Beat the AI: Investigating Adversarial Human Annotation for Reading Comprehension

Max Bartolo Alastair Roberts Johannes Welbl Sebastian Riedel Pontus Stenetorp

Department of Computer Science
University College London
{m.bartolo,a.roberts,j.welbl,s.riedel,p.stenetorp}@cs.ucl.ac.uk

Abstract

Innovations in annotation methodology have been a catalyst for Reading Comprehension (RC) datasets and models. One recent trend to challenge current RC models is to involve a model in the annotation process: Humans create questions adversarially, such that the model fails to answer them correctly. In this work we investigate this annotation methodology and apply it in three different settings, collecting a total of 36,000 samples with progressively stronger models in the annotation loop. This allows us to explore questions such as the reproducibility of the adversarial effect, transfer from data collected with varying model-in-the-loop strengths, and generalization to data collected without a model. We find that training on adversarially collected samples leads to strong generalization to non-adversarially collected datasets, yet with progressive performance deterioration with increasingly stronger models-in-the-loop. Furthermore, we find that stronger models can still learn from datasets collected with substantially weaker models-in-the-loop. When trained on data collected with a BiDAF model in the loop, RoBERTa achieves 39.9F$_1$ on questions that it cannot answer when trained on SQuAD—only marginally lower than when trained on data collected using RoBERTa itself (41.0F$_1$).

1 Introduction

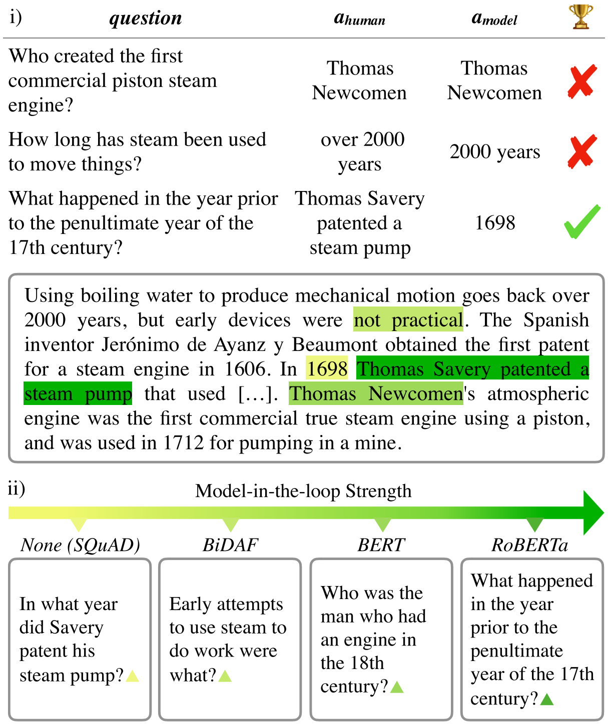
Data collection is a fundamental prerequisite for Machine Learning-based approaches to Natural Language Processing (NLP). Innovations in data acquisition methodology, such as crowdsourcing, have led to major breakthroughs in scalability and preceded the ‘‘deep learning revolution’’, for which they can arguably be seen as co-responsible (Deng et al., 2009; Bowman et al., 2015; Rajpurkar et al., 2016). Annotation approaches include expert annotation, for example, relying on trained linguists (Marcus et al., 1993), crowd-sourcing by non-experts (Snow et al., 2008), distant supervision (Mintz et al., 2009; Joshi et al., 2017), and leveraging document structure (Hermann et al., 2015). The concrete data collection paradigm chosen dictates the degree of scalability, annotation cost, precise task structure (often arising as a compromise of the above) and difficulty, domain coverage, as well as resulting dataset biases and model blind spots (Jia and Liang, 2017; Schwartz et al., 2017; Gururangan et al., 2018).

A recently emerging trend in NLP dataset creation is the use of a *model-in-the-loop* when composing samples: A contemporary model is used either as a filter or directly during annotation, to identify samples wrongly predicted by the model. Examples of this method are realized in *Build It Break It, The Language Edition* (Ettinger et al., 2017), HotpotQA (Yang et al., 2018a), SWAG (Zellers et al., 2018), Mechanical Turker Descent (Yang et al., 2018b), DROP (Dua et al., 2019), CODAH (Chen et al., 2019), Quoref (Dasigi et al., 2019), and AdversarialNLI (Nie et al., 2019).[1] This approach probes model robustness and ensures that the resulting datasets pose a challenge to current models, which drives research to tackle new sets of problems.

We study this approach in the context of Reading Comprehension (RC), and investigate its robustness in the face of continuously progressing models—do adversarially constructed datasets quickly become outdated in their usefulness as models grow stronger?

[1] The idea was alluded to at least as early as Richardson et al. (2013), but it has only recently seen wider adoption.

Transactions of the Association for Computational Linguistics, vol. 8, pp. 662–678, 2020. https://doi.org/10.1162/tacl_a_00338
Action Editor: Christopher Potts. Submission batch: 3/2020; Revision batch: 6/2020; Published 10/2020.



i) *question*	a_{human}	a_{model}	🏆
Who created the first commercial piston steam engine?	Thomas Newcomen	Thomas Newcomen	✗
How long has steam been used to move things?	over 2000 years	2000 years	✗
What happened in the year prior to the penultimate year of the 17th century?	Thomas Savery patented a steam pump	1698	✓

Using boiling water to produce mechanical motion goes back over 2000 years, but early devices were not practical. The Spanish inventor Jerónimo de Ayanz y Beaumont obtained the first patent for a steam engine in 1606. In 1698 Thomas Savery patented a steam pump that used […]. Thomas Newcomen's atmospheric engine was the first commercial true steam engine using a piston, and was used in 1712 for pumping in a mine.

ii)
Model-in-the-loop Strength
None (SQuAD)
BiDAF
BERT
RoBERTa
In what year did Savery patent his steam pump?
Early attempts to use steam to do work were what?
Who was the man who had an engine in the 18th century?
What happened in the year prior to the penultimate year of the 17th century?

Figure 1: Human annotation with a model in the loop, showing: i) the ''Beat the AI'' annotation setting where only questions that the model does not answer correctly are accepted, and ii) questions generated this way, with a progressively stronger model in the annotation loop.

Based on models trained on the widely used SQuAD dataset, and following the same annotation protocol, we investigate the annotation setup where an annotator has to compose questions for which the model predicts the wrong answer. As a result, only samples that the model fails to predict correctly are retained in the dataset—see Figure 1 for an example.

We apply this annotation strategy with three distinct models in the loop, resulting in datasets with 12,000 samples each. We then study the reproducibility of the adversarial effect when retraining the models with the same data, as well as the generalization ability of models trained using datasets produced with and without a model adversary. Models can, to a considerable degree, learn to generalize to more challenging questions, based on training sets collected with both stronger and also weaker models in the loop. Compared to training on SQuAD, training on adversarially composed questions leads to a similar degree of generalization to non-adversarially written questions, both for SQuAD and NaturalQuestions (Kwiatkowski et al., 2019). It furthermore leads to general improvements across the model-in-the-loop datasets we collect, as well as improvements of more than 20.0F_1 for both BERT and RoBERTa on an extractive subset of DROP (Dua et al., 2019), another adversarially composed dataset. When conducting a systematic analysis of the concrete questions different models fail to answer correctly, as well as non-adversarially composed questions, we see that the nature of the resulting questions changes: Questions composed with a model in the loop are overall more diverse, use more paraphrasing, multi-hop inference, comparisons, and background knowledge, and are generally less easily answered by matching an explicit statement that states the required information literally. Given our observations, we believe a model-in-the-loop approach to annotation shows promise and should be considered when creating future RC datasets.

To summarize, our contributions are as follows: First, an investigation into the model-in-the-loop approach to RC data collection based on three progressively stronger models, together with an empirical performance comparison when trained on datasets constructed with adversaries of different strength. Second, a comparative investigation into the nature of questions composed to be unsolvable by a sequence of progressively stronger models. Third, a study of the reproducibility of the adversarial effect and the generalization ability of models trained in various settings.

2 Related Work

Constructing Challenging Datasets Recent efforts in dataset construction have driven considerable progress in RC, yet datasets are structurally diverse and annotation methodologies vary. With its large size and combination of free-form questions with answers as extracted spans, SQuAD1.1 (Rajpurkar et al., 2016) has become an established benchmark that has inspired the construction of a series of similarly structured datasets. However, mounting evidence suggests that models can achieve strong generalization performance merely by relying on superficial cues—such as lexical overlap, term frequencies, or entity type matching (Chen et al., 2016; Weissenborn et al., 2017; Sugawara et al., 2018). It has thus become an increasingly important consideration to construct datasets that RC models find challenging, and for which natural language understanding is a requisite for generalization. Attempts to achieve this non-trivial aim have typically revolved around extensions to the SQuAD dataset annotation methodology. They include unanswerable questions (Trischler et al., 2017; Rajpurkar et al., 2018; Reddy et al., 2019; Choi et al., 2018), adding the option of ''Yes'' or ''No'' answers (Dua et al., 2019; Kwiatkowski et al., 2019), questions requiring reasoning over multiple sentences or documents (Welbl et al., 2018; Yang et al., 2018a), questions requiring rule interpretation or context awareness (Saeidi et al., 2018; Choi et al., 2018; Reddy et al., 2019), limiting annotator passage exposure by sourcing questions first (Kwiatkowski et al., 2019), controlling answer types by including options for dates, numbers, or spans from the question (Dua et al., 2019), as well as questions with free-form answers (Nguyen et al., 2016; Kočiský et al., 2018; Reddy et al., 2019).

Adversarial Annotation One recently adopted approach to constructing challenging datasets involves the use of an adversarial model to select examples that it does not perform well on, an approach which superficially is akin to active learning (Lewis and Gale, 1994). Here, we make a distinction between two sub-categories of adversarial annotation: i) *adversarial filtering*, where the adversarial model is applied offline in a separate stage of the process, usually after data generation; examples include SWAG (Zellers et al., 2018), ReCoRD (Zhang et al., 2018), HotpotQA (Yang et al., 2018a), and HellaSWAG (Zellers et al., 2019); ii) *model-in-the-loop adversarial annotation*, where the annotator can directly interact with the adversary during the annotation process and uses the feedback to further inform the generation process; examples include CODAH (Chen et al., 2019), Quoref (Dasigi et al., 2019), DROP (Dua et al., 2019), FEVER2.0 (Thorne et al., 2019), AdversarialNLI (Nie et al., 2019), as well as work by Dinan et al. (2019), Kaushik et al. (2020), and Wallace et al. (2019) for the Quizbowl task.

