Know What You Don't Know: Unanswerable Questions for SQuAD

Pranav Rajpurkar* **Robin Jia*** **Percy Liang**
Computer Science Department, Stanford University
{pranavsr,robinjia,pliang}@cs.stanford.edu

Abstract

Extractive reading comprehension systems can often locate the correct answer to a question in a context document, but they also tend to make unreliable guesses on questions for which the correct answer is not stated in the context. Existing datasets either focus exclusively on answerable questions, or use automatically generated unanswerable questions that are easy to identify. To address these weaknesses, we present SQuADRUn, a new dataset that combines the existing Stanford Question Answering Dataset (SQuAD) with over 50,000 unanswerable questions written adversarially by crowdworkers to look similar to answerable ones. To do well on SQuADRUn, systems must not only answer questions when possible, but also determine when no answer is supported by the paragraph and abstain from answering. SQuADRUn is a challenging natural language understanding task for existing models: a strong neural system that gets 86% F1 on SQuAD achieves only 66% F1 on SQuADRUn. We release SQuADRUn to the community as the successor to SQuAD.

Article: Endangered Species Act
Paragraph: "*. . . Other legislation followed, including the Migratory Bird Conservation Act of 1929, a 1937 treaty prohibiting the hunting of right and gray whales, and the Bald Eagle Protection Act of 1940. These later laws had a low cost to society—the species were relatively rare—and little opposition was raised.*"

Question 1: "*Which laws faced significant opposition?*"
Plausible Answer: *later laws*

Question 2: "*What was the name of the 1937 treaty?*"
Plausible Answer: *Bald Eagle Protection Act*

Figure 1: Two unanswerable questions written by crowdworkers, along with plausible (but incorrect) answers. Relevant keywords are shown in blue.

1 Introduction

Machine reading comprehension has become a central task in natural language understanding, fueled by the creation of many large-scale datasets (Hermann et al., 2015; Hewlett et al., 2016; Rajpurkar et al., 2016; Nguyen et al., 2016; Trischler et al., 2017; Joshi et al., 2017). In turn, these datasets have spurred a diverse array of model architecture improvements (Seo et al., 2016; Hu et al., 2017; Wang et al., 2017; Clark and Gardner, 2017; Huang et al., 2018). Recent work has even produced systems that surpass human-level exact match accuracy on the Stanford Question Answering Dataset (SQuAD), one of the most widely-used reading comprehension benchmarks (Rajpurkar et al., 2016).

Nonetheless, these systems are still far from true language understanding. Recent analysis shows that models can do well at SQuAD by learning context and type-matching heuristics (Weissenborn et al., 2017), and that success on SQuAD does not ensure robustness to distracting sentences (Jia and Liang, 2017). One root cause of these problems is SQuAD's focus on questions for which a correct answer is guaranteed to exist in the context document. Therefore, models only need to select the span that seems most related to the question, instead of checking that the answer is actually entailed by the text.

In this work, we construct SQuADRUn,[1] a new dataset that combines the existing questions in SQuAD with 53,775 new, unanswerable ques-

[*] The first two authors contributed equally to this paper.

[1] SQuAD with adve**R**sarial **Un**answerable questions

Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Short Papers), pages 784–789
Melbourne, Australia, July 15 - 20, 2018. ©2018 Association for Computational Linguistics

tions about the same paragraphs. Crowdworkers crafted these questions so that (1) they are *relevant* to the paragraph, and (2) the paragraph contains a *plausible answer*—something of the same type as what the question asks for. Two such examples are shown in Figure 1.

We confirm that SQUADRUN is both challenging and high-quality. A state-of-the-art model achieves only 66.3% F1 score when trained and tested on SQUADRUN, whereas human accuracy is 89.5% F1, a full 23.2 points higher. The same model architecture trained on SQuAD gets 85.8% F1, only 5.4 points worse than humans. We also show that our unanswerable questions are more challenging than ones created automatically, either via distant supervision (Clark and Gardner, 2017) or a rule-based method (Jia and Liang, 2017). We release SQUADRUN to the public as the successor to SQuAD, and designate it SQuAD 2.0 on the official SQuAD leaderboard.[2] We are optimistc that this new dataset will encourage the development of reading comprehension systems that know what they don't know.

2 Desiderata

We first outline our goals for SQUADRUN. Besides the generic goals of large size, diversity, and low noise, we posit two desiderata specific to unanswerable questions:

Relevance. The unanswerable questions should appear relevant to the topic of the context paragraph. Otherwise, simple heuristics (e.g., based on word overlap) could distinguish answerable and unanswerable questions (Yih et al., 2013).

