Do Question Answering Modeling Improvements Hold Across Benchmarks?

Nelson F. Liu[♠] **Tony Lee**[♠] **Robin Jia**[♡] **Percy Liang**[♠]
[♠]Computer Science Department, Stanford University, Stanford, CA
[♡]Department of Computer Science, University of Southern California, Los Angeles, CA
{nfliu, tonyhlee, pliang}@cs.stanford.edu
robinjia@usc.edu

Abstract

Do question answering (QA) modeling improvements (e.g., choice of architecture and training procedure) hold consistently across the diverse landscape of QA benchmarks? To study this question, we introduce the notion of *concurrence*—two benchmarks have high concurrence on a set of modeling approaches if they rank the modeling approaches similarly. We measure the concurrence between 32 QA benchmarks on a set of 20 diverse modeling approaches and find that human-constructed benchmarks have high concurrence amongst themselves, even if their passage and question distributions are very different. Surprisingly, even downsampled human-constructed benchmarks (i.e., collecting less data) and programmatically-generated benchmarks (e.g., cloze-formatted examples) have high concurrence with human-constructed benchmarks. These results indicate that, despite years of intense community focus on a small number of benchmarks, the modeling improvements studied hold broadly.

1 Introduction

The NLP community has created a diverse landscape of extractive question answering (QA) benchmarks—their context passages may come from different sources, their questions may focus on different phenomena or be written by different populations, or other aspects of the data collection process may differ. Driven to improve benchmark performance, researchers have proposed a variety of QA modeling approaches. However, not all benchmarks receive equal attention from the community (Koch et al., 2021); many QA modeling approaches are developed on a small handful of benchmarks, especially those with popular leaderboards (e.g., SQuAD; Rajpurkar et al., 2016). As a result, it is conceivable that some modeling improvements may not hold because they are (perhaps inadvertently) benchmark-specific, while others (e.g., pre-training on more data) hold more broadly.

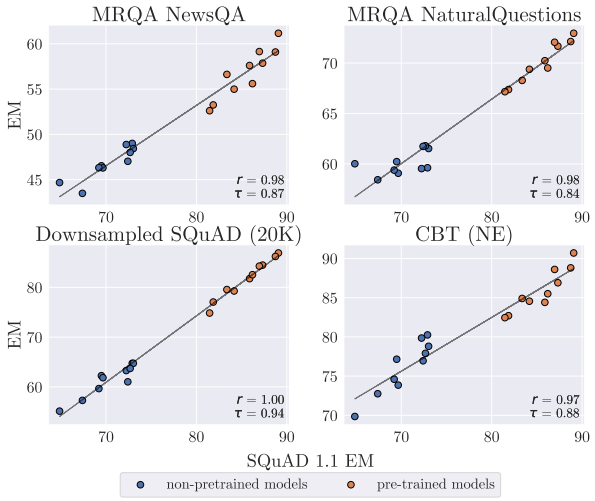

Figure 1: Two benchmarks have high concurrence if they rank a set of modeling approaches similarly. Surprisingly, we find that human-constructed benchmarks (e.g., SQuAD, NaturalQuestions) have high concurrence with other human-constructed benchmarks, downsampled human-constructed benchmarks, and even programmatically-generated cloze benchmarks (e.g., the Children's Book Test; CBT). In addition, we are able to construct synthetic benchmarks that have high concurrence with human-constructed benchmarks despite lacking natural language passages or questions.

In this work, we evaluate whether improvements from modeling *approaches* hold (e.g., choices in model architecture or training procedure)—if a particular modeling approach improves performance when trained and evaluated on one benchmark, does it also improve performance on others? Although much existing work studies whether *systems* generalize (i.e., a model with a particular set of parameters; Jia and Liang, 2017; Talmor and Berant, 2019; Miller et al., 2020), research value often comes not from the systems themselves (e.g., model weights), but from the underlying ideas, techniques, and approaches. We study the comparatively under-investigated question of whether

Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics
Volume 1: Long Papers, pages 13186–13218
July 9-14, 2023

such modeling *approaches* generalize.

To study whether modeling improvements hold across benchmarks, we introduce the notion of *concurrence*. We say that two benchmarks have high concurrence on a set of modeling approaches if they rank the modeling approaches similarly. To assess whether modeling improvements hold across the space of QA benchmarks, we measure the concurrence between 32 diverse QA benchmarks on a testbed of 20 representative modeling approaches introduced between 2016 and 2020.

Overall, we find that benchmarks that differ substantially still often have high concurrence. Human-constructed benchmarks (e.g., SQuAD and MRQA NaturalQuestions) have high concurrence with each other, despite differences in crowdsourcing setups, passage and question distributions, and even linguistic phenomena of focus (§3).

How different can a benchmark be, while still maintaining high concurrence with human-constructed benchmarks? In §4.1, we investigate the role of training dataset size by measuring concurrence with downsampled training datasets (e.g., using 20K SQuAD training examples rather than the full 88K). We find that downsampled training datasets are sufficient for high concurrence with other human-constructed benchmarks. In §4.2, we measure concurrence between human-constructed and programmatically-generated benchmarks (e.g., cloze-formatted or synthetic) to better understand the importance of human-written questions and passages. We find that cloze-formatted benchmarks have high concurrence with human-constructed benchmarks, so human-written questions and passages are not strictly necessary for concurrence. However, programmatically-generated synthetic benchmarks (e.g., the bAbI task suite) have low concurrence. Having found this breaking point of low concurrence, we construct two minimal synthetic benchmarks that achieve high concurrence with human-constructed benchmarks, despite lacking linguistic structure. Intuitively, the benchmarks that concur with human-constructed benchmarks are those that require model capabilities that are also useful for better performance on human-constructed benchmarks (e.g., identifying paraphrase and lexical overlap; §4.3-4.5).

Our results have several implications for the future development of benchmarks and modeling approaches. To summarize:

1. Human-constructed benchmarks have high concurrence with each other on our testbed of 20 modeling approaches. The modeling approaches studied are not particularly benchmark-specific and that their modeling improvements largely hold across different benchmarks, despite intense community focus on a small number of benchmarks. This is especially true of recent modeling improvements driven by better pre-training, which is largely downstream benchmark-agnostic.
2. Many benchmarks require reasoning over predicate-argument structure (e.g., SQuAD, NewsQA, NaturalQuestions), and improvements on these benchmarks also transfer to more specialized benchmarks (e.g., HotpotQA or MRQA DROP) because (1) almost all benchmarks involve reasoning over predicate-argument structure and/or (2) better reasoning over predicate-argument structure is correlated with improvements on other phenomena.
3. Human-constructed benchmarks are not strictly necessary for improving performance on other human-constructed benchmarks. Synthetic benchmarks may be useful tools for isolating, understanding, and improving on particular model capabilities.
4. Downsampling benchmarks to as few as 10K training examples does not significantly affect concurrence, especially since recent pre-trained modeling approaches have greater sample efficiency. We recommend the community build benchmarks that are smaller but more challenging (e.g., harder/more expensive to label per-example).
5. Since human-constructed benchmarks have high concurrence amongst themselves, we encourage researchers to seek diversity and build benchmarks that explore qualitatively different modeling capabilities that push research in new directions.

2 Measuring Concurrence

Informally, we say that two benchmarks have high *concurrence* on a set of modeling approaches if the two benchmarks rank the modeling approaches similarly. We compare the performance of a modeling approach when trained and tested on one benchmark with its performance when trained and tested on another benchmark—we use each benchmark's original *i.i.d.* train-test split, so all evaluation is in-domain. Repeating this process for many modeling

approaches, we can assess whether performance gains *between* modeling approaches are generally preserved when moving between benchmarks.

Formally, define a benchmark B as a pair of datasets $(D_{\text{train}}, D_{\text{test}})$, where $D_{\text{train}} \subseteq \mathcal{X} \times \mathcal{Y}$ and $D_{\text{test}} \subseteq \mathcal{X} \times \mathcal{Y}$ for an input space $\mathcal{X}$ and an output space $\mathcal{Y}$. A *system* is a function $s : \mathcal{X} \rightarrow \mathcal{Y}$ (i.e., a trained model with a particular set of parameters). In contrast, a *modeling approach* (i.e., a neural architecture coupled with a training procedure) is a function a that takes in a training dataset D_{train} and outputs a system. Let EVAL denote an evaluation function, where $\text{EVAL}(a, B)$ returns the performance (under a given evaluation function, e.g., exact match) of a modeling approach a when trained on the train split of B and tested on the test split of B. Finally, $\text{CONCUR}(B_1, B_2; \mathcal{A}, \text{EVAL})$ is the *concurrence* between the benchmarks B_1 and B_2 with respect to a set of modeling approaches $\mathcal{A}$ and the evaluation function EVAL. Let $a \sim \text{uniform}(\mathcal{A})$, where $\text{uniform}(\mathcal{A})$ denotes the uniform distribution over the set of modeling approaches $\mathcal{A}$. Defining the random variables $P_1 = \text{EVAL}(a, B_1)$ and $P_2 = \text{EVAL}(a, B_2)$, we finally define

$$\text{CONCUR}(B_1, B_2; \mathcal{A}, \text{EVAL}) = \text{CORR}(P_1, P_2)\,,$$

where CORR is some correlation function.