We are primarily interested in the latter category, as this feedback loop creates an environment where the annotator can probe the model directly to explore its weaknesses and formulate targeted adversarial attacks. Although Dua et al. (2019) and Dasigi et al. (2019) make use of adversarial annotations for RC, both annotation setups limit the reach of the model-in-the-loop: In DROP, primarily due to the imposition of specific answer types, and in Quoref by focusing on coreference, which is already a known RC model weakness.

In contrast, we investigate a scenario where annotators interact with a model in its original task setting—annotators must thus explore a range of natural adversarial attacks, as opposed to filtering out ''easy'' samples during the annotation process.

3 Annotation Methodology

3.1 Annotation Protocol

The data annotation protocol is based on SQuAD1.1, with a model in the loop, and the additional instruction that questions should only have one answer in the passage, which directly mirrors the setting in which these models were trained.

Formally, provided with a passage p, a human annotator generates a question q and selects a (human) answer a_h by highlighting the corresponding span in the passage. The input (p, q) is then given to the model, which returns a predicted (model) answer a_m. To compare the two, a word-overlap F_1 score between a_h and a_m is computed; a score above a threshold of 40% is considered a ''win'' for the model.[2] This process is repeated until the human ''wins''; Figure 2 gives a schematic overview of the process. All successful (p, q, a_h) triples, that is, those which the model is unable to answer correctly, are then retained for further validation.

3.2 Annotation Details

Models in the Annotation Loop We begin by training three different models, which are used as adversaries during data annotation. As a seed dataset for training the models we select the widely used SQuAD1.1 (Rajpurkar et al., 2016) dataset, a large-scale resource for which a variety of mature and well-performing models are readily available. Furthermore, unlike cloze-based datasets, SQuAD is robust to passage/question-only adversarial attacks (Kaushik and Lipton, 2018). We will compare dataset annotation with a series of three progressively stronger models as adversary in the loop, namely, BiDAF (Seo

[2] This threshold is set after initial experiments to not be overly restrictive given acceptable answer spans, e.g., a human answer of ''New York'' vs. model answer ''New York City'' would still lead to a model ''win''.

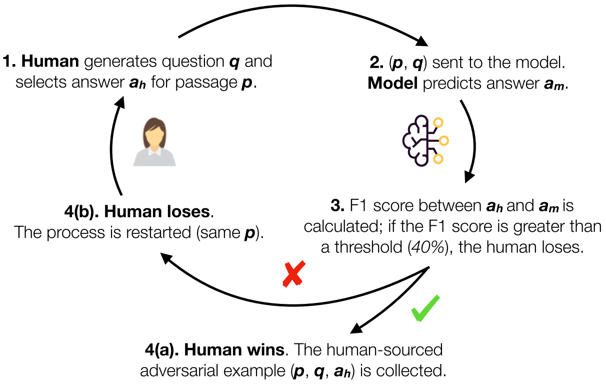

1. Human generates question q and selects answer a_h for passage p.
2. (p, q) sent to the model. Model predicts answer a_m.
3. F1 score between a_h and a_m is calculated; if the F1 score is greater than a threshold (40%), the human loses.
4(a). Human wins. The human-sourced adversarial example (p, q, a_h) is collected.
4(b). Human loses. The process is restarted (same p).

Figure 2: Overview of the annotation process to collect adversarially written questions from humans using a model in the loop.

et al., 2017), BERT$_{\text{LARGE}}$ (Devlin et al., 2019), and RoBERTa$_{\text{LARGE}}$ (Liu et al., 2019b). Each of these will serve as a model adversary in a separate annotation experiment and result in three distinct datasets; we will refer to these as $\mathcal{D}_{\text{BiDAF}}$, $\mathcal{D}_{\text{BERT}}$, and $\mathcal{D}_{\text{RoBERTa}}$ respectively. Examples from the validation set of each are shown in Table 1. We rely on the *AllenNLP* (Gardner et al., 2018) and *Transformers* (Wolf et al., 2019) model implementations, and our models achieve EM/F$_1$ scores of 65.5%/77.5%, 82.7%/90.3% and 86.9%/93.6% for BiDAF, BERT, and RoBERTa, respectively, on the SQuAD1.1 validation set, consistent with results reported in other work.

Our choice of models reflects both the transition from LSTM-based to pre-trained transformer-based models, as well as a graduation among the latter; we investigate how this is reflected in datasets collected with each of these different models in the annotation loop. For each of the models we collect 10,000 training, 1,000 validation, and 1,000 test examples. Dataset sizes are motivated by the data efficiency of transformer-based pretrained models (Devlin et al., 2019; Liu et al., 2019b), which has improved the viability of smaller-scale data collection efforts for investigative and analysis purposes.

To ensure the experimental integrity provided by reporting all results on a held-out test set, we split the existing SQuAD1.1 validation set in half (stratified by document title) as the official test set is not publicly available. We maintain passage consistency across the training, validation and test sets of all datasets to enable like-for-like comparisons. Lastly, we use the majority vote answer as ground truth for SQuAD1.1 to ensure that all our datasets have one valid answer per question, enabling us to fairly draw direct comparisons. For clarity, we will hereafter refer to this modified version of SQuAD1.1 as $\mathcal{D}_{\text{SQuAD}}$.

Crowdsourcing We use custom-designed Human Intelligence Tasks (HITs) served through Amazon Mechanical Turk (AMT) for all annotation efforts. Workers are required to be based in Canada, the UK, or the US, have a HIT Approval Rate greater than 98%, and have previously completed at least 1,000 HITs successfully. We experiment with and without the AMT *Master* requirement and find no substantial difference in quality, but observe a throughput reduction of nearly 90%. We pay USD 2.00 for every question generation HIT, during which workers are required to compose up to five questions that ‘‘beat’’ the model in the loop (cf. Figure 3). The mean HIT completion times for BiDAF, BERT, and RoBERTa are 551.8s, 722.4s, and 686.4s. Furthermore, we find that human workers are able to generate questions that successfully ‘‘beat’’ the model in the loop 59.4% of the time for BiDAF, 47.1% for BERT, and 44.0% for RoBERTa. These metrics broadly reflect the relative strength of the models.

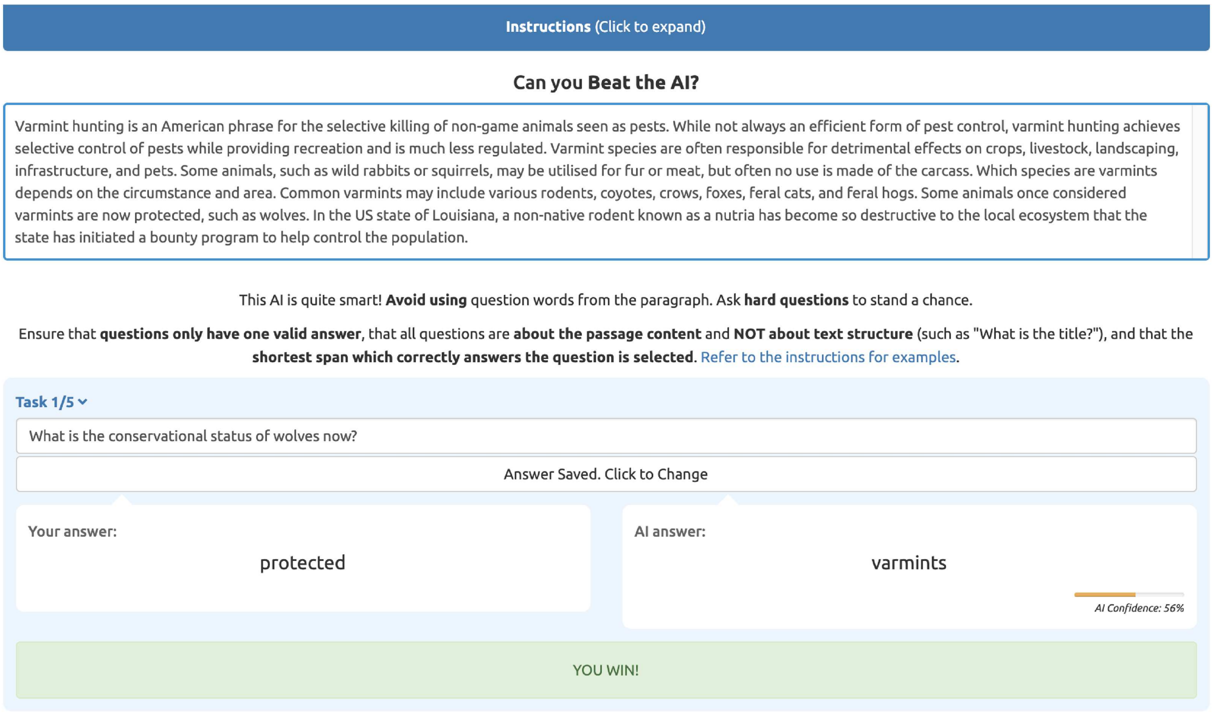
3.3 Quality Control

Training and Qualification We provide a two-part worker training interface in order to i) familiarize workers with the process, and ii) conduct a first screening based on worker outputs. The interface familiarizes workers with formulating questions, and answering them through span selection. Workers are asked to generate questions for two given answers, to highlight answers for two given questions, to generate one full question-answer pair, and finally to complete a question generation HIT with BiDAF as the model in the loop. Each worker's output is then reviewed manually (by the authors); those who pass the screening are added to the pool of qualified annotators.