Existence of plausible answers. There should be some span in the context whose type matches the type of answer the question asks for. For example, if the question asks, "*What company was founded in 1992?*", then some company should be mentioned in the context. Otherwise, type-matching heuristics could distinguish answerable and unanswerable questions (Weissenborn et al., 2017).

3 Existing datasets

Next, we survey existing reading comprehension datasets with these criteria in mind. We use the term "negative example" to refer to a context passage paired with an unanswerable question.

3.1 Extractive datasets

In extractive reading comprehension datasets, a system must extract the correct answer to a question from a context document or paragraph. The Zero-shot Relation Extraction dataset (Levy et al., 2017) contains negative examples generated with distant supervision. Levy et al. (2017) found that 65% of these negative examples do not have a plausible answer, making them easy to identify.

Other distant supervision strategies can also create negative examples. TriviaQA (Joshi et al., 2017) retrieves context documents from the web or Wikipedia for each question. Some documents do not contain the correct answer, yielding negative examples; however, these are excluded from the final dataset. Clark and Gardner (2017) generate negative examples for SQuAD by pairing existing questions with other paragraphs from the same article based on TF-IDF overlap; we refer to these as TFIDF examples. In general, distant supervision does not ensure the existence of a plausible answer in the retrieved context, and might also add noise, as the context might contain a paraphrase of the correct answer. Moreover, when retrieving from a small set of possible contexts, as in Clark and Gardner (2017), we find that the retrieved paragraphs are often not very relevant to the question, making these negative examples easy to identify.

The NewsQA data collection process also yields unanswerable questions, because crowdworkers write questions given only a summary of an article, not the full text (Trischler et al., 2017). Only 9.5% of their questions are unanswerable, making this strategy hard to scale. Of this fraction, we found that some are misannotated as unanswerable, and others are out-of-scope (e.g., summarization questions). Trischler et al. (2017) also exclude negative examples from their final dataset.

Jia and Liang (2017) propose a rule-based procedure for editing SQuAD questions to make them unanswerable. Their questions are not very diverse: they only replace entities and numbers with similar words, and replace nouns and adjectives with WordNet antonyms. We refer to these unanswerable questions as RULEBASED questions.

3.2 Answer sentence selection datasets

Sentence selection datasets test whether a system can rank sentences that answer a question higher

[2] As with previous versions of SQuAD, we release SQUADRUN under the CC BY-SA 4.0 license.

Reasoning	Description	Example	Percentage
Negation	Negation word inserted or removed.	Sentence: “*Several hospital pharmacies have decided to outsource high risk preparations . . .*” Question: “*What types of pharmacy functions have **never** been outsourced?*”	9%
Antonym	Antonym used.	S: “*the extinction of the dinosaurs. . . allowed the tropical rainforest to spread out across the continent.*” Q: “*The extinction of what led to the **decline** of rainforests?*”	20%
Entity Swap	Entity, number, or date replaced with other entity, number, or date.	S: “*These values are much greater than the 9–88 cm as projected . . . in its Third Assessment Report.*” Q: “*What was the projection of sea level increases in the **fourth assessment report**?*”	21%
Mutual Exclusion	Word or phrase is mutually exclusive with something for which an answer is present.	S: “*BSkyB. . . waiv[ed] the charge for subscribers whose package included two or more premium channels.*” Q: “*What service did BSkyB **give away for free unconditionally**?*”	15%
Impossible Condition	Asks for condition that is not satisfied by anything in the paragraph.	S: “*Union forces left Jacksonville and confronted a Confederate Army at the Battle of Olustee. . . Union forces then retreated to Jacksonville and held the city for the remainder of the war.*” Q: “*After what battle did Union forces leave Jacksonville **for good**?*”	4%
Other Neutral	Other cases where the paragraph does not imply any answer.	S: “*Schuenemann et al. concluded in 2011 that the Black Death. . . was caused by a variant of Y. pestis. . .*” Q: “*Who **discovered** Y. pestis?*”	24%
Answerable	Question is answerable (i.e. dataset noise).		7%

Table 1: Types of negative examples in SQuADRUn exhibiting a wide range of phenomena.

than sentences that do not. Wang et al. (2007) constructed the QASent dataset from questions in the TREC 8-13 QA tracks. Yih et al. (2013) showed that lexical baselines are highly competitive on this dataset. WikiQA (Yang et al., 2015) pairs questions from Bing query logs with sentences from Wikipedia. Like TFIDF examples, these sentences are not guaranteed to have plausible answers or high relevance to the question. The dataset is also limited in scale (3,047 questions, 1,473 answers).