We use the SQuAD exact match (EM) metric as our evaluation function EVAL, and we consider the Pearson correlation coefficient (r) and the Kendall rank correlation coefficient (τ) as our correlation functions CORR. The former measures whether the relationship between model performance on the two benchmarks is approximately linear, whereas the latter measures whether pairwise rank comparisons between models are preserved between benchmarks. As a rough guideline, we consider $\tau > 0.8$ to be high concurrence, though interpreting concurrence often requires more than comparing overall correlation.

Extractive QA modeling approaches. To assess concurrence in this work, we use a representative set of 20 diverse modeling approaches introduced between 2016 to 2020 ($\mathcal{A}$). These modeling approaches include RaSoR (Lee et al., 2016), BiDAF (Seo et al., 2017), DocumentReader (Chen et al., 2017), QANet (Yu et al., 2018), BiDAF++ (Clark and Gardner, 2018), MnemonicReader (Hu et al., 2017), FusionNet (Huang et al., 2018), BERT (Devlin et al., 2019), ALBERT (Lan et al., 2020), RoBERTa (Liu et al., 2019), ELECTRA (Clark et al., 2020), and SpanBERT (Joshi et al., 2020).[1]

10 of our 20 modeling approaches are *non-pretrained*. These approaches generally propose (1) better sequence encoders for passages and questions (e.g., Lee et al., 2016; Yang et al., 2017; Yu et al., 2018) and/or (2) improved attention mechanisms for question-passage interactions (e.g., Seo et al., 2017; Wang et al., 2017; Huang et al., 2018).

In contrast, the other 10 of our 20 modeling approaches are *pre-trained*; these modeling approaches all use the Transformer architecture (Vaswani et al., 2017), but improve performance by proposing better pre-training procedures and objectives. These pre-trained modeling approaches are generally evaluated on a suite of downstream tasks, in contrast to non-pretrained modeling approaches, which generally evaluate on a single benchmark.

All of these modeling approaches were originally evaluated on SQuAD, though several (e.g., SpanBERT) were also evaluated on other QA benchmarks. We evaluate each modeling approach on each benchmark with the same training hyperparameters used for SQuAD, as well as 5 additional randomly sampled hyperparameter settings.

Extractive QA benchmarks. In this work, we study concurrence between three broad classes of extractive QA benchmarks: (i) human-constructed, (ii) cloze, and (iii) synthetic. Human-constructed benchmarks contain human-written natural language questions and passages; examples include SQuAD, NewsQA (Trischler et al., 2017), and NaturalQuestions (Kwiatkowski et al., 2019). On the other hand, cloze benchmarks (e.g., Children's Book Test or CNN; Hill et al., 2016; Hermann et al., 2015) contain cloze questions, which are "fill-in-the-blank" statements with masked answers. These questions are usually automatically-generated from human-written natural language passages. Finally, synthetic benchmarks contain programmatically-generated questions and passages (e.g., the bAbI task suite; Weston et al., 2016).

3 Do Modeling Improvements Hold Across Human-Constructed Benchmarks?

Many extractive question answering benchmarks are human-constructed—they contain human-written natural language questions and passages.

[1] See Appendix A for more details about the modeling approaches used to calculate concurrence.

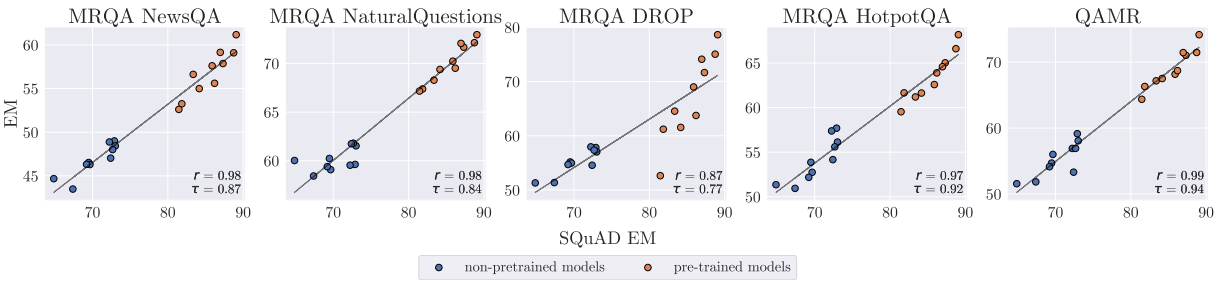

Figure 2: Human-constructed benchmarks have high concurrence with each other on both pre-trained and non-pretrained modeling approaches.

	MRQA NewsQA	MRQA NQ	MRQA DROP	MRQA HotpotQA	QAMR
SQuAD	0.87	0.84	0.77	0.92	0.94
MRQA NewsQA	-	0.82	0.83	0.92	0.87
MRQA NQ	0.82	-	0.69	0.80	0.80
MRQA DROP	0.83	0.69	-	0.79	0.83
MRQA HotpotQA	0.92	0.80	0.79	-	0.89

Table 1: Concurrence between human-constructed benchmarks. Despite differences in their crowdsourcing setup, passage and question distributions, and even linguistic phenomena of interest, human-constructed benchmarks generally have high concurrence (τ) with each other on our testbed of modeling approaches.

However, differences in the data collection procedure may yield benchmarks with dramatically different passage and question distributions. Do modeling improvements hold across benchmarks despite these differences?

Setup. We study the concurrence between six human-constructed benchmarks: SQuAD, NewsQA, NaturalQuestions, DROP (Dua et al., 2019), HotpotQA (Yang et al., 2018), and QAMR (Michael et al., 2018). We use the MRQA versions of NewsQA, NaturalQuestions, DROP, and HotpotQA (Fisch et al., 2019). Table 2 summarizes their high-level differences. See Appendix C.1 for examples from human-constructed benchmarks.

3.1 Results

Human-constructed benchmarks have high concurrence amongst themselves. Despite differences in benchmark crowdsourcing setups, passage and questions distributions, and even linguistic phenomena of interest, modeling improvements generally hold across human-constructed benchmarks (Table 1). Furthermore, concurrence is high over both non-pretrained and pre-trained modeling approaches (Figure 2).

For example, SQuAD, NewsQA, and NaturalQuestions differ in their passage-question joint relationship. In SQuAD, crowdworkers are employed to write questions given Wikipedia passages, but this results in questions with high lexical overlap with salient passage sentences. To minimize such overlap in NewsQA, crowdworkers write questions given only bullet-point summaries of the passages, rather than the passages themselves. Finally, questions in NaturalQuestions are written independently of their provided passage. These different crowdsourcing protocols drastically affect the ease and cost of benchmark construction, but SQuAD, NewsQA, and NaturalQuestions have high concurrence despite these differences.

Concurrence is high even when benchmarks focus on different phenomena. We also see that MRQA DROP and MRQA HotpotQA have surprisingly high concurrence with other human-constructed benchmarks (e.g., SQuAD and NaturalQuestions), despite their relatively specialized focus on particular linguistic phenomena (numerical and multi-hop reasoning, respectively).[2] This suggests that modeling improvements on benchmarks that target general reasoning over predicate-argument structure also improve performance on benchmarks that focus on different phenomena. We hypothesize this occurs because benchmarks are more similar than we'd otherwise expect (e.g., due to reasoning shortcuts; Min et al., 2019), and better reasoning over predicate-argument structure may be generally useful for other phenomena of interest.

4 Exploring the Limits of Concurrence

Our results in §3 indicate that human-constructed benchmarks have high concurrence with each other,

[2]Note that MRQA DROP is a subset of the original benchmark that removes questions with non-extractive answers (e.g., answer is the result of an arithmetic operation).