Manual Worker Validation In the second annotation stage, qualified workers produce data for the ‘‘Beat the AI’’ question generation task. A sample of every worker's HITs is manually reviewed based on their total number of completed tasks n, determined by $\lfloor 5 \cdot \log_{10}(n) + 1 \rfloor$, chosen for

BiDAF	**Passage:** *[. . .] the United Methodist Church has placed great emphasis on the importance of education. As such, the United Methodist Church established and is affiliated with around one hundred colleges [. . .] of Methodist-related Schools, Colleges, and Universities. The church operates three hundred sixty schools and institutions overseas.* **Question:** *The United Methodist Church has how many schools internationally?*
BiDAF	**Passage:** *In a purely capitalist mode of production (i.e. where professional and labor organizations cannot limit the number of workers) the workers wages will not be controlled by these organizations, or by the employer, but rather by the market. Wages work in the same way as prices for any other good. Thus, wages can be considered as a [. . .]* **Question:** *What determines worker wages?*
BiDAF	**Passage:** *[. . .] released to the atmosphere, and a separate source of water feeding the boiler is supplied. Normally water is the fluid of choice due to its favourable properties, such as non-toxic and unreactive chemistry, abundance, low cost, and its thermodynamic properties. Mercury is the working fluid in the mercury vapor turbine [. . .]* **Question:** *What is the most popular type of fluid?*
BERT	**Passage:** *[. . .] Jochi was secretly poisoned by an order from Genghis Khan. Rashid al-Din reports that the great Khan sent for his sons in the spring of 1223, and while his brothers heeded the order, Jochi remained in Khorasan. Juzjani suggests that the disagreement arose from a quarrel between Jochi and his brothers in the siege of Urgench [. . .]* **Question:** *Who went to Khan after his order in 1223?*
BERT	**Passage:** *In the Sandgate area, to the east of the city and beside the river, resided the close-knit community of keelmen and their families. They were so called because [. . .] transfer coal from the river banks to the waiting colliers, for export to London and elsewhere. In the 1630s about 7,000 out of 20,000 inhabitants of Newcastle died of plague [. . .]* **Question:** *Where did almost half the people die?*
BERT	**Passage:** *[. . .] was important to reduce the weight of coal carried. Steam engines remained the dominant source of power until the early 20th century, when advances in the design of electric motors and internal combustion engines gradually resulted in the replacement of reciprocating (piston) steam engines, with shipping in the 20th-century [. . .]* **Question:** *Why did steam engines become obsolete?*
RoBERTa	**Passage:** *[. . .] and seven other hymns were published in the Achtliederbuch, the first Lutheran hymnal. In 1524 Luther developed his original four-stanza psalm paraphrase into a five-stanza Reformation hymn that developed the theme of "grace alone" more fully. Because it expressed essential Reformation doctrine, this expanded version of "Aus [. . .]* **Question:** *Luther's reformed hymn did not feature stanzas of what quantity?*
RoBERTa	**Passage:** *[. . .] tight end Greg Olsen, who caught a career-high 77 passes for 1,104 yards and seven touchdowns, and wide receiver Ted Ginn, Jr., who caught 44 passes for 739 yards and 10 touchdowns; [. . .] receivers included veteran Jerricho Cotchery (39 receptions for 485 yards), rookie Devin Funchess (31 receptions for 473 yards and [. . .]* **Question:** *Who caught the second most passes?*
RoBERTa	**Passage:** *Other prominent alumni include anthropologists David Graeber and Donald Johanson, who is best known for discovering the fossil of a female hominid australopithecine known as "Lucy" in the Afar Triangle region, psychologist John B. Watson, American psychologist who established the psychological school of behaviorism, communication theorist Harold Innis, chess grandmaster Samuel Reshevsky, and conservative international relations scholar and White House Coordinator of Security Planning for the National Security Council Samuel P. Huntington.* **Question:** *Who thinks three moves ahead?*

Table 1: Validation set examples of questions collected using different RC models (BiDAF, BERT, and RoBERTa) in the annotation loop. The answer to the question is highlighted in the passage.



Can you Beat the AI?

Varmint hunting is an American phrase for the selective killing of non-game animals seen as pests. While not always an efficient form of pest control, varmint hunting achieves selective control of pests while providing recreation and is much less regulated. Varmint species are often responsible for detrimental effects on crops, livestock, landscaping, infrastructure, and pets. Some animals, such as wild rabbits or squirrels, may be utilised for fur or meat, but often no use is made of the carcass. Which species are varmints depends on the circumstance and area. Common varmints may include various rodents, coyotes, crows, foxes, feral cats, and feral hogs. Some animals once considered varmints are now protected, such as wolves. In the US state of Louisiana, a non-native rodent known as a nutria has become so destructive to the local ecosystem that the state has initiated a bounty program to help control the population.

This AI is quite smart! **Avoid using** question words from the paragraph. Ask **hard questions** to stand a chance.

Ensure that **questions only have one valid answer**, that all questions are **about the passage content** and **NOT about text structure** (such as "What is the title?"), and that the **shortest span which correctly answers the question is selected**. Refer to the instructions for examples.

Task 1/5

What is the conservational status of wolves now?

Your answer: protected

AI answer: varmints

AI Confidence: 56%

Figure 3: ‘‘Beat the AI’’ question generation interface. Human annotators are tasked with asking questions about a provided passage that the model in the loop fails to answer correctly.

convenience. This is done after every annotation batch; if workers fall below an 80% success threshold at any point, their qualification is revoked and their work is discarded in its entirety.

Question Answerability As the models used in the annotation task become stronger, the resulting questions tend to become more complex. However, this also means that it becomes more challenging to disentangle measures of dataset quality from inherent question difficulty. As such, we use the condition of human answerability for an annotated question-answer pair as follows: It is answerable if at least one of three additional non-expert human validators can provide an answer matching the original. We conduct answerability checks on both the validation and test sets, and achieve answerability scores of 87.95%, 85.41%, and 82.63% for $\mathcal{D}_{\text{BiDAF}}$, $\mathcal{D}_{\text{BERT}}$, and $\mathcal{D}_{\text{RoBERTa}}$. We discard all questions deemed unanswerable from the validation and test sets, and further discard all data from any workers with less than half of their questions considered answerable. It should be emphasized that the main purpose of this process is to create a level playing field for comparison across datasets constructed for different model adversaries, and can inevitably result in valid questions being discarded. The total cost for training and qualification, dataset construction, and validation is approximately USD 27,000.

	Dev		*Test*	
Resource	*EM*	F_1	*EM*	F_1
$\mathcal{D}_{\text{BiDAF}}$	63.0	76.9	62.6	78.5
$\mathcal{D}_{\text{BERT}}$	59.2	74.3	63.9	76.9
$\mathcal{D}_{\text{RoBERTa}}$	58.1	72.0	58.7	73.7

Table 2: Non-expert human performance results for a randomly-selected validator per question.

Human Performance We select a randomly chosen validator's answer to each question and compute Exact Match (EM) and word overlap F_1 scores with the original to calculate non-expert human performance; Table 2 shows the result. We observe a clear trend: The stronger the model in the loop used to construct the dataset, the harder the resulting questions become for humans.

3.4 Dataset Statistics

Table 3 provides general details on the number of passages and question-answer pairs used in the different dataset splits. The average number of words in questions and answers, as well as the

Resource	*#Passages*			*#QAs*		
	Train	*Dev*	*Test*	*Train*	*Dev*	*Test*
$\mathcal{D}_{\text{SQuAD}}$	18,891	971	1,096	87,599	5,278	5,292
$\mathcal{D}_{\text{BiDAF}}$	2,523	278	277	10,000	1,000	1,000
$\mathcal{D}_{\text{BERT}}$	2,444	283	292	10,000	1,000	1,000
$\mathcal{D}_{\text{RoBERTa}}$	2,552	341	333	10,000	1,000	1,000

Table 3: Number of passages and question-answer pairs for each data resource.

	$\mathcal{D}_{\text{SQuAD}}$	$\mathcal{D}_{\text{BiDAF}}$	$\mathcal{D}_{\text{BERT}}$	$\mathcal{D}_{\text{RoBERTa}}$
Question length	10.3	9.8	9.8	10.0
Answer length	2.6	2.9	3.0	3.2
N-Gram overlap	3.0	2.2	2.1	2.0

Table 4: Average number of words per question and answer, and average longest n-gram overlap between passage and question.

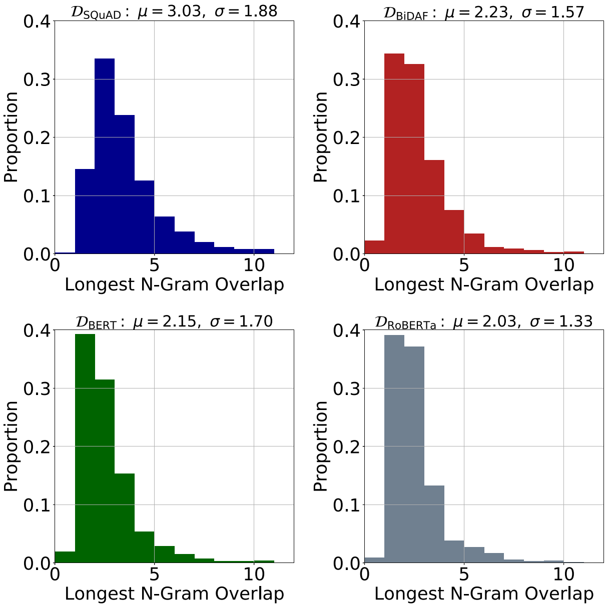


Figure 4: Distribution of longest n-gram overlap between passage and question for different datasets. μ: mean; σ: standard deviation.

Model	Resource	*Original*		*Re-init.*	
		EM	*F$_1$*	*EM*	*F$_1$*
BiDAF	$\mathcal{D}_{\text{BiDAF}}{}^{dev}$	0.0	5.3	10.7$_{0.8}$	20.4$_{1.0}$
BERT	$\mathcal{D}_{\text{BERT}}{}^{dev}$	0.0	4.9	19.7$_{1.0}$	30.1$_{1.2}$
RoBERTa	$\mathcal{D}_{\text{RoBERTa}}{}^{dev}$	0.0	6.1	15.7$_{0.9}$	25.8$_{1.2}$
BiDAF	$\mathcal{D}_{\text{BiDAF}}{}^{test}$	0.0	5.5	11.6$_{1.0}$	21.3$_{1.2}$
BERT	$\mathcal{D}_{\text{BERT}}{}^{test}$	0.0	5.3	18.9$_{1.2}$	29.4$_{1.1}$
RoBERTa	$\mathcal{D}_{\text{RoBERTa}}{}^{test}$	0.0	5.9	16.1$_{0.8}$	26.7$_{0.9}$

Table 5: Consistency of the adversarial effect (or lack thereof) when retraining the models in the loop on the same data again, but with different random seeds. We report the mean and standard deviation (subscript) over 10 re-initialization runs.

average longest n-gram overlap between passage and question are given in Table 4.