3.3 Multiple choice datasets

Finally, some datasets, like MCTest (Richardson et al., 2013) and RACE (Lai et al., 2017), pose multiple choice questions, which can have a “none of the above” option. In practice, multiple choice options are often unavailable, making these datasets less suited for training user-facing systems. Multiple choice questions also tend to be quite different from extractive ones, with more emphasis on fill-in-the-blank, interpretation, and summarization (Lai et al., 2017).

4 The SQuADRUn dataset

We now describe our new dataset, which we constructed to satisfy both the relevance and plausible answer desiderata from Section 2.

4.1 Dataset creation

We employed crowdworkers on the Daemo crowdsourcing platform (Gaikwad et al., 2015) to write unanswerable questions. Each task consisted of an entire article from the original SQuAD dataset. For each paragraph in the article, workers were asked to pose up to five questions that were impossible to answer based on the paragraph alone, while referencing entities in the paragraph and ensuring that a plausible answer is present. As inspiration, we also showed questions from SQuAD for each paragraph; this further encouraged unanswerable questions to look similar to answerable ones. Workers were asked to spend 7 minutes per paragraph, and were paid $10.50 per hour. Screenshots of our interface are shown in Appendix A.1.

We removed questions from workers who wrote 25 or fewer questions on that article; this filter helped remove noise from workers who had trouble understanding the task, and therefore quit before completing the whole article. We applied this filter to both our new data and the existing answerable questions in SQuAD. To generate train, development, and test splits, we used the same partition of articles as SQuAD, and combined the existing SQuAD data with our new data for each split. For the SQuADRUn development and test sets, we removed articles for which we did not col-

	SQuAD	SQuADRUn
Train		
Total examples	87,599	130,319
Negative examples	0	43,498
Total articles	442	442
Articles with negatives	0	285
Development		
Total examples	10,570	11,873
Negative examples	0	5,945
Total articles	48	35
Articles with negatives	0	35
Test		
Total examples	9,533	8,862
Negative examples	0	4,332
Total articles	46	28
Articles with negatives	0	28

Table 2: Dataset statistics of SQuADRUn, compared to the original SQuAD dataset.

lect unanswerable questions. This resulted in a roughly one-to-one ratio of answerable to unanswerable questions in these splits, whereas the train data has roughly twice as many answerable questions as unanswerable ones. Table 2 summarizes overall statistics of SQuADRUn.

4.2 Human accuracy

To confirm that our dataset is clean, we hired additional crowdworkers to answer all questions in the SQuADRUn development and test sets. In each task, we showed workers an entire article from the dataset. For each paragraph, we showed all associated questions; unanswerable and answerable questions were shuffled together. For each question, workers were told to either highlight the answer in the paragraph, or mark it as unanswerable. Workers were told to expect every paragraph to have some answerable and some unanswerable questions. They were asked to spend one minute per question, and were paid $10.50 per hour.

To reduce crowdworker noise, we collected multiple human answers for each question and selected the final answer by majority vote, breaking ties in favor of answering questions and preferring shorter answers to longer ones. On average, we collected 4.8 answers per question. We note that for the original SQuAD, Rajpurkar et al. (2016) evaluated a single human's performance; therefore, they likely underestimate human accuracy.

4.3 Analysis

We manually inspected 100 randomly chosen negative examples from our development set to understand the challenges these examples present. In Table 1, we define different categories of negative examples, and give examples and their frequency in SQuADRUn. We observe a wide range of phenomena, extending beyond expected phenomena like negation, antonymy, and entity changes. In particular, SQuADRUn is much more diverse than RuleBased, which creates unanswerable questions by applying entity, number, and antonym swaps to existing SQuAD questions. We also found that 93% of the sampled negative examples are indeed unanswerable.

5 Experiments

5.1 Models

We evaluated three existing model architectures: the BiDAF-No-Answer (BNA) model proposed by Levy et al. (2017), and two versions of the DocumentQA No-Answer (DocQA) model from Clark and Gardner (2017), namely versions with and without ELMo (Peters et al., 2018). These models all learn to predict the probability that a question is unanswerable, in addition to a distribution over answer choices. At test time, models abstain whenever their predicted probability that a question is unanswerable exceeds some threshold. We tune this threshold separately for each model on the development set. When evaluating on the test set, we use the threshold that maximizes F1 score on the development set. We find this strategy does slightly better than simply taking the argmax prediction, possibly due to the different proportions of negative examples at training and test time.