Benchmark	Question (Q)	Passage (P)	Phenomena of Interest	|Q|	|P|	Q ⫫ P
SQuAD	Crowdsourced	Wikipedia	Predicate-Argument Structure	11	137	✗
QAMR	Crowdsourced	Wikipedia	Predicate-Argument Structure	7	25	✗
NewsQA	Crowdsourced	News articles	Predicate-Argument Structure	8	599	✓
NaturalQuestions	Search logs	Wikipedia	Predicate-Argument Structure	9	153	✓
HotpotQA	Crowdsourced	Wikipedia	Multi-Hop Reasoning	22	232	✗
DROP	Crowdsourced	Wikipedia	Numerical Reasoning	11	243	✗

Table 2: Differences between the various human-constructed benchmarks evaluated. $Q \perp\!\!\!\perp P$ is true (✓) if the question was written independently from the associated passage. |Q| and |P| denote average question and passage token length, respectively.

despite differences in their phenomena of interest and passage and question distributions. Just how different can a benchmark be, while maintaining high concurrence with human-constructed benchmarks? In §4.1 we investigate the role of training dataset size on concurrence—while larger training datasets often yield better systems with higher end-task accuracy, are they necessary for comparing modeling approaches? In §4.2, we measure concurrence between human-constructed and cloze benchmarks to better understand the role of human-written questions and passages in concurrence. Cloze benchmarks have high concurrence with human-constructed benchmarks, indicating that human-written questions and passages are not necessary for concurrence with human-constructed benchmarks. To take this to an extreme, §4.3 evaluates concurrence between programmatically-generated synthetic benchmarks (the bAbI task suite) with human-constructed benchmarks. Our results show that the bAbI tasks have low concurrence with human-constructed benchmarks. Having found this breaking point, we work backwards to build a minimal benchmark with high concurrence, which will enable us to better understand sufficient conditions for concurrence. In §4.4, we construct a benchmark that has no linguistic structure or complex reasoning but still has high concurrence with human-constructed benchmarks over non-pretrained models. Finally, §4.5 shows that a synthetic benchmark that requires richer reasoning between question and passage tokens can achieve high concurrence with human-constructed benchmarks on *both* pre-trained and non-pretrained modeling approaches.

4.1 Downsampling Benchmarks

Many existing human-constructed extractive QA benchmarks contain a large number of examples, increasing their cost of construction. For example, SQuAD has 87,599 question-answer pairs in its training split. Are large training datasets necessary for comparing modeling approaches?

	Downsampled SQuAD Size				
	60K	40K	20K	20K	1K
SQuAD	0.96	0.96	0.94	0.87	0.77
MRQA NewsQA	0.92	0.92	0.89	0.89	0.77
MRQA NQ	0.84	0.84	0.81	0.78	0.63

Table 3: Beyond a baseline threshold of 20K examples, downsampling the SQuAD training set minimally affects concurrence with the full SQuAD benchmark and other human-constructed benchmarks

Setup. We study the extent to which subsamples of SQuAD concur with the full SQuAD benchmark (88K examples) and five other human-constructed benchmarks. We experiment with randomly generated subsets of the SQuAD training set with 1K, 10K, 20K, 40K, and 60K training examples. We use the original SQuAD development set (∼10K examples) for evaluation.

Results. Downsampling the SQuAD training set from 88K to 20K examples does not substantially affect concurrence with the full SQuAD benchmark and other human-constructed benchmarks (Table 3). Concurrence is high on both non-pretrained and pre-trained modeling approaches (Figure 3). Downsampling to 10K examples slightly reduces concurrence with non-pretrained modeling approaches. Concurrence with pre-trained models only begins to degrades when using 1K training examples, indicating that few-shot settings are likely categorically different and worth studying separately.

4.2 Cloze Benchmarks

To better understand the importance of human-written questions and passages, we measure concurrence between human-constructed benchmarks and cloze benchmarks. Cloze extractive question answering benchmarks contain cloze questions, which are "fill-in-the-blank" statements

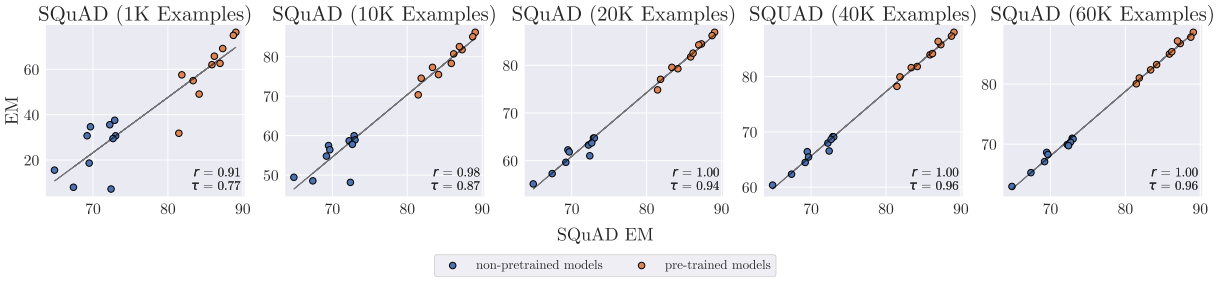

Figure 3: Downsampling the SQuAD training dataset can yield high concurrence with the full SQuAD benchmark on both pre-trained and non-pretrained modeling approaches. In particular, 10K training examples are sufficient for high concurrence on pre-trained models, and 20K examples yields high concurrence on non-pretrained mdoels.

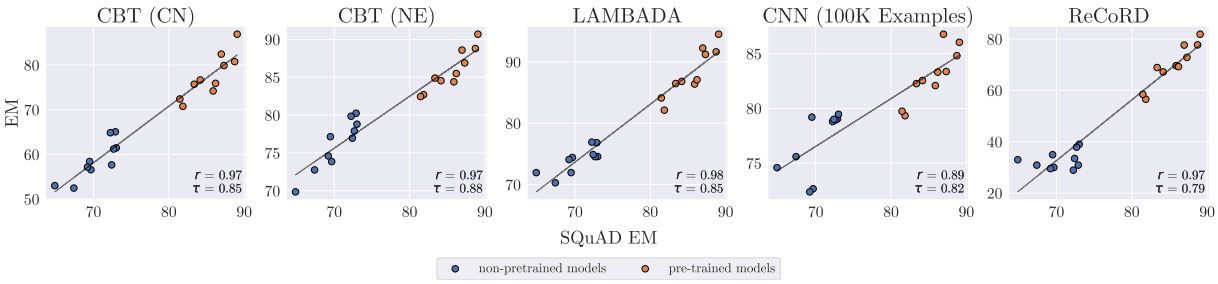

Figure 4: Despite their cloze-formatted question, which differ from questions typically found in human-constructed benchmarks, cloze benchmarks can have high concurrence with SQuAD (CBT-CN, CBT-NE, and LAMBADA), though this is not always the case (CNN, ReCoRD).

with masked answers. Large cloze benchmarks are cheap to construct because examples can be automatically generated by eliding spans from naturally-occurring text. Although the passages in cloze benchmarks are natural language, their fill-in-the-blank require more guessing from context, rather than the answer deduction typically found in human-constructed benchmarks.

Setup. We study the Children's Book Test (CBT; Hill et al., 2016), LAMBADA (Paperno et al., 2016), CNN (Hermann et al., 2015), and ReCoRD (Zhang et al., 2018) cloze benchmarks and measure their concurrence with human-constructed benchmarks on our testbed of modeling approaches. We follow prior work (Dhingra et al., 2017) and evaluate on subsets of CBT where the answer token is either a common noun (CBT-CN) or a named entity (CBT-NE). In addition, we use a subsampled version of the CNN benchmark with 100K training examples to save compute. See Appendix C.2 for examples from the cloze benchmarks we study.

Results. Despite using programmatically-generated cloze questions, cloze benchmarks (e.g., CBT and LAMBADA) can have high concurrence with human-constructed benchmarks (Table 4). On the other hand, CNN and ReCoRD have lower concurrence with human-constructed benchmarks, especially on non-pretrained modeling approaches—the performance improvements between pre-trained modeling approaches are still largely preserved (Figure 4).

	CBT (CN)	CBT (NE)	LAMBADA	CNN (100K)	ReCoRD
SQuAD	0.85	0.88	0.85	0.82	0.79
MRQA NewsQA	0.92	0.93	0.87	0.76	0.77
MRQA NQ	0.78	0.79	0.75	0.79	0.88

Table 4: Concurrence between programmatically-generated cloze benchmarks and human-constructed benchmarks can be high (e.g., CBT and LAMBADA), but not always (CNN and ReCoRD).