We can again observe two clear trends: From weaker towards stronger models used in the annotation loop, the average length of answers increases, and the largest n-gram overlap drops from 3 to 2 tokens. That is, on average there is a trigram overlap between the passage and question for $\mathcal{D}_{\text{SQuAD}}$, but only a bigram overlap for $\mathcal{D}_{\text{RoBERTa}}$ (Figure 4).[3] This is in line with prior observations on lexical overlap as a predictive cue in SQuAD (Weissenborn et al., 2017; Min et al., 2018); questions with less overlap are harder to answer for any of the three models.

We furthermore analyze question types based on the question *wh*-word. We find that—in contrast to $\mathcal{D}_{\text{SQuAD}}$—the datasets collected with a model in the annotation loop have fewer *when*, *how*, and *in* questions, and more *which*, *where*, and *why* questions, as well as questions in the *other* category, which indicates increased question diversity. In terms of answer types, we observe more common noun and verb phrase clauses than in $\mathcal{D}_{\text{SQuAD}}$, as well as fewer dates, names, and numeric answers. This reflects on the strong answer-type matching capabilities of contemporary RC models. The training and validation sets used in this analysis ($\mathcal{D}_{\text{BiDAF}}$, $\mathcal{D}_{\text{BERT}}$, and $\mathcal{D}_{\text{RoBERTa}}$) will be publicly released.

4 Experiments

4.1 Consistency of the Model in the Loop

We begin with an experiment regarding the consistency of the adversarial nature of the models in the annotation loop. Our annotation pipeline is designed to reject all samples where the model correctly predicts the answer. How reproducible is this when retraining the model with the same training data? To measure this, we evaluate the performance of instances of BiDAF, BERT, and RoBERTa, which only differ from the model used during annotation in their random initialization

[3] Note that the original SQuAD1.1 dataset can be considered a limit case of the adversarial annotation framework, in which the model in the loop always predicts the wrong answer, thus every question is accepted.

Model	Trained On	Evaluation (Test) Dataset											
		$\mathcal{D}_{\text{SQuAD}}$		$\mathcal{D}_{\text{BiDAF}}$		$\mathcal{D}_{\text{BERT}}$		D_{RoBERTa}		$\mathcal{D}_{\text{DROP}}$		$\mathcal{D}_{\text{NQ}}$	
		EM	F_1	*EM*	F_1	*EM*	F_1	*EM*	F_1	*EM*	F_1	*EM*	F_1
BiDAF	$\mathcal{D}_{\text{SQuAD(10K)}}$	$\underline{40.9}_{0.6}$	$\underline{54.3}_{0.6}$	$7.1_{0.6}$	$\underline{15.7}_{0.6}$	$5.6_{0.3}$	$13.5_{0.4}$	$5.7_{0.4}$	$13.5_{0.4}$	$3.8_{0.4}$	$8.6_{0.6}$	$\underline{25.1}_{1.1}$	$\underline{38.7}_{0.7}$
	$\mathcal{D}_{\text{BiDAF}}$	$11.5_{0.4}$	$20.9_{0.4}$	$5.3_{0.4}$	$11.6_{0.5}$	$7.1_{0.4}$	$14.8_{0.6}$	$6.8_{0.5}$	$13.5_{0.6}$	$6.5_{0.5}$	$12.4_{0.4}$	$15.7_{1.1}$	$28.7_{0.8}$
	$\mathcal{D}_{\text{BERT}}$	$10.8_{0.3}$	$19.8_{0.4}$	$\underline{7.2}_{0.5}$	$14.4_{0.6}$	$6.9_{0.3}$	$14.5_{0.4}$	$8.1_{0.4}$	$15.0_{0.6}$	$7.8_{0.9}$	$14.5_{0.9}$	$16.5_{0.6}$	$28.3_{0.9}$
	$\mathcal{D}_{\text{RoBERTa}}$	$10.7_{0.2}$	$20.2_{0.3}$	$6.3_{0.7}$	$13.5_{0.8}$	$\underline{9.4}_{0.6}$	$\underline{17.0}_{0.6}$	$\underline{8.9}_{0.9}$	$\underline{16.0}_{0.8}$	$\underline{15.3}_{0.8}$	$\underline{22.9}_{0.8}$	$13.4_{0.9}$	$27.1_{1.2}$
BERT	$\mathcal{D}_{\text{SQuAD(10K)}}$	$\underline{69.4}_{0.5}$	$\underline{82.7}_{0.4}$	$35.1_{1.9}$	$49.3_{2.2}$	$15.6_{2.0}$	$27.3_{2.1}$	$11.9_{1.5}$	$23.0_{1.4}$	$18.9_{2.3}$	$28.9_{3.2}$	$52.9_{1.0}$	$68.2_{1.0}$
	$\mathcal{D}_{\text{BiDAF}}$	$66.5_{0.7}$	$80.6_{0.6}$	$\underline{46.2}_{1.2}$	$\underline{61.1}_{1.2}$	$\underline{37.8}_{1.4}$	$\underline{48.8}_{1.5}$	$\underline{30.6}_{0.8}$	$\underline{42.5}_{0.6}$	$\underline{41.1}_{2.3}$	$\underline{50.6}_{2.0}$	$\underline{54.2}_{1.2}$	$\underline{69.8}_{0.9}$
	$\mathcal{D}_{\text{BERT}}$	$61.2_{1.8}$	$75.7_{1.6}$	$42.9_{1.9}$	$57.5_{1.8}$	$37.4_{2.1}$	$47.9_{2.0}$	$29.3_{2.1}$	$40.0_{2.3}$	$39.4_{2.2}$	$47.6_{2.2}$	$49.9_{2.3}$	$65.7_{2.3}$
	$\mathcal{D}_{\text{RoBERTa}}$	$57.0_{1.7}$	$71.7_{1.8}$	$37.0_{2.3}$	$52.0_{2.5}$	$34.8_{1.5}$	$45.9_{2.0}$	$30.5_{2.2}$	$41.2_{2.2}$	$39.0_{3.1}$	$47.4_{2.8}$	$45.8_{2.4}$	$62.4_{2.5}$
RoBERTa	$\mathcal{D}_{\text{SQuAD(10K)}}$	$\underline{68.6}_{0.5}$	$\underline{82.8}_{0.3}$	$37.7_{1.1}$	$53.8_{1.1}$	$20.8_{1.2}$	$34.0_{1.0}$	$11.0_{0.8}$	$22.1_{0.9}$	$25.0_{2.2}$	$39.4_{2.4}$	$43.9_{3.8}$	$62.8_{3.1}$
	$\mathcal{D}_{\text{BiDAF}}$	$64.8_{0.7}$	$80.0_{0.4}$	$\underline{48.0}_{1.2}$	$\underline{64.3}_{1.1}$	$\underline{40.0}_{1.5}$	$\underline{51.5}_{1.3}$	$29.0_{1.9}$	$39.9_{1.8}$	$\underline{44.5}_{2.1}$	$\underline{55.4}_{1.9}$	$\underline{48.4}_{1.1}$	$\underline{66.9}_{0.8}$
	$\mathcal{D}_{\text{BERT}}$	$59.5_{1.0}$	$75.1_{0.9}$	$45.4_{1.5}$	$60.7_{1.5}$	$38.4_{1.8}$	$49.8_{1.7}$	$28.2_{1.5}$	$38.8_{1.5}$	$42.2_{2.3}$	$52.6_{2.0}$	$45.8_{1.1}$	$63.6_{1.1}$
	$\mathcal{D}_{\text{RoBERTa}}$	$56.2_{0.7}$	$72.1_{0.7}$	$41.4_{0.8}$	$57.1_{0.8}$	$38.4_{1.1}$	$49.5_{0.9}$	$\underline{30.2}_{1.3}$	$\underline{41.0}_{1.2}$	$41.2_{0.9}$	$51.2_{0.8}$	$43.6_{1.1}$	$61.6_{0.9}$

Table 6: Training models on various datasets, each with 10,000 samples, and measuring their generalization to different evaluation datasets. Results underlined indicate the best result per model. We report the mean and standard deviation (subscript) over 10 runs with different random seeds.

and order of mini-batch samples during training. These results are shown in Table 5.

First, we observe—as expected given our annotation constraints—that model performance is 0.0EM on datasets created with the same respective model in the annotation loop. We observe, however, that retrained models do not reliably perform as poorly on those samples. For example, BERT reaches 19.7EM, whereas the original model used during annotation provides no correct answer with 0.0EM. This demonstrates that random model components can substantially affect the adversarial annotation process. The evaluation furthermore serves as a baseline for subsequent model evaluations: This much of the performance range can be learned merely by retraining the same model. A possible takeaway for using the model-in-the-loop annotation strategy in the future is to rely on ensembles of adversaries and reduce the dependency on one particular model instantiation, as investigated by Grefenstette et al. (2018).

4.2 Adversarial Generalization

A potential problem with the focus on challenging questions is that they might be very distinct from one another, leading to difficulties in learning to generalize to and from them. We conduct a series of experiments in which we train on $\mathcal{D}_{\text{BiDAF}}$, $\mathcal{D}_{\text{BERT}}$, and $\mathcal{D}_{\text{RoBERTa}}$, and observe how well models can learn to generalize to the respective test portions of these datasets. Table 6 shows the results, and there is a multitude of observations.

First, one clear trend we observe across all training data setups is a negative performance progression when evaluated against datasets constructed with a stronger model in the loop. This trend holds true for all but the BiDAF model, in each of the training configurations, and for each of the evaluation datasets. For example, RoBERTa trained on $\mathcal{D}_{\text{RoBERTa}}$ achieves 72.1, 57.1, 49.5, and 41.0F$_1$ when evaluated on $\mathcal{D}_{\text{SQuAD}}$, $\mathcal{D}_{\text{BiDAF}}$, $\mathcal{D}_{\text{BERT}}$, and $\mathcal{D}_{\text{RoBERTa}}$ respectively.

Second, we observe that the BiDAF model is not able to generalize well to datasets constructed with a model in the loop, independent of its training setup. In particular, it is unable to learn from $\mathcal{D}_{\text{BiDAF}}$, thus failing to overcome some of its own blind spots through adversarial training. Irrespective of the training dataset, BiDAF consistently performs poorly on the adversarially collected evaluation datasets, and we also note a substantial performance drop when trained on $\mathcal{D}_{\text{BiDAF}}$, $\mathcal{D}_{\text{BERT}}$, or $\mathcal{D}_{\text{RoBERTa}}$ and evaluated on $\mathcal{D}_{\text{SQuAD}}$.

