction requiring a broad discourse context. *arXiv preprint arXiv:1606.06031* .

Jeffrey Pennington, Richard Socher, and Christopher D. Manning. 2014. Glove: Global vectors for word representation. In *Empirical Methods in Natural Language Processing (EMNLP)*. pages 1532–1543. http://www.aclweb.org/anthology/D14-1162.

Juan Pino and Maxine Esknazi. 2009. Semi-automatic generation of cloze question distractors effect of students' l1. In *SLaTE*. ISCA, pages 65–68.

Juan Pino, Michael Heilman, and Maxine Eskenazi. 2008. A Selection Strategy to Improve Cloze Question Quality. In *Proceedings of the Workshop on Intelligent Tutoring Systems for Ill-Defined Domains. 9th International Conference on Intelligent Tutoring Systems.*.

P. Rajpurkar, J. Zhang, K. Lopyrev, and P. Liang. 2016. Squad: 100,000+ questions for machine comprehension of text. In *Empirical Methods in Natural Language Processing (EMNLP)*.

Mrinmaya Sachan, Avinava Dubey, and Eric P. Xing. 2016. Science question answering using instructional materials. *CoRR* abs/1602.04375. http://arxiv.org/abs/1602.04375.

Keisuke Sakaguchi, Yuki Arase, and Mamoru Komachi. 2013. Discriminative approach to fill-in-the-blank quiz generation for language learners. In *Proceedings of the 51st Annual Meeting of the Association for Computational Linguistics, ACL 2013, 4-9 August 2013, Sofia, Bulgaria, Volume 2: Short Papers*. pages 238–242. http://aclweb.org/anthology/P/P13/P13-2043.pdf.

Carissa Schoenick, Peter Clark, Oyvind Tafjord, Peter Turney, and Oren Etzioni. 2016. Moving beyond the turing test with the allen ai science challenge. *arXiv preprint arXiv:1604.04315* .

Min Joon Seo, Aniruddha Kembhavi, Ali Farhadi, and Hannaneh Hajishirzi. 2016. Bidirectional attention flow for machine comprehension. *CoRR* abs/1611.01603. http://arxiv.org/abs/1611.01603.

Alessandro Sordoni, Phillip Bachman, and Yoshua Bengio. 2016. Iterative alternating neural attention for machine reading. *CoRR* abs/1606.02245. http://arxiv.org/abs/1606.02245.

Eiichiro Sumita, Fumiaki Sugaya, and Seiichi Yamamoto. 2005. Measuring non-native speakers' proficiency of english by using a test with automatically-generated fill-in-the-blank questions. In *Proceedings of the Second Workshop on Building Educational Applications Using NLP*. Association for Computational Linguistics, Stroudsburg, PA, USA, EdAppsNLP 05, pages 61–68. http://dl.acm.org/citation.cfm?id=1609829.1609839.

Yi Yang, Scott Wen-tau Yih, and Chris Meek. 2015. Wikiqa: A challenge dataset for open-domain question answering. ACL Association for Computational Linguistics. https://www.microsoft.com/en-us/research/publication/wikiqa-a-challenge-dataset-for-open-domain-question-answering/.

Torsten Zesch and Oren Melamud. 2014. Automatic generation of challenging distractors using context-sensitive inference rules. In *Proceedings of the Ninth Workshop on Innovative Use of NLP for Building Educational Applications, BEA@ACL 2014, June 26, 2014, Baltimore, Maryland, USA*. pages 143–148. http://aclweb.org/anthology/W/W14/W14-1817.pdf.

A List of Study Books

The following is a list of the books we used as data source:

- OpenStax, Anatomy & Physiology. OpenStax. 25 April 2013[9]
- OpenStax, Biology. OpenStax. May 20, 2013[10]
- OpenStax, Chemistry. OpenStax. 11 March 2015[11]
- OpenStax, College Physics. OpenStax. 21 June 2012[12]
- OpenStax, Concepts of Biology. OpenStax. 25 April 2013[13]
- Biofundamentals 2.0 – by Michael Klymkowsky, University of Colorado & Melanie Cooper, Michigan State University[14]
- Earth Systems, An Earth Science Course on www.curriki.org[15]
- General Chemistry, Principles, Patterns, and Applications by Bruce Averill, Strategic Energy Security Solutions and Patricia Eldredge, R.H. Hand, LLC; Saylor Foundation[16]
- General Biology; Paul Doerder, Cleveland State University & Ralph Gibson, Cleveland State University [17]
- Introductory Chemistry by David W. Ball, Cleveland State University. Saylor Foundation [18]
- The Basics of General, Organic, and Biological Chemistry by David Ball, Cleveland State University & John Hill, University of Wisconsin & Rhonda Scott, Southern Adventist University. Saylor Foundation[19]
- Barron's New York State Grade 4 Elementary-Level Science Test, by Joyce Thornton Barry and Kathleen Cahill [20]
- Campbell Biology: Concepts & Connections by Jane B. Reece, Martha R. Taylor, Eric J. Simon, Jean L. Dickey[21]
- CK-12 Peoples Physics Book Basic [22]
- CK-12 Biology Advanced Concepts [23]
- CK-12 Biology Concepts [24]
- CK-12 Biology [25]
- CK-12 Chemistry - Basic [26]
- CK-12 Chemistry Concepts – Intermediate [27]
- CK-12 Earth Science Concepts For Middle School[28]
- CK-12 Earth Science Concepts For High School[29]