Concurrence on CNN is lower due to a pair of outlier modeling approaches—DocumentReader, with and without external linguistic features. We hypothesize that these models do poorly on CNN because some aspects of their preprocessing are SQuAD-specific; this may have also influenced architecture design. ReCoRD's low overall concurrence comes from the poor performance of non-pretrained modeling approaches. This may be due to ReCoRD's construction procedure, since a filtering step removed all examples that were correctly

answered by a strong non-pretrained modeling approach (SAN, with SQuAD dev. EM of 76.24; Liu et al., 2018). ReCoRD has low concurrence with SQuAD on modeling approaches that are weaker than SAN, and high concurrence on modeling approaches that outperform SAN.

4.3 High Concurrence Is Not Universal: Improvements Do Not Hold On bAbI

Having established that human-written passages are not necessary for high concurrence with human-constructed benchmarks (§4.2), we take this to an extreme by evaluating concurrence between human-constructed benchmarks and synthetic extractive question answering benchmarks, which contain questions and passages that are programmatically generated (and possibly not even natural language). The bAbI task suite contains 20 synthetic question-answering benchmarks, each of which focuses on a particular skill required by a competent dialogue system (e.g., fact retrieval, subject-object relations, counting). The textual data is generated from a simulated toy environment.

Setup. We consider the 11 tasks that can be losslessly converted to an extractive format (Tasks 1, 2, 3, 4, 5, 11, 12, 13, 14, 15, 16). For each task, we use the two officially-released data settings: one setting has 900 training examples and 100 development examples, and the other has 9,000 training examples and 1,000 development examples. In this section, we focus on the setting with 900 training examples, since all modeling approaches do nearly perfectly on almost all tasks with 9,000 examples (Appendix D.3). See Appendix C.3 for examples from the existing synthetic benchmarks we study.

Results and Discussion. The bAbI tasks have low concurrence with human-constructed benchmarks—high concurrence is not universal. Modeling approaches often have either near-perfect or near-random performance (Figure 5).

4.4 What is Sufficient for Concurrence on Non-Pretrained Modeling Approaches?

To better understand the sufficient conditions for concurrence with human-constructed benchmarks, we are interested in constructing a minimal synthetic benchmark with high concurrence. Given that human-written passages and questions are not necessary for high concurrence with human-constructed benchmarks (§4.2), but the programmatically-generated bAbI synthetic benchmarks have low concurrence (§4.3), we design a minimal synthetic benchmark with high concurrence with human-constructed benchmarks over non-pretrained modeling approaches.

Setup. Questions in extractive QA benchmarks can often be answered by exploiting lexical overlap between question and passage tokens (Weissenborn et al., 2017; Krishna et al., 2020). To better understand the limits of concurrence, we build a minimal synthetic cloze benchmark (FuzzySyntheticQA) that explicitly targets this fuzzy pattern-matching and find that it has high concurrence with SQuAD on non-pretrained modeling approaches. Figure 6 shows a sample passage and question-answering pairs. We use 10,000 questions for training and 10,000 questions for evaluation. See Appendix E for further details about FuzzySyntheticQA's construction.

Passage Generation. We generate the passage by randomly sampling 150 tokens from the uniform distribution over a token vocabulary. The token vocabulary is taken from the WikiText-2 training set (Merity et al., 2017) and has 68,429 types.

Answer Generation. The answer token is randomly selected from the generated passage.

Cloze Question Generation. To generate the cloze question, we first extract the answer token's local context (up to 10 tokens) and mask out the answer token. Then, we corrupt the cloze question by (1) randomly replacing its tokens with related tokens (100 approximate nearest neighbor tokens in the vocabulary, measured by vector distance in the pre-trained English FastText embeddings), (2) locally permuting its tokens (within 3 positions), and (3) applying word dropout (with rate 0.2).

Results and Discussion. FuzzySyntheticQA has high concurrence with human-constructed benchmarks, but only on non-pretrained modeling approaches—concurrence on pre-trained modeling approaches is much lower (Figure 7). Even benchmarks that lack much linguistic structure can have high concurrence with human-constructed benchmarks, as long as they require similar phenomena (in this case, fuzzy lexical matching between the question and passage).

Why do improvements in pre-training not hold on FuzzySyntheticQA? One potential reason is that passages in FuzzySyntheticQA lack of linguistic structure. To evaluate this hypothesis, we generate FuzzySyntheticQA questions from English Wikipedia passages, rather than sampling from the

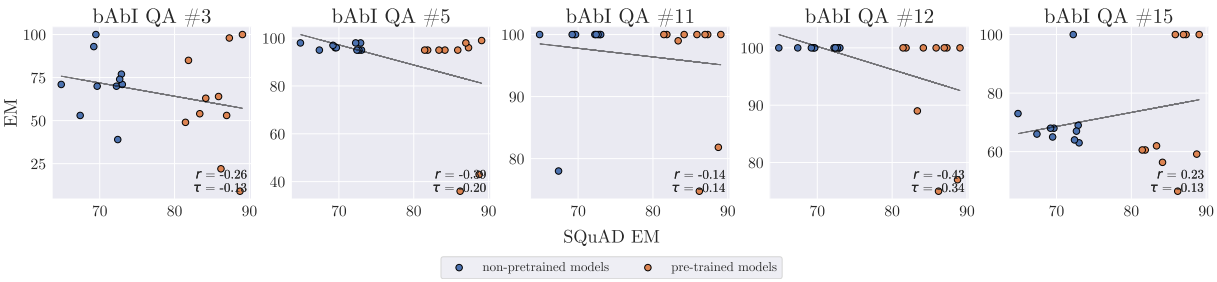

Figure 5: Modeling approaches perform perfectly or at near-chance performance on the bAbI tasks, limiting their ability to recapitulate historical findings on SQuAD (see Appendix D.3 for the full results on all tasks).

Passage Snippet: *... chests Melchior divorced might whereof 37th Kadima milling raved Salib melanocephala Pilgrims chop Prosser draftsmanship 203 Caesarius madam Deconstruction Guevara Amalia ...*
Question: *Pigs corncrake XXXXX 286 airmanship Kition gracious Modernism Raul*
Answer: *chop*

Figure 6: Example passage and question-answer pair from FuzzySyntheticQA.

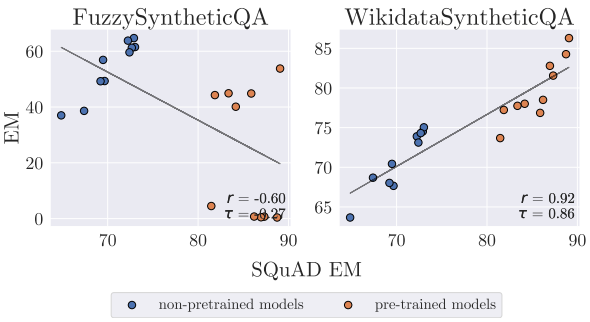

Figure 7: **Left:** FuzzySyntheticQA has high concurrence with SQuAD on non-pretrained modeling approaches, but pre-training does not increase performance, leading to low overall concurrence. **Right:** Despite lacking natural language structure, WikidataSyntheticQA has high concurrence with SQuAD.

uniform distribution over tokens, but this still results in low concurrence with human-constructed benchmarks on pre-trained modeling approaches ($r = -0.49$, $\tau = -0.19$), indicating that the low concurrence comes from more than just a lack of natural language passages (Appendix F).

4.5 What is Sufficient for Concurrence on Pre-Trained and Non-Pretrained Modeling Approaches?

Having found a minimal synthetic benchmark that achieves high concurrence with human-constructed benchmarks on non-pretrained modeling approaches (§4.4), we show that a synthetic

Passage Snippet: *Mae Jemison profession astronaut . STS-47 orbits completed 126.0 . STS-47 crew member Mae Carol Jemison . Mae Jemison worked for NASA . Mae C. Jemison award received Rachel Carson Award . Mae Jemison birthplace Decatur . ...*
Question: *human spaceflight orbits completed XXXXX*
Answer: *126.0*
Question: *Rachel Carson Award honor received by XXXXX*
Answer: *Mae C. Jemison*
Question: *Human XXXXX The River City*
Answer: *birthplace*

Figure 8: Example passage and question-answer pairs from WikidataSyntheticQA.

benchmark that requires richer reasoning between question and passage tokens is sufficient for high concurrence on *both* non-pretrained and pre-trained modeling approaches.