In contrast, BERT and RoBERTa are able to partially overcome their blind spots through training on data collected with a model in the loop, and to a degree that far exceeds what would be expected from random retraining (cf. Table 5).

Model	**Training Dataset**	**Evaluation (Test) Dataset**							
		$\mathcal{D}_{\text{SQuAD}}$		$\mathcal{D}_{\text{BiDAF}}$		$\mathcal{D}_{\text{BERT}}$		$\mathcal{D}_{\text{RoBERTa}}$	
		EM	F_1	*EM*	F_1	*EM*	F_1	*EM*	F_1
BiDAF	$\mathcal{D}_{\text{SQuAD}}$	$\underline{56.7}_{0.5}$	$\underline{70.1}_{0.3}$	$11.6_{1.0}$	$21.3_{1.1}$	$8.6_{0.6}$	$17.3_{0.8}$	$8.3_{0.7}$	$16.8_{0.5}$
	$\mathcal{D}_{\text{SQuAD}} + \mathcal{D}_{\text{BiDAF}}$	$56.3_{0.6}$	$69.7_{0.4}$	$14.4_{0.9}$	$24.4_{0.9}$	$15.6_{1.1}$	$24.7_{1.1}$	$14.3_{0.5}$	$23.3_{0.7}$
	$\mathcal{D}_{\text{SQuAD}} + \mathcal{D}_{\text{BERT}}$	$56.2_{0.6}$	$69.4_{0.6}$	$14.4_{0.7}$	$24.2_{0.8}$	$15.7_{0.6}$	$25.1_{0.6}$	$13.9_{0.8}$	$22.7_{0.8}$
	$\mathcal{D}_{\text{SQuAD}} + \mathcal{D}_{\text{RoBERTa}}$	$56.2_{0.7}$	$69.6_{0.6}$	$\underline{14.7}_{0.9}$	$\underline{24.8}_{0.8}$	$\underline{17.9}_{0.5}$	$\underline{26.7}_{0.6}$	$\underline{16.7}_{1.1}$	$\underline{25.0}_{0.8}$
BERT	$\mathcal{D}_{\text{SQuAD}}$	$74.8_{0.3}$	$86.9_{0.2}$	$46.4_{0.7}$	$60.5_{0.8}$	$24.4_{1.2}$	$35.9_{1.1}$	$17.3_{0.7}$	$28.9_{0.9}$
	$\mathcal{D}_{\text{SQuAD}} + \mathcal{D}_{\text{BiDAF}}$	$75.2_{0.4}$	$\underline{87.2}_{0.2}$	$52.4_{0.9}$	$66.5_{0.9}$	$40.9_{1.3}$	$51.2_{1.5}$	$32.9_{0.9}$	$44.1_{0.8}$
	$\mathcal{D}_{\text{SQuAD}} + \mathcal{D}_{\text{BERT}}$	$75.1_{0.3}$	$87.1_{0.3}$	$\underline{54.1}_{1.0}$	$\underline{68.0}_{0.8}$	$43.7_{1.1}$	$54.1_{1.3}$	$34.7_{0.7}$	$45.7_{0.8}$
	$\mathcal{D}_{\text{SQuAD}} + \mathcal{D}_{\text{RoBERTa}}$	$\underline{75.3}_{0.4}$	$87.1_{0.3}$	$53.0_{1.1}$	$67.1_{0.8}$	$\underline{44.1}_{1.1}$	$\underline{54.4}_{0.9}$	$\underline{36.6}_{0.8}$	$\underline{47.8}_{0.5}$
RoBERTa	$\mathcal{D}_{\text{SQuAD}}$	$73.2_{0.4}$	$86.3_{0.2}$	$48.9_{1.1}$	$64.3_{1.1}$	$31.3_{1.1}$	$43.5_{1.2}$	$16.1_{0.8}$	$26.7_{0.9}$
	$\mathcal{D}_{\text{SQuAD}} + \mathcal{D}_{\text{BiDAF}}$	$\underline{73.9}_{0.4}$	$\underline{86.7}_{0.2}$	$55.0_{1.4}$	$69.7_{0.9}$	$46.5_{1.1}$	$57.3_{1.1}$	$31.9_{0.8}$	$42.4_{1.0}$
	$\mathcal{D}_{\text{SQuAD}} + \mathcal{D}_{\text{BERT}}$	$73.8_{0.2}$	$\underline{86.7}_{0.2}$	$55.4_{1.0}$	$70.1_{0.9}$	$48.9_{1.0}$	$59.0_{1.2}$	$32.9_{1.3}$	$43.7_{1.4}$
	$\mathcal{D}_{\text{SQuAD}} + \mathcal{D}_{\text{RoBERTa}}$	$73.5_{0.3}$	$86.5_{0.2}$	$\underline{55.9}_{0.7}$	$\underline{70.6}_{0.7}$	$\underline{49.1}_{1.2}$	$\underline{59.5}_{1.2}$	$\underline{34.7}_{1.0}$	$\underline{45.9}_{1.2}$

Table 7: Training models on SQuAD, as well as SQuAD combined with different adversarially created datasets. Results underlined indicate the best result per model. We report the mean and standard deviation (subscript) over 10 runs with different random seeds.

For example, BERT reaches 47.9F$_1$ when trained and evaluated on $\mathcal{D}_{\text{BERT}}$, while RoBERTa trained on $\mathcal{D}_{\text{RoBERTa}}$ reaches 41.0F$_1$ on $\mathcal{D}_{\text{RoBERTa}}$, both considerably better than random retraining or when training on the non-adversarially collected $\mathcal{D}_{\text{SQuAD(10K)}}$, showing gains of 20.6F$_1$ for BERT and 18.9F$_1$ for RoBERTa. These observations suggest that there exists learnable structure among harder questions that can be picked up by some of the models, yet not all, as BiDAF fails to achieve this. The fact that even BERT can learn to generalize to $\mathcal{D}_{\text{RoBERTa}}$, but not BiDAF to $\mathcal{D}_{\text{BERT}}$ suggests the existence of an inherent limitation to what BiDAF can learn from these new samples, compared with BERT and RoBERTa.

More generally, we observe that training on $\mathcal{D}_{\text{S}}$, where S is a stronger RC model, helps generalize to $\mathcal{D}_{\text{W}}$, where W is a weaker model—for example, training on $\mathcal{D}_{\text{RoBERTa}}$ and testing on $\mathcal{D}_{\text{BERT}}$. On the other hand, training on $\mathcal{D}_{\text{W}}$ also leads to generalization towards $\mathcal{D}_{\text{S}}$. For example, RoBERTa trained on 10,000 SQuAD samples reaches 22.1F$_1$ on $\mathcal{D}_{\text{RoBERTa}}$ ($\mathcal{D}_{\text{S}}$), whereas training RoBERTa on $\mathcal{D}_{\text{BiDAF}}$ and $\mathcal{D}_{\text{BERT}}$ ($\mathcal{D}_{\text{W}}$) bumps this number to 39.9F$_1$ and 38.8F$_1$, respectively.

Third, we observe similar performance degradation patterns for both BERT and RoBERTa on $\mathcal{D}_{\text{SQuAD}}$ when trained on data collected with increasingly stronger models in the loop. For example, RoBERTa evaluated on $\mathcal{D}_{\text{SQuAD}}$ achieves 82.8, 80.0, 75.1, and 72.1F$_1$ when trained on $\mathcal{D}_{\text{SQuAD(10K)}}$, $\mathcal{D}_{\text{BiDAF}}$, $\mathcal{D}_{\text{BERT}}$, and $\mathcal{D}_{\text{RoBERTa}}$, respectively. This may indicate a gradual shift in the distributions of composed questions as the model in the loop gets stronger.

These observations suggest an encouraging takeaway for the model-in-the-loop annotation paradigm: Even though a particular model might be chosen as an adversary in the annotation loop, which at some point falls behind more recent state-of-the-art models, these future models can still benefit from data collected with the weaker model, and also generalize better to samples composed with the stronger model in the loop.

We further show experimental results for the same models and training datasets, but now including SQuAD as additional training data, in Table 7. In this training setup we generally see improved generalization to $\mathcal{D}_{\text{BiDAF}}$, $\mathcal{D}_{\text{BERT}}$, and $\mathcal{D}_{\text{RoBERTa}}$. Interestingly, the relative differences between $\mathcal{D}_{\text{BiDAF}}$, $\mathcal{D}_{\text{BERT}}$, and $\mathcal{D}_{\text{RoBERTa}}$ as training sets used in conjunction with SQuAD are much diminished, and especially $\mathcal{D}_{\text{RoBERTa}}$ as (part of) the training set now generalizes substantially better. We see that BERT and RoBERTa both show consistent performance gains with the

Model	**Evaluation (Test) Dataset**							
	$\mathcal{D}_{\text{SQuAD}}$		$\mathcal{D}_{\text{BiDAF}}$		$\mathcal{D}_{\text{BERT}}$		$\mathcal{D}_{\text{RoBERTa}}$	
	EM	F_1	*EM*	F_1	*EM*	F_1	*EM*	F_1
BiDAF	$57.1_{0.4}$	$70.4_{0.3}$	$17.1_{0.8}$	$27.0_{0.9}$	$20.0_{1.0}$	$29.2_{0.8}$	$18.3_{0.6}$	$27.4_{0.7}$
BERT	$\underline{75.5}_{0.2}$	$\underline{87.2}_{0.2}$	$57.7_{1.0}$	$71.0_{1.1}$	$52.1_{0.7}$	$62.2_{0.7}$	$\underline{43.0}_{1.1}$	$\underline{54.2}_{1.0}$
RoBERTa	$74.2_{0.3}$	$86.9_{0.3}$	$\underline{59.8}_{0.5}$	$\underline{74.1}_{0.6}$	$\underline{55.1}_{0.6}$	$\underline{65.1}_{0.7}$	$41.6_{1.0}$	$52.7_{1.0}$

Table 8: Training models on SQuAD combined with all the adversarially created datasets $\mathcal{D}_{\text{BiDAF}}$, $\mathcal{D}_{\text{BERT}}$, and $\mathcal{D}_{\text{RoBERTa}}$. Results underlined indicate the best result per model. We report the mean and standard deviation (subscript) over 10 runs with different random seeds.

addition of the original SQuAD1.1 training data, but unlike in Table 6, this comes without any noticeable decline in performance on $\mathcal{D}_{\text{SQuAD}}$, suggesting that the adversarially constructed datasets expose inherent model weaknesses, as investigated by Liu et al. (2019a).