[9] Download for free at http://cnx.org/content/col11496/latest/

[10] Download for free at http://cnx.org/content/col11448/latest/

[11] Download for free at http://cnx.org/content/col11760/latest/

[12] Download for free at http://cnx.org/content/col11406/latest

[13] Download for free at http://cnx.org/content/col11487/latest

[14] https://open.umn.edu/opentextbooks/BookDetail.aspx?bookId=350

[15] http://www.curriki.org/xwiki/bin/view/Group_CLRN-OpenSourceEarthScienceCourse/

[16] https://www.saylor.org/site/textbooks/General%20Chemistry%20Principles,%20Patterns,%20and%20Applications.pdf

[17] https://upload.wikimedia.org/wikipedia/commons/4/40/GeneralBiology.pdf

[18] https://www.saylor.org/site/textbooks/Introductory%20Chemistry.pdf

[19] http://web.archive.org/web/20131024125808/http://www.saylor.org/site/textbooks/The%20Basics%20of%20General,%20Organic%20and%20Biological%20Chemistry.pdf

[20] We do not include documents from this resource in the dataset.

[21] We do not include documents from this resource in the dataset.

[22] http://www.ck12.org/book/Peoples-Physics-Book-Basic/

[23] http://www.ck12.org/book/CK-12-Biology-Advanced-Concepts/

[24] http://www.ck12.org/book/CK-12-Biology-Concepts/

[25] http://www.ck12.org/book/CK-12-Biology/

[26] http://www.ck12.org/book/CK-12-Chemistry-Basic/

[27] http://www.ck12.org/book/CK-12-Chemistry-Concepts-Intermediate/

[28] http://www.ck12.org/book/CK-12-Earth-Science-Concepts-For-Middle-School/

[29] http://www.ck12.org/book/CK-12-Earth-Science-Concepts-For-High-School/

- CK-12 Earth Science For Middle School [30]
- CK-12 Life Science Concepts For Middle School [31]
- CK-12 Life Science For Middle School [32]
- CK-12 Physical Science Concepts For Middle School[33]
- CK-12 Physical Science For Middle School [34]
- CK-12 Physics Concepts - Intermediate [35]
- CK-12 People's Physics Concepts [36]

CK-12 books were obtained under the Creative Commons Attribution-Non-Commercial 3.0 Unported (CC BY-NC 3.0) License [37].

B Training and Implementation Details

Multiple Choice Reading Comprehension. During training of the AS Reader and GA Reader, we monitored model performance after each epoch and stopped training when the error on the validation set had increased (early stopping, with a patience of one). We set a hard limit of ten epochs, but most models reached their peak validation accuracy after the first or second epoch. Test set evaluation, when applicable, used model parameters at the epoch of their peak validation accuracy. We implemented the models in Keras, and ran them with the Theano backend on a Tesla K80 GPU.

The hyperparameters for each of the models were adopted from previous work. For the AS Reader, we use an embedding dimension of 256 and GRU hidden layer dimension of 384 (obtained through correspondence with the authors of Onishi et al. (2016)) and use the hyperparameters reported in the original paper (Kadlec et al., 2016) for the rest. For the GA Reader, we use three gated-attention layers with the multiplicative gating mechanism. We do not use the character-level embedding features or the question-evidence common word features, but we do follow their work by using pretrained 100-dimension GloVe vectors to initialize a fixed word embedding layer. Between each gated attention layer, we apply dropout with a rate of 0.3. The other hyperparameters are the same as their original work (Dhingra et al., 2016).

Direct Answer Reading Comprehension. We implemented the Bidirectional Attention Flow model exactly as described in Seo et al. (2016) and adopted the hyperparameters used in the paper.

[30] http://www.ck12.org/book/CK-12-Earth-Science-For-Middle-School/
[31] http://www.ck12.org/book/CK-12-Life-Science-Concepts-For-Middle-School/
[32] http://www.ck12.org/book/CK-12-Life-Science-For-Middle-School/
[33] http://www.ck12.org/book/CK-12-Physical-Science-Concepts-For-Middle-School/
[34] http://www.ck12.org/book/CK-12-Physical-Science-For-Middle-School/
[35] http://www.ck12.org/book/CK-12-Physics-Concepts-Intermediate/
[36] http://www.ck12.org/book/Peoples-Physics-Concepts/
[37] http://creativecommons.org/licenses/by-nc/3.0/