Setup. We construct WikidataSyntheticQA, a benchmark derived from Wikidata triples; Figure 8 shows a sample passage and question-answering pairs. Knowledge graphs like Wikidata are rich sources of complex relations between entities, which enables us to increase the complexity of question-passage token relations beyond the simple noising and corruptions of FuzzySyntheticQA. We use 10,000 questions for training and 9,835 question-answer pairs for evaluation. See Appendix G for further details about WikidataSyntheticQA's construction.

Wikidata Background. Wikidata is a knowledge graph connecting entities via relations. Wikidata entities and relations include a *label*, the most common name that an entity is known by, and *aliases*, alternative names for entities. For example, the entity Mae_C._Jemison has the label *"Mae C. Jemison"*, with aliases *"Mae Jemison"* and *"Mae Carol Jemison"*. We treat labels and aliases as potential surface realizations of entities and relations.

Generation Preliminaries. Generating a passage requires a set of Wikidata triples. To select these triples, we first randomly choose a seed entity from the 10,000 Wikidata entities with the highest PageRank score (Page et al., 1999). We then extract the triples from the seed entity and all entities connected to the seed entity. Finally, we randomly sample 50 triples for use in generation.

Passage Generation. Given the set of 50 Wikidata triples, we realize triples into textual surface forms by selecting a random Wikidata label or alias for each triple element. The final passage is formed by concatenating the realizations of all triples and adding a delimiter token between them to mimic sentential structure.

Answer Generation. We generate an answer span by selecting a random triple used in the passage generation process, and then choosing a random element of that triple. The passage realization of this random element is the answer span.

Cloze Question Generation. To generate the cloze question, we take the triple used for answer generation and mask out the particular element marked as the answer. We realize the non-answer triple elements into textual forms by selecting a random Wikidata label or alias for each triple element. Then, we optionally and randomly replace the predicate with its inverse (if one exists), reversing the subject and the object to maintain consistency. We also optionally and randomly replace the remaining unmasked entity (i.e., the triple subject or object that was not masked) with one of its hypernyms, challenging models' knowledge of such relations.

Results and Discussion. As Figure 7 shows, WikidataSyntheticQA has high concurrence with human-constructed benchmarks, despite its lack of natural language passages or questions.

We hypothesize that WikidataSyntheticQA has higher concurrence with human-constructed benchmarks than FuzzySyntheticQA because correctly answering its examples often requires reasoning about hypernymy relations between entities and inverse relations between predicates—it is conceivable that pre-trained modeling approaches are better-equipped to handle and use these lexical relations. In addition, the Wikidata aliases provide sufficient lexical variation such that the benchmark is not trivially solvable through string pattern-matching (removing aliases from the generation procedure results in near-perfect performance from all modeling approaches). In contrast, high performance on FuzzySyntheticQA simply requires matching similar tokens in the passage and question—models can achieve high performance by simply learning the similarity relationships in the FastText vector space.

5 Related Work

A recent line of work examines whether *systems* have overfit to particular test sets by taking existing systems and evaluating them on newly-constructed test sets (Recht et al., 2019; Yadav and Bottou, 2019; Miller et al., 2020). Recent work has also studied whether higher-performing systems are more robust by studying the correlation between in-domain and out-of-domain improvements (Taori et al., 2020; Djolonga et al., 2020).

In contrast, this work examines whether improvements from *modeling approaches* hold across benchmarks. We train and test modeling approaches on a variety of existing and newly-constructed benchmarks. In this regard, our work is similar to the study of Kornblith et al. (2019), who find that performance improvements on ImageNet are well-correlated with performance improvements on other benchmarks.

6 Conclusion

This work studies whether QA modeling improvements hold across the diverse landscape of QA benchmarks. We develop the notion of *concurrence*, which quantifies the similarity between benchmarks' rankings of modeling approaches. Experiments with 32 QA benchmarks and 20 diverse modeling approaches indicate that human-constructed benchmarks largely have high concurrence amongst themselves, even when their passage and question distributions or linguistic phenomena of focus are very different. To better understand how different benchmark attributes affect concurrence, we explore downsampled benchmarks and various programmatically-generated benchmarks, the latter having high concurrence only when they target phenomena that are also useful for better performance on human-constructed benchmarks (e.g., identifying paraphrase and lexical overlap). Our results indicate that the modeling improvements studied hold broadly, despite years of intense community focus on a small number of benchmarks.

Acknowledgements

We thank the anonymous reviewers for their feedback and comments that helped improve this work. NL was supported by an NSF Graduate Research Fellowship under grant number DGE-1656518. Other funding was provided by a PECASE Award.

Limitations

While we conducted an extensive set of experiments to gain a broad picture of whether modeling improvements hold between benchmarks, it is always possible to investigate more settings. While our study covers a representative set of 20 non-pretrained and pre-trained modeling approaches, it is conceivable that evaluating more modeling approaches (or a different set of modeling approaches) on additional benchmarks (or a different set of benchmarks) would have led to different results.

Furthermore, although we evaluate each modeling approach on each benchmark with the same training hyperparameters used for SQuAD, as well as 5 additional randomly sampled hyperparameter settings ($20 \times 32 \times 6 = 3840$ experiments in total), it is possible that the SQuAD hyperparameters for some modeling approaches happen to be more general than other modeling approaches. Ideally, each modeling approach would be individually tuned to maximize performance on every benchmark, but doing so requires prohibitive amounts of compute and researcher effort—we believe that our experiments have enough coverage with respect to hyperparameter optimization.

References

Erik Bernhardsson and the Annoy development team. 2020. github.com/spotify/annoy.

Danqi Chen, Adam Fisch, Jason Weston, and Antoine Bordes. 2017. Reading Wikipedia to answer open-domain questions. In *Proc. of ACL*.

Pengxiang Cheng and Katrin Erk. 2020. Attending to entities for better text understanding. In *Proc. of AAAI*.

Christopher Clark and Matt Gardner. 2018. Simple and effective multi-paragraph reading comprehension. In *Proc. of ACL*.

Kevin Clark, Minh-Thang Luong, Quoc V. Le, and Christopher D. Manning. 2020. ELECTRA: Pre-training text encoders as discriminators rather than generators. In *Proc. of ICLR*.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. BERT: Pre-training of deep bidirectional transformers for language understanding. In *Proc. of NAACL*.

Bhuwan Dhingra, Hanxiao Liu, Zhilin Yang, William W. Cohen, and Ruslan Salakhutdinov. 2017. Gated-attention readers for text comprehension. In *Proc. of ACL*.

Josip Djolonga, Jessica Yung, Michael Tschannen, Rob Romijnders, Lucas Beyer, Alexander Kolesnikov, Joan Puigcerver, Matthias Minderer, Alexander D'Amour, Dan Moldovan, Sylvan Gelly, Neil Houlsby, Xiaohua Zhai, and Mario Lucic. 2020. On robustness and transferability of convolutional neural networks. ArXiv:2007.08558.

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 *Proc. of NAACL*.

Adam Fisch, Alon Talmor, Robin Jia, Minjoon Seo, Eunsol Choi, and Danqi Chen. 2019. MRQA 2019 shared task: Evaluating generalization in reading comprehension. In *Proc. of MRQA*.

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 *Proc. of NLP-OSS*.

Karl Moritz Hermann, Tomas Kocisky, Edward Grefenstette, Lasse Espeholt, Will Kay, Mustafa Suleyman, and Phil Blunsom. 2015. Teaching machines to read and comprehend. *Proc. of NeurIPS*.

Felix Hill, Antoine Bordes, Sumit Chopra, and Jason Weston. 2016. The Goldilocks Principle: Reading children's books with explicit memory representations. In *Proc. of ICLR*.

Minghao Hu, Yuxing Peng, and Xipeng Qiu. 2017. Reinforced Mnemonic Reader for machine reading comprehension. ArXiv:1705.02798v3.

Hsin-Yuan Huang, Chenguang Zhu, Yelong Shen, and Weizhu Chen. 2018. FusionNet: Fusing via fully-aware attention with application to machine comprehension. In *Proc. of ICLR*.

Robin Jia and Percy Liang. 2017. Adversarial examples for evaluating reading comprehension systems. In *Proc. of EMNLP*.

Mandar Joshi, Danqi Chen, Yinhan Liu, Daniel S. Weld, Luke Zettlemoyer, and Omer Levy. 2020. SpanBERT: Improving pre-training by representing and predicting spans. *Transactions of the Association for Computational Linguistics*, 8:64–77.