Furthermore, RoBERTa achieves the strongest results on the adversarially collected evaluation sets, in particular when trained on $\mathcal{D}_{\text{SQuAD}} + \mathcal{D}_{\text{RoBERTa}}$. This stands in contrast to the results in Table 6, where training on $\mathcal{D}_{\text{BiDAF}}$ in several cases led to better generalization than training on $\mathcal{D}_{\text{RoBERTa}}$. A possible explanation is that training on $\mathcal{D}_{\text{RoBERTa}}$ leads to a larger degree of overfitting to specific adversarial examples in $\mathcal{D}_{\text{RoBERTa}}$ than training on $\mathcal{D}_{\text{BiDAF}}$, and that the inclusion of a large number of standard SQuAD training samples can mitigate this effect.

Results for the models trained on all the datasets combined ($\mathcal{D}_{\text{SQuAD}}$, $\mathcal{D}_{\text{BiDAF}}$, $\mathcal{D}_{\text{BERT}}$, and $\mathcal{D}_{\text{RoBERTa}}$) are shown in Table 8. These further support the previous observations and provide additional performance gains where, for example, RoBERTa achieves F_1 scores of 86.9 on $\mathcal{D}_{\text{SQuAD}}$, 74.1 on $\mathcal{D}_{\text{BiDAF}}$, 65.1 on $\mathcal{D}_{\text{BERT}}$, and 52.7 on $\mathcal{D}_{\text{RoBERTa}}$, surpassing the best previous performance on all adversarial datasets.

Finally, we identify a risk of datasets constructed with weaker models in the loop becoming outdated. For example, RoBERTa achieves 58.2EM/73.2F_1 on $\mathcal{D}_{\text{BiDAF}}$, in contrast to 0.0EM/5.5F_1 for BiDAF—which is not far from the non-expert human performance of 62.6EM/78.5F_1 (cf. Table 2).

It is also interesting to note that, even when training on all the combined data (cf. Table 8), BERT outperforms RoBERTa on $\mathcal{D}_{\text{RoBERTa}}$ and vice versa, suggesting that there may exist weaknesses inherent to each model class.

4.3 Generalization to Non-Adversarial Data

Compared with standard annotation, the model-in-the-loop approach generally results in new question distributions. Consequently, models trained on adversarially composed questions might not be able to generalize to standard (‘‘easy’’) questions, thus limiting the practical usefulness of the resulting data. To what extent do models trained on model-in-the-loop questions generalize differently to standard (‘‘easy’’) questions, compared with models trained on standard (‘‘easy’’) questions?

To measure this we further train each of our three models on either $\mathcal{D}_{\text{BiDAF}}$, $\mathcal{D}_{\text{BERT}}$, or $\mathcal{D}_{\text{RoBERTa}}$ and test on $\mathcal{D}_{\text{SQuAD}}$, with results in the $\mathcal{D}_{\text{SQuAD}}$ columns of Table 6. For comparison, the models are also trained on 10,000 SQuAD1.1 samples (referred to as $\mathcal{D}_{\text{SQuAD(10K)}}$) chosen from the same passages as the adversarial datasets, thus eliminating size and paragraph choice as potential confounding factors. The models are tuned for EM on the held-out $\mathcal{D}_{\text{SQuAD}}$ validation set. Note that, although performance values on the majority vote $\mathcal{D}_{\text{SQuAD}}$ dataset are lower than on the original, for the reasons described earlier, this enables direct comparisons across all datasets.

Remarkably, neither BERT nor RoBERTa show substantial drops when trained on $\mathcal{D}_{\text{BiDAF}}$ compared to training on SQuAD data ($-2.1F_1$, and $-2.8F_1$): Training these models on a dataset with a weaker model in the loop still leads to strong generalization even to data from the original SQuAD distribution, which all models in the loop are trained on. BiDAF, on the other hand, fails to learn such information from the adversarially collected data, and drops >30F_1 for each of the new training sets, compared to training on SQuAD.

We also observe a gradual decrease in generalization to SQuAD when training on $\mathcal{D}_{\text{BiDAF}}$ towards training on $\mathcal{D}_{\text{RoBERTa}}$. This suggests that the stronger the model, the more dissimilar the resulting data distribution becomes from the original SQuAD distribution. We later find further support for this explanation in a qualitative analysis (Section 5). It may, however, also be due to a limitation of BERT and RoBERTa—similar to BiDAF—in learning from a data distribution designed to beat these models; an even stronger model might learn more from, for example, $\mathcal{D}_{\text{RoBERTa}}$.

4.4 Generalization to DROP and NaturalQuestions

Finally, we investigate to what extent models can transfer skills learned on the datasets created with a model in the loop to two recently introduced datasets: DROP (Dua et al., 2019), and NaturalQuestions (Kwiatkowski et al., 2019). In this experiment we select the subsets of DROP and NaturalQuestions that align with the structural constraints of SQuAD to ensure a like-for-like analysis. Specifically, we only consider questions in DROP where the answer is a span in the passage and where there is only one candidate answer. For NaturalQuestions, we consider all non-tabular long answers as passages, remove HTML tags and use the short answer as the extracted span. We apply this filtering on the validation sets for both datasets. Next we split them, stratifying by document (as we did for $\mathcal{D}_{\text{SQuAD}}$), which results in $1409/1418$ validation and test set examples for DROP, and $964/982$ for NaturalQuestions, respectively. We denote these datasets as $\mathcal{D}_{\text{DROP}}$ and $\mathcal{D}_{\text{NQ}}$ for clarity and distinction from their unfiltered versions. We consider the same models and training datasets as before, but tune on the respective validation sets of $\mathcal{D}_{\text{DROP}}$ and $\mathcal{D}_{\text{NQ}}$. Table 6 shows the results of these experiments in the respective $\mathcal{D}_{\text{DROP}}$ and $\mathcal{D}_{\text{NQ}}$ columns.

First, we observe clear generalization improvements towards $\mathcal{D}_{\text{DROP}}$ across all models compared to training on $\mathcal{D}_{\text{SQuAD(10K)}}$ when training on any of $\mathcal{D}_{\text{BiDAF}}$, $\mathcal{D}_{\text{BERT}}$, or $\mathcal{D}_{\text{RoBERTa}}$. That is, including a model in the loop for the training dataset leads to improved transfer towards $\mathcal{D}_{\text{DROP}}$. Note that DROP also makes use of a BiDAF model in the loop during annotation; these results are in line with our prior observations when testing the same setups on $\mathcal{D}_{\text{BiDAF}}$, $\mathcal{D}_{\text{BERT}}$, and $\mathcal{D}_{\text{RoBERTa}}$, compared to training on $\mathcal{D}_{\text{SQuAD(10K)}}$.

Second, we observe overall strong transfer results towards $\mathcal{D}_{\text{NQ}}$, with up to 69.8F_1 for a BERT model trained on $\mathcal{D}_{\text{BiDAF}}$. Note that this result is similar to, and even slightly improves over, model training with SQuAD data of the same size. That is, relative to training on SQuAD data, training on adversarially collected data $\mathcal{D}_{\text{BiDAF}}$ does not impede generalization to the $\mathcal{D}_{\text{NQ}}$ dataset, which was created without a model in the annotation loop. We then, however, see a similar negative performance progression as observed before when testing on $\mathcal{D}_{\text{SQuAD}}$: The stronger the model in the annotation loop of the training dataset, the lower the test accuracy on test data from a data distribution composed without a model in the loop.

5 Qualitative Analysis

Having applied the general model-in-the-loop methodology on models of varying strength, we next perform a qualitative comparison of the nature of the resulting questions. As reference points we also include the original SQuAD questions, as well as DROP and NaturalQuestions, in this comparison: these datasets are both constructed to overcome limitations in SQuAD and have subsets sufficiently similar to SQuAD to make an analysis possible. Specifically, we seek to understand the qualitative differences in terms of reading comprehension challenges posed by the questions in each of these datasets.

5.1 Comprehension Requirements

There exists a variety of prior work that seeks to understand the types of knowledge, comprehension skills, or types of reasoning required to answer questions based on text (Rajpurkar et al., 2016; Clark et al., 2018; Sugawara et al., 2019; Dua et al., 2019; Dasigi et al., 2019); we are, however, unaware of any commonly accepted formalism. We take inspiration from these but develop our own taxonomy of comprehension requirements which suits the datasets analyzed. Our taxonomy contains 13 labels, most of which are commonly used in other work. However, the following three deserve additional clarification: i) *explicit*–for which the answer is stated nearly

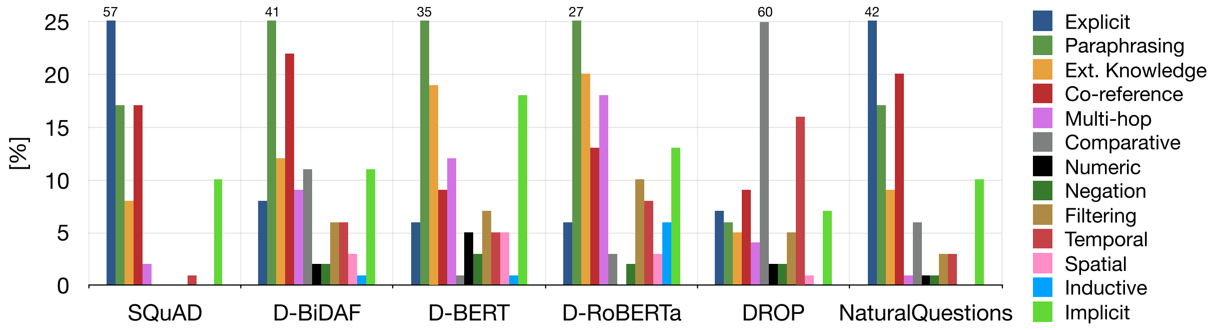


Figure 5: Comparison of comprehension types for the questions in different datasets. The label types are neither mutually exclusive nor comprehensive. Values above columns indicate excess of the axis range.

word-for-word in the passage as it is in the question, ii) *filtering*–a set of answers is narrowed down to select one by some particular distinguishing feature, and iii) *implicit*–the answer builds on information implied by the passage and does not otherwise require any of the other types of reasoning.