5.2 Main results

First, we trained and tested all three models on SQuADRUn, as shown in Table 3. Following Rajpurkar et al. (2016), we report average exact match and F1 scores.[3] The best model, DocQA + ELMo, achieves only 66.3 F1 on the test set, 23.2 points lower than the human accuracy of 89.5 F1. Note that a baseline that always abstains gets 48.9 test F1; existing models are closer to this baseline than they are to human performance. Therefore, we see significant room for model improvement on this task. We also compare with reported test numbers for analogous model architectures on SQuAD. There is a much larger gap between humans and machines on SQuADRUn compared to SQuAD, which confirms that SQuADRUn is a much harder dataset for existing models.

[3] For negative examples, abstaining receives a score of 1, and any other response gets 0, for both exact match and F1.

System	SQuAD test		SQuADRUn dev		SQuADRUn test	
	EM	F1	EM	F1	EM	F1
BNA	68.0	77.3	59.8	62.6	59.2	62.1
DocQA	72.1	81.0	61.9	64.8	59.3	62.3
DocQA + ELMo	**78.6**	**85.8**	**65.1**	**67.6**	**63.4**	**66.3**
Human	82.3	91.2	86.3	89.0	86.9	89.5
Human–Machine Gap	3.7	5.4	**21.2**	**21.4**	**23.5**	**23.2**

Table 3: Exact Match (EM) and F1 scores on SQuADRUn and SQuAD. The gap between humans and the best tested model is much larger on SQuADRUn, suggesting there is a great deal of room for model improvement.

System	SQuAD + TfIdf		SQuAD + RuleBased		SQuADRUn dev	
	EM	F1	EM	F1	EM	F1
BNA	72.7	76.6	80.1	84.8	59.8	62.6
DocQA	75.6	79.2	80.8	84.8	61.9	64.8
DocQA + ELMo	**79.4**	**83.0**	**85.7**	**89.6**	**65.1**	**67.6**

Table 4: Exact Match (EM) and F1 scores on the SQuADRUn development set, compared with SQuAD with two types of automatically generated negative examples. SQuADRUn is more challenging for current models.

5.3 Automatically generated negatives

Next, we investigated whether automatic ways of generating negative examples can also yield a challenging dataset. We trained and tested all three model architectures on SQuAD augmented with either TfIdf or RuleBased examples. To ensure a fair comparison with SQuADRUn, we generated training data by applying TfIdf or RuleBased only to the 285 articles for which SQuADRUn has unanswerable questions. We tested on the articles and answerable questions in the SQuADRUn development set, adding unanswerable questions in a roughly one-to-one ratio with answerable ones. These results are shown in Table 4. The highest score on SQuADRUn is 15.4 F1 points lower than the highest score on either of the other two datasets, suggesting that automatically generated negative examples are much easier for existing models to detect.

5.4 Plausible answers as distractors

Finally, we measured how often systems were fooled into answering the plausible but incorrect answers provided by crowdworkers for our unanswerable questions. For both computer systems and humans, roughly half of all wrong answers on unanswerable questions exactly matched the plausible answers. This suggests that the plausible answers do indeed serve as effective distractors. Full results are shown in Appendix A.2.

6 Discussion

SQuADRUn forces models to understand whether a paragraph entails that a certain span is the answer to a question. Similarly, recognizing textual entailment (RTE) requires systems to decide whether a hypothesis is entailed by, contradicted by, or neutral with respect to a premise (Marelli et al., 2014; Bowman et al., 2015). Relation extraction systems must understand when a possible relationship between two entities is not entailed by the text (Zhang et al., 2017).

Jia and Liang (2017) created adversarial examples that fool pre-trained SQuAD models at test time. However, models that train on similar examples are not easily fooled by their method. In contrast, the adversarial examples in SQuADRUn are difficult even for models trained on examples from the same distribution.

In conclusion, we have presented SQuADRUn, a challenging, diverse, and large-scale dataset that forces models to understand when a question cannot be answered given the context. We are optimistic that SQuADRUn will encourage the development of new reading comprehension models that know what they don't know, and therefore understand language at a deeper level.

Reproducibility. All code, data, experiments are available on the Codalab platform at https://bit.ly/2rDHBgY.

Acknowledgments. We would like to thank the anonymous reviewers, Arun Chaganty, Peng Qi, and Sharon Zhou for their constructive feedback. We are grateful to Durim Morina and Michael Bernstein for their help with the Daemo platform. This work was supported by funding from Facebook. R.J. is supported by an NSF Graduate Research Fellowship under Grant No. DGE-114747.

References

S. Bowman, G. Angeli, C. Potts, and C. D. Manning. 2015. A large annotated corpus for learning natural language inference. In *Empirical Methods in Natural Language Processing (EMNLP)*.

C. Clark and M. Gardner. 2017. Simple and effective multi-paragraph reading comprehension. *arXiv preprint arXiv:1710.10723* .