Mandar Joshi, Eunsol Choi, Omer Levy, Daniel Weld, and Luke Zettlemoyer. 2019. pair2vec: Compositional word-pair embeddings for cross-sentence inference. In *Proc. of NAACL*.

Bernard Koch, Emily Denton, Alex Hanna, and Jacob Gates Foster. 2021. Reduced, reused and recycled: The life of a dataset in machine learning research. In *Proc. of NeurIPS Datasets and Benchmarks Track*.

Simon Kornblith, Jonathon Shlens, and Quoc V. Le. 2019. Do better ImageNet models transfer better? In *Proc. of CVPR*.

Kalpesh Krishna, Gaurav Singh Tomar, Ankur P. Parikh, Nicolas Papernot, and Mohit Iyyer. 2020. Thieves on sesame street! model extraction of BERT-based APIs. In *Proc. of ICLR*.

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

Zhenzhong Lan, Mingda Chen, Sebastian Goodman, Kevin Gimpel, Piyush Sharma, and Radu Soricut. 2020. ALBERT: A lite BERT for self-supervised learning of language representations. In *Proc. of ICLR*.

Kenton Lee, Shimi Salant, Tom Kwiatkowski, Ankur Parikh, Dipanjan Das, and Jonathan Berant. 2016. Learning recurrent span representations for extractive question answering. ArXiv:1611.01436.

Xiaodong Liu, Yelong Shen, Kevin Duh, and Jianfeng Gao. 2018. Stochastic answer networks for machine reading comprehension. In *Proc. of ACL*.

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

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

Stephen Merity, Caiming Xiong, James Bradbury, and Richard Socher. 2017. Pointer sentinel mixture models. In *Proc. of ICLR*.

Julian Michael, Gabriel Stanovsky, Luheng He, Ido Dagan, and Luke Zettlemoyer. 2018. Crowdsourcing question-answer meaning representations. In *Proc. of NAACL*.

John Miller, Karl Krauth, Benjamin Recht, and Ludwig Schmidt. 2020. The effect of natural distribution shift on question answering models. *Proc. of ICML*.

Sewon Min, Eric Wallace, Sameer Singh, Matt Gardner, Hannaneh Hajishirzi, and Luke Zettlemoyer. 2019. Compositional questions do not necessitate multi-hop reasoning. In *Proc. of ACL*.

Lawrence Page, Sergey Brin, Rajeev Motwani, and Terry Winograd. 1999. The PageRank citation ranking: Bringing order to the web. Technical report, Stanford InfoLab.

Denis Paperno, Germán Kruszewski, Angeliki Lazaridou, Ngoc Quan Pham, Raffaella Bernardi, Sandro Pezzelle, Marco Baroni, Gemma Boleda, and Raquel Fernández. 2016. The LAMBADA dataset: Word prediction requiring a broad discourse context. In *Proc. of ACL*.

Alec Radford, Jeff Wu, Rewon Child, David Luan, Dario Amodei, and Ilya Sutskever. 2019. Language models are unsupervised multitask learners. Technical report, OpenAI.

Pranav Rajpurkar, Robin Jia, and Percy Liang. 2018. Know what you don't know: Unanswerable questions for SQuAD. In *Proc. of ACL*.

Pranav Rajpurkar, Jian Zhang, Konstantin Lopyrev, and Percy Liang. 2016. SQuAD: 100,000+ questions for machine comprehension of text. In *Proc. of EMNLP*.

Benjamin Recht, Rebecca Roelofs, Ludwig Schmidt, and Vaishaal Shankar. 2019. Do ImageNet classifiers generalize to ImageNet? *Proc. of ICML*.

Minjoon Seo, Aniruddha Kembhavi, Ali Farhadi, and Hannaneh Hajishirzi. 2017. Bidirectional attention flow for machine comprehension. In *Proc. of ICLR*.

Alon Talmor and Jonathan Berant. 2019. MultiQA: An empirical investigation of generalization and transfer in reading comprehension. In *Proc. of ACL*.

Rohan Taori, Achal Dave, Vaishaal Shankar, Nicholas Carlini, Benjamin Recht, and Ludwig Schmidt. 2020. Measuring robustness to natural distribution shifts in image classification. In *Proc. of NeurIPS*.

Adam Trischler, Tong Wang, Xingdi Yuan, Justin Harris, Alessandro Sordoni, Philip Bachman, and Kaheer Suleman. 2017. NewsQA: A machine comprehension dataset. In *Proc. of RepL4NLP*.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, Łukasz Kaiser, and Illia Polosukhin. 2017. Attention is all you need. In *Proc. of NeurIPS*.

Wenhui Wang, Nan Yang, Furu Wei, Baobao Chang, and Ming Zhou. 2017. Gated self-matching networks for reading comprehension and question answering. In *Proc. of ACL*.

Dirk Weissenborn, Georg Wiese, and Laura Seiffe. 2017. Making neural QA as simple as possible but not simpler. In *Proc. of CoNLL*.

Jason Weston, Antoine Bordes, Sumit Chopra, Sasha Rush, Bart van Merrienboer, Armand Joulin, and Tomas Mikolov. 2016. Towards AI-complete question answering: A set of prerequisite toy tasks. In *Proc. of ICLR*.

Thomas Wolf, Lysandre Debut, Victor Sanh, Julien Chaumond, Clement Delangue, Anthony Moi, Pierric Cistac, Tim Rault, Rémi Louf, Morgan Funtowicz, Joe Davison, Sam Shleifer, Patrick von Platen, Clara Ma, Yacine Jernite, Julien Plu, Canwen Xu, Teven Le Scao, Sylvain Gugger, Mariama Drame, Quentin Lhoest, and Alexander M. Rush. 2020. Transformers: State-of-the-art natural language processing. In *Proc. of EMNLP (System Demonstrations)*.

Felix Wu, Boyi Li, Lequn Wang, Ni Lao, John Blitzer, and Kilian Q. Weinberger. 2019. FastFusionNet: New state-of-the-art for DAWNBench SQuAD. ArXiv:1902.11291.

Chhavi Yadav and Léon Bottou. 2019. Cold case: The lost MNIST digits. In *Proc. of NeurIPS*.

Zhilin Yang, Bhuwan Dhingra, Ye Yuan, Junjie Hu, William W. Cohen, and Ruslan Salakhutdinov. 2017. Words or characters? fine-grained gating for reading comprehension. In *Proc. of ICLR*.

Zhilin Yang, Peng Qi, Saizheng Zhang, Yoshua Bengio, William Cohen, Ruslan Salakhutdinov, and Christopher D. Manning. 2018. HotpotQA: A dataset for diverse, explainable multi-hop question answering. In *Proc. of EMNLP*.

Adams Wei Yu, David Dohan, Minh-Thang Luong, Rui Zhao, Kai Chen, Mohammad Norouzi, and Quoc V. Le. 2018. QANet: Combining local convolution with global self-attention for reading comprehension. In *Proc. of ICLR*.

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:1810.12885.

Appendices

A Implementation Details of Modeling Approaches Evaluated

We evaluated a representative subset of 20 extractive question answering modeling approaches, published between 2016 to 2020 (Table 5). Below, we describe implementation details for all the modeling approaches evaluated.

Modeling Approach	SQuAD 1.1 Dev. EM	
	Our Reproduction	Published
RaSoR	64.9	66.4
BiDAF	67.4	67.7
DocumentReader	69.7	69.5
DocumentReader (no external features)	69.2	-
BiDAF++	69.5	71.6
MnemonicReader	73.0	71.8
MnemonicReader (no external features)	72.7	-
QANet	72.4	73.6
FusionNet	72.9	75.0
FusionNet (no external features)	72.2	-
BERT (base, uncased)	81.5	80.8
BERT (large, uncased)	84.2	84.1
BERT (large, uncased, whole-word masking)	87.3	86.7
ALBERT (base, V1)	81.9	82.3
ALBERT (xxlarge, V1)	89.1	89.3
RoBERTa (base)	83.4	-
RoBERTa (large)	87.0	88.9
ELECTRA (base)	85.9	84.5
SpanBERT (base)	86.2	-
SpanBERT (large)	88.7	88.1

Table 5: Published and reproduced SQuAD 1.1 EM of all 20 modeling approaches used for assessing concurrence. "-" indicates that the modeling approach has no published SQuAD 1.1 EM result.

RaSoR We reimplement the RaSoR model of (Lee et al., 2016) with PyTorch in the AllenNLP (Gardner et al., 2018) framework, following the original paper as closely as possible. While the authors released an implementation of their method (github.com/shimisalant/rasor), the codebase is in Theano and inexplicably fails on passages that are significantly longer than those found in SQuAD (e.g., those found in the CNN benchmark).