We annotate questions with labels from this catalogue in a manner that is not mutually exclusive, and neither fully comprehensive; the development of such a catalogue is itself very challenging. Instead, we focus on capturing the most salient characteristics of each given question, and assign it up to three of the labels in our catalogue. In total, we analyze 100 samples from the validation set of each of the datasets; Figure 5 shows the results.

5.2 Observations

An initial observation is that the majority (57%) of answers to SQuAD questions are stated explicitly, without comprehension requirements beyond the literal level. This number decreases substantially for any of the model-in-the-loop datasets derived from SQuAD (e.g., 8% for $\mathcal{D}_{\text{BiDAF}}$) and also $\mathcal{D}_{\text{DROP}}$, yet 42% of questions in $\mathcal{D}_{\text{NQ}}$ share this property. In contrast to SQuAD, the model-in-the-loop questions generally tend to involve more paraphrasing. They also require more external knowledge, and multi-hop inference (beyond co-reference resolution) with an increasing trend for stronger models used in the annotation loop. Model-in-the-loop questions further fan out into a variety of small, but non-negligible proportions of more specific types of inference required for comprehension, for example, spatial or temporal inference (both going beyond explicitly stated spatial or temporal information)—SQuAD questions rarely require these at all. Some of these more particular inference types are common features of the other two datasets, in particular *comparative* questions for DROP (60%) and to a small extent also NaturalQuestions. Interestingly, $\mathcal{D}_{\text{BiDAF}}$ possesses the largest number of comparison questions (11%) among our model-in-the-loop datasets, whereas $\mathcal{D}_{\text{BERT}}$ and $\mathcal{D}_{\text{RoBERTa}}$ only possess 1% and 3%, respectively. This offers an explanation for our previous observation in Table 6, where BERT and RoBERTa perform better on $\mathcal{D}_{\text{DROP}}$ when trained on $\mathcal{D}_{\text{BiDAF}}$ rather than on $\mathcal{D}_{\text{BERT}}$ or $\mathcal{D}_{\text{RoBERTa}}$. It is likely that BiDAF as a model in the loop is worse than BERT and RoBERTa at *comparative* questions, as evidenced by the results in Table 6 with BiDAF reaching 8.6F_1, BERT reaching 28.9F_1, and RoBERTa reaching 39.4F_1 on $\mathcal{D}_{\text{DROP}}$ (when trained on $\mathcal{D}_{\text{SQuAD(10K)}}$).

The distribution of NaturalQuestions contains elements of both the SQuAD and $\mathcal{D}_{\text{BiDAF}}$ distributions, which offers a potential explanation for the strong performance on $\mathcal{D}_{\text{NQ}}$ of models trained on $\mathcal{D}_{\text{SQuAD(10K)}}$ and $\mathcal{D}_{\text{BiDAF}}$. Finally, the gradually shifting distribution away from both SQuAD and NaturalQuestions as the model-in-the-loop strength increases reflects our prior observations on the decreasing performance on SQuAD and NaturalQuestions of models trained on datasets with progressively stronger models in the loop.

6 Discussion and Conclusions

We have investigated an RC annotation paradigm that requires a model in the loop to be ‘‘beaten’’ by an annotator. Applying this approach with progressively stronger models in the loop

(BiDAF, BERT, and RoBERTa), we produced three separate datasets. Using these datasets, we investigated several questions regarding the annotation paradigm, in particular, whether such datasets grow outdated as stronger models emerge, and their generalization to standard (non-adversarially collected) questions. We found that stronger models can still learn from data collected with a weak adversary in the loop, and their generalization improves even on datasets collected with a stronger adversary. Models trained on data collected with a model in the loop further generalize well to non-adversarially collected data, both on SQuAD and on NaturalQuestions, yet we observe a gradual deterioration in performance with progressively stronger adversaries.

We see our work as a contribution towards the emerging paradigm of model-in-the-loop annotation. Although this paper has focused on RC, with SQuAD as the original dataset used to train model adversaries, we see no reason in principle why findings would not be similar for other tasks using the same annotation paradigm, when crowdsourcing challenging samples with a model in the loop. We would expect the insights and benefits conveyed by model-in-the-loop annotation to be the greatest on mature datasets where models exceed human performance: Here the resulting data provides a magnifying glass on model performance, focused in particular on samples which models struggle on. On the other hand, applying the method to datasets where performance has not yet plateaued would likely result in a more similar distribution to the original data, which is challenging to models a priori. We hope that the series of experiments on replicability, observations on transfer between datasets collected using models of different strength, as well as our findings regarding generalization to non-adversarially collected data, can support and inform future research and annotation efforts using this paradigm.

Acknowledgments

The authors would like to thank Christopher Potts for his detailed and constructive feedback, and our reviewers. This work was supported by the European Union's Horizon 2020 Research and Innovation Programme under grant agreement no. 875160 and the UK Defence Science and Technology Laboratory (Dstl) and Engineering and Physical Research Council (EPSRC) under grant EP/R018693/1 as a part of the collaboration between US DOD, UK MOD, and UK EPSRC under the Multidisciplinary University Research Initiative (MURI).

References

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*, pages 632–642, Lisbon, Portugal. Association for Computational Linguistics. **DOI:** https://doi.org/10.18653/v1/D15-1075

Danqi Chen, Jason Bolton, and Christopher D. Manning. 2016. A thorough examination of the CNN/Daily Mail reading comprehension task. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 2358–2367. Berlin, Germany. Association for Computational Linguistics. **DOI:** https://doi.org/10.18653/v1/P16-1223, **PMID:** 30036459

Michael Chen, Mike D'Arcy, Alisa Liu, Jared Fernandez, and Doug Downey. 2019. CODAH: An adversarially-authored question answering dataset for common sense. In *Proceedings of the 3rd Workshop on Evaluating Vector Space Representations for NLP*, pages 63–69, Minneapolis, USA. Association for Computational Linguistics.

Eunsol Choi, He He, Mohit Iyyer, Mark Yatskar, Wen-tau Yih, Yejin Choi, Percy Liang, and Luke Zettlemoyer. 2018. QuAC: Question answering in context. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 2174–2184, Brussels, Belgium. Association for Computational Linguistics. **DOI:** https://doi.org/10.18653/v1/D18-1241, **PMID:** 30142985

Peter Clark, Isaac Cowhey, Oren Etzioni, Tushar Khot, Ashish Sabharwal, Carissa Schoenick, and Oyvind Tafjord. 2018. Think you have solved question answering? Try ARC, the AI2 reasoning challenge. *CoRR*, abs/1803.05457.

Pradeep Dasigi, Nelson F. Liu, Ana Marasović, Noah A. Smith, and Matt Gardner. 2019. Quoref: A reading comprehension dataset with questions requiring coreferential reasoning. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 5925–5932, Hong Kong, China. Association for Computational Linguistics. **DOI:** `https://doi.org/10.18653/v1/D19-1606`

Jia Deng, R. Socher, Li Fei-Fei, Wei Dong, Kai Li, and Li-Jia Li. 2009. ImageNet: A large-scale hierarchical image database. In *2009 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 248–255. **DOI:** `https://doi.org/10.1109/CVPR.2009.5206848`

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. BERT: Pre-training of deep bidirectional transformers for language understanding. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 4171–4186, Minneapolis, Minnesota. Association for Computational Linguistics.

Emily Dinan, Samuel Humeau, Bharath Chintagunta, and Jason Weston. 2019. Build it break it fix it for dialogue safety: Robustness from adversarial human attack. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 4537–4546, Hong Kong, China. Association for Computational Linguistics. **DOI:** `https://doi.org/10.18653/v1/D19-1461`

Dheeru Dua, Yizhong Wang, Pradeep Dasigi, Gabriel Stanovsky, Sameer Singh, and Matt Gardner. 2019. DROP: A reading comprehension benchmark requiring discrete reasoning over paragraphs. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 2368–2378, Minneapolis, Minnesota. Association for Computational Linguistics.

Allyson Ettinger, Sudha Rao, Hal Daumé III, and Emily M. Bender. 2017. Towards linguistically generalizable NLP systems: A workshop and shared task. *CoRR*, abs/1711.01505.

Matt Gardner, Joel Grus, Mark Neumann, Oyvind Tafjord, Pradeep Dasigi, Nelson F. Liu, Matthew Peters, Michael Schmitz, and Luke Zettlemoyer. 2018. AllenNLP: A deep semantic natural language processing platform. In *Proceedings of Workshop for NLP Open Source Software (NLP-OSS)*, pages 1–6, Melbourne, Australia. Association for Computational Linguistics. **DOI:** `https://doi.org/10.18653/v1/W18-2501`, **PMCID:** PMC5753512

Edward Grefenstette, Robert Stanforth, Brendan O'Donoghue, Jonathan Uesato, Grzegorz Swirszcz, and Pushmeet Kohli. 2018. Strength in numbers: Trading-off robustness and computation via adversarially-trained ensembles. *CoRR*, abs/1811.09300.

Suchin Gururangan, Swabha Swayamdipta, Omer Levy, Roy Schwartz, Samuel Bowman, and Noah A. Smith. 2018. Annotation artifacts in natural language inference data. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 2 (Short Papers)*, pages 107–112, New Orleans, Louisiana. Association for Computational Linguistics. **DOI:** `https://doi.org/10.18653/v1/N18-2017`

Karl Moritz Hermann, Tomas Kocisky, Edward Grefenstette, Lasse Espeholt, Will Kay, Mustafa Suleyman, and Phil Blunsom. 2015. Teaching machines to read and comprehend. In C. Cortes, N. D. Lawrence, D. D. Lee, M. Sugiyama, and R. Garnett, editors, *Advances in Neural Information Processing Systems 28*, pages 1693–1701. Curran Associates, Inc.

Robin Jia and Percy Liang. 2017. Adversarial examples for evaluating reading comprehension systems. In *Proceedings of the 2017 Conference* *on Empirical Methods in Natura Language Processing*, pages 2021–2031, Copenhagen, Denmark. Association for Computational Linguistics. **DOI:** https://doi.org/10.18653/v1/D17-1215

Mandar Joshi, Eunsol Choi, Daniel Weld, and Luke Zettlemoyer. 2017. TriviaQA: A large scale distantly supervised challenge dataset for reading comprehension. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1601–1611, Vancouver, Canada. Association for Computational Linguistics. **DOI:** https://doi.org/10.18653/v1/P17-1147

Divyansh Kaushik, Eduard Hovy, and Zachary Lipton. 2020. Learning the difference that makes a difference with counterfactually-augmented data. In *International Conference on Learning Representations*.