S. N. Gaikwad, D. Morina, R. Nistala, M. Agarwal, A. Cossette, R. Bhanu, S. Savage, V. Narwal, K. Rajpal, J. Regino, et al. 2015. Daemo: A self-governed crowdsourcing marketplace. In *Proceedings of the 28th Annual ACM Symposium on User Interface Software & Technology*. pages 101–102.

K. M. Hermann, T. Koisk, E. Grefenstette, L. Espeholt, W. Kay, M. Suleyman, and P. Blunsom. 2015. Teaching machines to read and comprehend. In *Advances in Neural Information Processing Systems (NIPS)*.

D. Hewlett, A. Lacoste, L. Jones, I. Polosukhin, A. Fandrianto, J. Han, M. Kelcey, and D. Berthelot. 2016. Wikireading: A novel large-scale language understanding task over Wikipedia. In *Association for Computational Linguistics (ACL)*.

M. Hu, Y. Peng, and X. Qiu. 2017. Reinforced mnemonic reader for machine comprehension. *arXiv* .

H. Huang, C. Zhu, Y. Shen, and W. Chen. 2018. Fusionnet: Fusing via fully-aware attention with application to machine comprehension. In *International Conference on Learning Representations (ICLR)*.

R. Jia and P. Liang. 2017. Adversarial examples for evaluating reading comprehension systems. In *Empirical Methods in Natural Language Processing (EMNLP)*.

M. Joshi, E. Choi, D. Weld, and L. Zettlemoyer. 2017. TriviaQA: A large scale distantly supervised challenge dataset for reading comprehension. In *Association for Computational Linguistics (ACL)*.

G. Lai, Q. Xie, H. Liu, Y. Yang, and E. Hovy. 2017. Race: Large-scale reading comprehension dataset from examinations. *arXiv preprint arXiv:1704.04683* .

O. Levy, M. Seo, E. Choi, and L. Zettlemoyer. 2017. Zero-shot relation extraction via reading comprehension. In *Computational Natural Language Learning (CoNLL)*.

M. Marelli, S. Menini, M. Baroni, L. Bentivogli, R. bernardi, and R. Zamparelli. 2014. A SICK cure for the evaluation of compositional distributional semantic models. In *Language Resources and Evaluation Conference (LREC)*.

T. Nguyen, M. Rosenberg, X. Song, J. Gao, S. Tiwary, R. Majumder, and L. Deng. 2016. MS MARCO: A human generated machine reading comprehension dataset. In *Workshop on Cognitive Computing at NIPS*.

M. E. Peters, M. Neumann, M. Iyyer, M. Gardner, C. Clark, K. Lee, and L. Zettlemoyer. 2018. Deep contextualized word representations. In *North American Association for Computational Linguistics (NAACL)*.

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)*.

M. Richardson, C. J. Burges, and E. Renshaw. 2013. Mctest: A challenge dataset for the open-domain machine comprehension of text. In *Empirical Methods in Natural Language Processing (EMNLP)*. pages 193–203.

M. Seo, A. Kembhavi, A. Farhadi, and H. Hajishirzi. 2016. Bidirectional attention flow for machine comprehension. *arXiv* .

A. Trischler, T. Wang, X. Yuan, J. Harris, A. Sordoni, P. Bachman, and K. Suleman. 2017. NewsQA: A machine comprehension dataset. In *Workshop on Representation Learning for NLP*.

M. Wang, N. A. Smith, and T. Mitamura. 2007. What is the jeopardy model? a quasi-synchronous grammar for QA. In *Empirical Methods in Natural Language Processing (EMNLP)*.

W. Wang, N. Yang, F. Wei, B. Chang, and M. Zhou. 2017. Gated self-matching networks for reading comprehension and question answering. In *Association for Computational Linguistics (ACL)*.

D. Weissenborn, G. Wiese, and L. Seiffe. 2017. Making neural QA as simple as possible but not simpler. In *Computational Natural Language Learning (CoNLL)*.

Y. Yang, W. Yih, and C. Meek. 2015. WikiQA: A challenge dataset for open-domain question answering. In *Empirical Methods in Natural Language Processing (EMNLP)*. pages 2013–2018.

W. Yih, M. Chang, C. Meek, and A. Pastusiak. 2013. Question answering using enhanced lexical semantic models. In *Association for Computational Linguistics (ACL)*.

Y. Zhang, V. Zhong, D. Chen, G. Angeli, and C. D. Manning. 2017. Position-aware attention and supervised data improve slot filling. In *Empirical Methods in Natural Language Processing (EMNLP)*.