BiDAF We use the reimplementation of BiDAF (Seo et al., 2017) found in AllenNLP (Gardner et al., 2018).

DocumentReader (with and without external features) We use an reimplementation of DocumentReader (Chen et al., 2017) released at github.com/felixgwu/FastFusionNet. The original DocumentReader approach uses external features from a part-of-speech tagger and named entity recognition system. To fairly compare to systems that do not use such external resources, we also run the models without these features. We keep the hand-crafted term-frequency and token exact match features defined in the DocumentReader paper.

We also make some changes to the DocumentReader preprocessing code. In particular, the original implementation (github.com/facebookresearch/DrQA) of these two modeling approaches (intended for training and evaluation on SQuAD) replaces all tokens without a pre-trained GloVe embedding (trained on 840B tokens from the Common Crawl) with a special unknown token—the reimplementation we use adopts the same practice. This preprocessing assumption works well for SQuAD, since the vast majority of SQuAD tokens also appear in the GloVe vocabulary. However, this preprocessing assumption does

not apply to CNN—many of the special @entity*N* and @placeholder markers, which anonymize entities to prevent models from deriving answers from world knowledge, are not in the GloVe vocabulary. As a result, the original DocumentReader implementation maps them all to a single unknown token, effectively preventing the model from telling valid answer choices apart and yielding a model that performs no better than the majority baseline. Keeping these special tokens in the model's vocabulary enables differentiating between different entities in a passage, which naturally improves performance (and are the reported numbers)—however, the modeling approaches' improvements on SQuAD still do not transfer to CNN.

BiDAF++ We modify an AllenNLP (Gardner et al., 2018) reimplementation of the BiDAF++ Clark and Gardner (2018) model originally used in pair2vec (Joshi et al., 2019) for evaluation on SQuAD 2.0 (Rajpurkar et al., 2018).

MnemonicReader We use an reimplementation of MnemonicReader (Hu et al., 2017; note the specific arXiv revision) released at github.com/HKUST-KnowComp/MnemonicReader. In particular, the reimplementation is of the vanilla MnemonicReader without reinforcement learning.

QANet We use the reimplementation of QANet (Yu et al., 2018) found in AllenNLP (Gardner et al., 2018). This reimplementation was used as a baseline method for DROP (Dua et al., 2019).

FusionNet We use an reimplementation of FusionNet (Chen et al., 2017) released at github.com/felixgwu/FastFusionNet. This reimplementation was used as a baseline in Wu et al. (2019). Drawing inspiration from DocumentReader, the FusionNet approach also uses external features from a part-of-speech tagger and named entity recognition system. As a result, we also run the models without these features to fairly compare to systems that do not use such external resources. We keep the hand-crafted term-frequency and token exact match features originally used in the FusionNet paper.

BERT (base, large, and wwm) We use the HuggingFace Transformers (Wolf et al., 2020) library to fine-tune BERT (Devlin et al., 2019) on extractive question answering benchmarks. In particular, we use the base, uncased, BERT pre-trained model, the large, uncased, BERT pre-trained model, and the large, uncased, BERT model pre-trained with whole-word masking.

ALBERT (base and xxlarge) We use the HuggingFace Transformers (Wolf et al., 2020) library to fine-tune ALBERT (Lan et al., 2020) on extractive question answering benchmarks. In particular, we use the base and xxlarge V1 ALBERT pre-trained models.

RoBERTa (base and large) We use the HuggingFace Transformers (Wolf et al., 2020) library to fine-tune RoBERTa (Liu et al., 2019) on extractive question answering benchmarks. In particular, we use the base and large RoBERTa pre-trained models.

ELECTRA (base) We use the HuggingFace Transformers (Wolf et al., 2020) library to fine-tune the ELECTRA base discriminator (Clark et al., 2020) on extractive question answering benchmarks.

SpanBERT (base and large) We use the author-released codebase (github.com/facebookresearch/SpanBERT) to fine-tune SpanBERT (Joshi et al., 2020) on extractive question answering benchmarks. In particular, we use the base and large SpanBERT pre-trained models.

B Preprocessing Existing Benchmarks

B.1 Existing Human-Constructed Benchmarks

We use the MRQA NewsQA, MRQA DROP, and MRQA HotpotQA benchmarks exactly as released by the MRQA 2019 shared task (Fisch et al., 2019). The passages in MRQA NaturalQuestions contain HTML entities (e.g., <P> and </P>). The tokenizers used in non-pretrained models frequently split these entities into separate tokens. For example, <P> may become <, P, and >. This is problematic because the entities are quite common in passages, and expanding them during tokenization drastically increases the passage lengths, which some non-pretrained modeling approaches cannot handle due to GPU memory limits. HTML entities are tokenized like this because they contain non-alphanumeric characters. As a result, we normalize HTML entities by replacing the non-alphanumeric characters. For example, <P> becomes BPB, and </P> becomes EEPE. These tokens are correctly kept intact. It's possible that

modeling approaches that use subword information will perform worse with these normalized HTML entities, but we empirically observe that this normalization does not have a measurable impact on model performance.

QAMR questions were originally collected at the sentence level, but we concatenate these sentences to reconstruct the original passages they were sourced from. We then pair these reconstructed passages with the original QAMR questions. It's possible for questions to become unanswerable at the passage-level. One case of his happens when two sentences have the same question—we filter out such questions that are asked for multiple sentences in a reconstructed passage. Questions can also become unanswerable if relations between entities change between sentences. For example, given the passage "Bill lived in California in 1920. Bill lived in Washington in 1921.", the question "Where did Bill live" is answerable within the context of a particular sentence, but not in the context of the entire passage. Manual examination of generated QAMR passages and questions suggests that this case is rather uncommon, but it may still introduce a small amount of noise into the benchmark.

B.2 Existing Cloze Benchmarks

To convert the CBT and CNN benchmarks to extractive format, we take the passages and question as-is. The answer span is designated as the first occurrence of the answer token in the passage. To convert LAMBADA into extractive format, we follow the setup of Cheng and Erk (2020). The ReCoRD benchmark is used as-is, since it includes span-level annotations of answer tokens in passages.

B.3 Existing Synthetic Benchmarks

We consider tasks 1, 2, 3, 4, 5, 11, 12, 13, 14, 15, 16. The other tasks cannot be converted to extractive format (e.g., they require "yes"/"no" answers that do not appear in passages). To convert the tasks in the bAbI benchmark to extractive format, we take the passages and question as-is. While the bAbI benchmark does not provide character-level span annotations for answers, questions come with "supporting facts"—sentences in the passage that contain the answer. Thus, choose the first occurrence of the answer token in the supporting fact sentence as our answer span.

Some of the bAbI tasks, while usable in an extractive format in theory, cannot be trivially converted to the extractive format via the procedure above because the released benchmark's annotations do not appear in the passage. For instance, consider Figure 9, which shows an example drawn from the training set of Task 15. The answer provided in the benchmark is "cat", although this token never appears in the passage—instead, "cats" does. In cases where the originally-labeled answer cannot be found in the supporting fact, but its pluralization is present, we use the pluralized answer as our answer span.

Passage: *Mice are afraid of cats. Gertrude is a mouse. Emily is a mouse. Wolves are afraid of sheep. Winona is a wolf. Jessica is a mouse. Cats are afraid of sheep. Sheep are afraid of cats.*

Question: *What is jessica afraid of?*
Answer: *cat*

Figure 9

C Examples From Existing Benchmarks

C.1 Examples From Existing Human-Constructed Benchmarks

Table 6 shows examples from the existing human-constructed benchmarks we study.