Divyansh Kaushik and Zachary C. Lipton. 2018. How much reading does reading comprehension require? A critical investigation of popular benchmarks. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 5010–5015, Brussels, Belgium. Association for Computational Linguistics. **DOI:** https://doi.org/10.18653/v1/D18-1546

Tomáš Kočiský, Jonathan Schwarz, Phil Blunsom, Chris Dyer, Karl Moritz Hermann, Gábor Melis, and Edward Grefenstette. 2018. The NarrativeQA reading comprehension challenge. *Transactions of the Association for Computational Linguistics*, 6:317–328. **DOI:** https://doi.org/10.1162/tacl_a_00023

Tom Kwiatkowski, Jennimaria Palomaki, Olivia Redfield, Michael Collins, Ankur Parikh, Chris Alberti, Danielle Epstein, Illia Polosukhin, Jacob Devlin, Kenton Lee, Kristina Toutanova, Llion Jones, Matthew Kelcey, Ming-Wei Chang, Andrew M. Dai, Jakob Uszkoreit, Quoc Le, and Slav Petrov. 2019. Natural Questions: A benchmark for question answering research. *Transactions of the Association for Computational Linguistics*, 7:453–466. **DOI:** https://doi.org/10.1162/tacl_a_00276

David D. Lewis and William A. Gale. 1994. A sequential algorithm for training text classifiers. In *SIGIR*, pages 3–12. ACM/Springer.

Nelson F. Liu, Roy Schwartz, and Noah A. Smith. 2019a. Inoculation by fine-tuning: A method for analyzing challenge datasets. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 2171–2179, Minneapolis, Minnesota. Association for Computational Linguistics.

Yinhan Liu, Myle Ott, Naman Goyal, Jingfei Du, Mandar Joshi, Danqi Chen, Omer Levy, Mike Lewis, Luke Zettlemoyer, and Veselin Stoyanov. 2019b. RoBERTa: A robustly optimized BERT pretraining approach. *CoRR*, abs/1907.11692.

Mitchell P. Marcus, Beatrice Santorini, and Mary Ann Marcinkiewicz. 1993. Building a large annotated corpus of English: The Penn Treebank. *Computational Linguistics*, 19(2): 313–330. **DOI:** https://doi.org/10.21236/ADA273556

Sewon Min, Victor Zhong, Richard Socher, and Caiming Xiong. 2018. Efficient and robust question answering from minimal context over documents. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1725–1735, Melbourne, Australia. Association for Computational Linguistics. **DOI:** https://doi.org/10.18653/v1/P18-1160

Mike Mintz, Steven Bills, Rion Snow, and Daniel Jurafsky. 2009. Distant supervision for relation extraction without labeled data. In *Proceedings of the Joint Conference of the 47th Annual Meeting of the ACL and the 4th International Joint Conference on Natural Language Processing of the AFNLP*, pages 1003–1011, Suntec, Singapore. Association for Computational Linguistics. **DOI:** https://doi.org/10.3115/1690219.1690287

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. *arXiv preprint arXiv:1611.09268*.

Yixin Nie, Adina Williams, Emily Dinan, Mohit Bansal, Jason Weston, and Douwe Kiela. 2019. Adversarial NLI: A new benchmark for natural language understanding. *arXiv preprint arXiv: 1910.14599*. **DOI:** https://doi.org/10.18653/v1/2020.acl-main.441

Pranav Rajpurkar, Robin Jia, and Percy Liang. 2018. Know what you don't know: Unanswerable questions for SQuAD. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, pages 784–789, Melbourne, Australia. Association for Computational Linguistics. **DOI:** https://doi.org/10.18653/v1/P18-2124

Pranav Rajpurkar, Jian Zhang, Konstantin Lopyrev, and Percy Liang. 2016. SQuAD: 100,000+ questions for machine comprehension of text. In *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing*, pages 2383–2392, Austin, Texas. Association for Computational Linguistics. **DOI:** https://doi.org/10.18653/v1/D16-1264

Siva Reddy, Danqi Chen, and Christopher D. Manning. 2019. CoQA: A conversational question answering challenge. *Transactions of the Association for Computational Linguistics*, 7:249–266. **DOI:** https://doi.org/10.1162/tacl_a_00266

Matthew Richardson, Christopher J. C. Burges, and Erin Renshaw. 2013. MCTest: A challenge dataset for the open-domain machine comprehension of text. In *Proceedings of the 2013 Conference on Empirical Methods in Natural Language Processing*, pages 193–203, Seattle, Washington, USA. Association for Computational Linguistics.

Marzieh Saeidi, Max Bartolo, Patrick Lewis, Sameer Singh, Tim Rocktäschel, Mike Sheldon, Guillaume Bouchard, and Sebastian Riedel. 2018. Interpretation of natural language rules in conversational machine reading. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 2087–2097, Brussels, Belgium. Association for Computational Linguistics. **DOI:** https://doi.org/10.18653/v1/D18-1233

Roy Schwartz, Maarten Sap, Ioannis Konstas, Leila Zilles, Yejin Choi, and Noah A. Smith. 2017. The effect of different writing tasks on linguistic style: A case study of the ROC story cloze task. In *Proceedings of the 21st Conference on Computational Natural Language Learning (CoNLL 2017)*, pages 15–25, Vancouver, Canada. Association for Computational Linguistics. **DOI:** https://doi.org/10.18653/v1/K17-1004

Minjoon Seo, Aniruddha Kembhavi, Ali Farhadi, and Hannaneh Hajishirzi. 2017. Bidirectional attention flow for machine comprehension. In *The International Conference on Learning Representations (ICLR)*.

Rion Snow, Brendan O'Connor, Daniel Jurafsky, and Andrew Ng. 2008. Cheap and fast – but is it good? Evaluating non-expert annotations for natural language tasks. In *Proceedings of the 2008 Conference on Empirical Methods in Natural Language Processing*, pages 254–263, Honolulu, Hawaii. Association for Computational Linguistics. **DOI:** https://doi.org/10.3115/1613715.1613751

Saku Sugawara, Kentaro Inui, Satoshi Sekine, and Akiko Aizawa. 2018. What makes reading comprehension questions easier? In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 4208–4219, Brussels, Belgium. Association for Computational Linguistics. **DOI:** https://doi.org/10.18653/v1/D18-1453

Saku Sugawara, Pontus Stenetorp, Kentaro Inui, and Akiko Aizawa. 2019. Assessing the benchmarking capacity of machine reading comprehension datasets. *CoRR*, abs/1911.09241.

James Thorne, Andreas Vlachos, Oana Cocarascu, Christos Christodoulopoulos, and Arpit Mittal. 2019. The FEVER2.0 shared task. In *Proceedings of the Second Workshop on Fact Extraction and VERification (FEVER)*, pages 1–6, Hong Kong, China. Association for Computational Linguistics. **DOI:** https://doi.org/10.18653/v1/D19-6601, **PMCID:** PMC6533707

Adam Trischler, Tong Wang, Xingdi Yuan, Justin Harris, Alessandro Sordoni, Philip Bachman, and Kaheer Suleman. 2017. NewsQA: A machine comprehension dataset. In *Proceedings of the 2nd Workshop on Representation Learning for NLP*, pages 191–200, Vancouver, Canada. Association for Computational Linguistics. **DOI:** `https://doi.org/10.18653/v1/W17-2623`

Eric Wallace, Pedro Rodriguez, Shi Feng, Ikuya Yamada, and Jordan Boyd-Graber. 2019. Trick me if you can: Human-in-the-loop generation of adversarial examples for question answering. *Transactions of the Association for Computational Linguistics*, 7:387–401. **DOI:** `https://doi.org/10.1162/tacl_a_00279`

Dirk Weissenborn, Georg Wiese, and Laura Seiffe. 2017. Making neural QA as simple as possible but not simpler. In *Proceedings of the 21st Conference on Computational Natural Language Learning (CoNLL 2017)*, pages 271–280, Vancouver, Canada. Association for Computational Linguistics. **DOI:** `https://doi.org/10.18653/v1/K17-1028`

Johannes Welbl, Pontus Stenetorp, and Sebastian Riedel. 2018. Constructing datasets for multi-hop reading comprehension across documents. *Transactions of the Association for Computational Linguistics*, 6:287–302. **DOI:** `https://doi.org/10.1162/tacl_a_00021`

Thomas Wolf, Lysandre Debut, Victor Sanh, Julien Chaumond, Clement Delangue, Anthony Moi, Pierric Cistac, Tim Rault, Rémi Louf, Morgan Funtowicz, and Jamie Brew. 2019. HuggingFace's Transformers: State-of-the-art Natural Language Processing.

Zhilin Yang, Peng Qi, Saizheng Zhang, Yoshua Bengio, William Cohen, Ruslan Salakhutdinov, and Christopher D. Manning. 2018a. HotpotQA: A dataset for diverse, explainable multi-hop question answering. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 2369–2380, Brussels, Belgium. Association for Computational Linguistics. **DOI:** `https://doi.org/10.18653/v1/D18-1259`, **PMCID:** PMC6156886

Zhilin Yang, Saizheng Zhang, Jack Urbanek, Will Feng, Alexander Miller, Arthur Szlam, Douwe Kiela, and Jason Weston. 2018b. Mastering the dungeon: Grounded language learning by mechanical turker descent. In *International Conference on Learning Representations*.

Rowan Zellers, Yonatan Bisk, Roy Schwartz, and Yejin Choi. 2018. SWAG: A large-scale adversarial dataset for grounded commonsense inference. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 93–104, Brussels, Belgium. Association for Computational Linguistics. **DOI:** `https://doi.org/10.18653/v1/D18-1009`

Rowan Zellers, Ari Holtzman, Yonatan Bisk, Ali Farhadi, and Yejin Choi. 2019. HellaSwag: Can a machine really finish your sentence? In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 4791–4800, Florence, Italy. Association for Computational Linguistics. **DOI:** `https://doi.org/10.18653/v1/P19-1472`

Sheng Zhang, Xiaodong Liu, Jingjing Liu, Jianfeng Gao, Kevin Duh, and Benjamin Van Durme. 2018. ReCoRD: Bridging the gap between human and machine commonsense reading comprehension. *arXiv preprint arXiv:1810.12885*.