Benchmark	**Passage** (some parts shortened with ...)	**Question**	**Answer**
MRQA NewsQA	(CNET) – When Facebook Chief Executive Mark Zuckerberg recently announced a "Like" button that publishers could place on their Web pages, he predicted it would make the Web smarter and "more social". What Zuckerberg didn't point out is that widespread use of the Like button allows Facebook to track people as they switch from CNN.com to Yelp.com to ESPN.com, all of which are sites that have said they will implement the feature...	What does the like button allow?	Facebook to track people
MRQA NaturalQuestions	BPB A shooting schedule is a project plan of each day 's shooting for a film production . It is normally created and managed by the assistant director , who reports to the production manager managing the production schedule . Both schedules represent a timeline stating where and when production resources are used . EEPE	who 's job is it to schedule each day 's shooting	assistant director
MRQA DROP	Coming off their win over the Chargers, the Bills flew to Dolphin Stadium for a Week 8 AFC East duel with the Miami Dolphins. In the first quarter, Buffalo trailed early as Dolphins QB Chad Pennington completed a 2-yard TD pass to TE Anthony Fasano. The Bills responded with kicker Rian Lindell getting a 19-yard field goal. In the second quarter, Buffalo took the lead as Lindell got a 43-yard and a 47-yard field goal...	Which team allowed the most first half points?	Dolphins
MRQA HotpotQA	[PAR] [TLE] John M. Brown [SEP] John Mifflin Brown (September 8, 1817 – March 16, 1893) was a bishop in the African Methodist Episcopal (AME) church. He was a leader in the underground railroad. He helped open a number of churches and schools, including the Payne Institute which became Allen University in Columbia, South Carolina and Paul Quinn College in Waco, Texas. He was also an early principal of Union Seminary which became Wilberforce University [PAR] [TLE] Waco, Texas [SEP] Waco ( ) is a city which is the county seat of McLennan County, Texas, United States. It is situated along the Brazos River and I-35, halfway between Dallas and Austin. The city had a 2010 population of 124,805, making it the 22nd-most populous city in the state. The US Census 2016 population estimate is 134,432 The Waco Metropolitan Statistical Area consists of McLennan and Falls Counties, which had a 2010 population of 234,906. Falls County was added to the Waco MSA in 2013. The US Census 2016 population estimate for the Waco MSA is 265,207.	What city is the home to Paul Quinn College and sets on the Brazos River between Dallas and Austin?	Waco, Texas
QAMR	An additional problem to face the empire came as a result of the involvement of Emperor Maurice -LRB- r. 582 – 602 -RRB- in Persian politics when he intervened in a succession dispute . This led to a period of peace , but when Maurice was overthrown , the Persians invaded and during the reign of Emperor Heraclius -LRB- r. 610 – 641 -RRB- controlled large chunks of the empire , including Egypt , Syria , and Anatolia until Heraclius ' successful counterattack . In 628 the empire secured a peace treaty and recovered all of its lost territories .	Whose politics did the empire get involved with?	Persian

Table 6: Example passages, questions, and answers from the existing human-constructed benchmarks we study.

C.2 Examples From Existing Cloze Benchmarks

Table 7 shows examples from the existing cloze benchmarks we study.

Benchmark	**Passage** (some parts shortened with ...)	**Question**	**Answer**
Children's Book Test (Common Nouns)	... Lady Latifa argued and urged her wishes , but in vain ; the prince was not to be moved . Then she called to the cupbearers for new wine , for she thought that when his head was hot with it he might consent to stay . The pure , clear wine was brought ; she filled a cup and gave to him . He said : ' O most enchanting sweetheart ! it is the rule for the host to drink first and then the guest . '	So to make him lose his head , she drained the XXXXX ; then filled it again and gave him .	cup
Children's Book Test (Named Entities)	... At last , however , the Sunball became aware how sad Letiko was . ... Then he sent them away , and called two hares to him , and said : ' Will you take Letiko home to her mother ? ' ' Yes , why not ? ' ' What will you eat and drink if you should become hungry and thirsty by the way ? ' ' We will eat grass and drink from streamlets . ' ' Then take her , and bring her home . '	Then the hares set out , taking XXXXX with them , and because it was a long way to her home they became hungry by the way .	Letiko
LAMBADA	sorry 's not going to win me my game tomorrow . my racket is . i ca n't believe i let you take it out of here in the first place ! " " but , dad , i 'm sure you made mistakes when you were a hippie teenager ! " " and i paid for them !	like you 're going to pay for my	racket
CNN	(@entity0) you 'll see some familiar faces in the @entity1 . @entity2 beat @entity3 66 - 52 on sunday , giving @entity4 ' coach @entity5 his 12th trip to the semifinals of the @entity6 men 's basketball tournament . @entity7 and @entity8 each scored 16 to help @entity2 win the @entity9 . @entity3 , led by 16 points from @entity10 , was hoping to earn its first trip to the @entity1 . here 's how the @entity1 , to be played in @entity11 , has shaped up : next saturday , @entity2 will face @entity12 in the first semifinal . in the next game , top seed @entity13 will battle @entity14	the @entity1 matchups : @placeholder vs. @entity12 and @entity13 vs. @entity14	@entity2
ReCoRD	Secretary of State Hillary Clinton on Monday tried to douse a political firestorm over the deadly assault on a U.S. diplomatic mission in Libya, saying she's responsible for the security of American diplomatic outposts. "I take responsibility," Clinton told CNN in an interview while on a visit to Peru. "I'm in charge of the State Department's 60,000-plus people all over the world, 275 posts. The president and the vice president wouldn't be knowledgeable about specific decisions that are made by security professionals. They're the ones who weigh all of the threats and the risks and the needs and make a considered decision." @highlight "What I want to avoid is some kind of political gotcha or blame game," Clinton says @highlight "I take this very personally," she says @highlight Diplomats need security but "can't hang out behind walls," she adds	Clinton also described a desperate scene in the @placeholder during the hours of the attack, as staff tried to find out what had happened.	State Department

Table 7: Example passages, questions, and answers from the existing cloze benchmarks we study.

C.3 Examples From Existing Synthetic Benchmarks

Table 8 shows examples from the existing synthetic benchmarks we study. The contents of this table are reproduced from Weston et al. (2016).

Benchmark	Passage	Question	Answer
bAbI Task 1 (Single Supporting Fact)	Mary went to the bathroom. John moved to the hallway. Mary travelled to the office.	Where is Mary?	office
bAbI Task 2 (Two Supporting Facts)	John is in the playground. John picked up the football. Bob went to the kitchen.	Where is the football?	playground
bAbI Task 3 (Three Supporting Facts)	John picked up the apple. John went to the office. John went to the kitchen. John dropped the apple.	Where was the apple before the kitchen?	office
bAbI Task 4 (Two Argument Relations)	The office is north of the bedroom. The bedroom is north of the bathroom. The kitchen is west of the garden.	What is north of the bedroom?	office
bAbI Task 5 (Three Argument Relations)	Mary gave the cake to Fred. Fred gave the cake to Bill. Jeff was given the milk by Bill.	Who did Fred give the cake to?	Bill
bAbI Task 11 (Basic Coreference)	Daniel was in the kitchen. Then he went to the studio. Sandra was in the office.	Where is Daniel?	studio
bAbI Task 12 (Conjunction)	Mary and Jeff went to the kitchen. Then Jeff went to the park.	Where is Jeff?	park
bAbI Task 13 (Compound Coreference)	Daniel and Sandra journeyed to the office. Then they went to the garden. Sandra and John travelled to the kitchen. After that they moved to the hallway.	Where is Daniel?	garden
bAbI Task 14 (Time Reasoning)	In the afternoon Julie went to the park. Yesterday Julie was at school. Julie went to the cinema this evening.	Where did Julie go after the park?	cinema
bAbI Task 15 (Basic Deduction)	Sheep are afraid of wolves. Cats are afraid of dogs. Mice are afraid of cats. Gertrude is a sheep.	What is Gertrude afraid of?	wolves
bAbI Task 16 (Basic Induction)	Lily is a swan. Lily is white. Bernhard is green. Greg is a swan.	What color is Greg?	white

Table 8: Example passages, questions, and answers from the existing synthetic benchmarks we study.

D Full Results on Existing Benchmarks

D.1 Full Results on Existing Human-Constructed Benchmarks

Table 9 and Table 10 show the performance of each modeling approach on each existing human-constructed benchmark.

	MRQA NewsQA	MRQA NaturalQuestions	MRQA DROP
RaSoR	44.68	60.02	51.30
BiDAF	43.49	58.43	51.36
DocumentReader	46.30	59.08	54.96
DocumentReader (no external features)	46.32	59.39	54.69
BiDAF++	46.53	60.23	55.16
MnemonicReader	48.43	61.53	57.02
MnemonicReader (no external features)	48.01	61.80	57.35
QANet	47.03	61.74	54.56
FusionNet	49.00	59.62	57.82
FusionNet (no external features)	48.88	59.54	57.95
BERT (base, uncased)	52.61	67.16	52.63
BERT (large, uncased)	54.99	69.38	61.54
BERT (large, uncased, whole-word masking)	57.86	71.67	71.66
ALBERT (base, V1)	53.25	67.37	61.21
ALBERT (xxlarge, V1)	61.16	72.95	78.64
RoBERTa (base)	56.62	68.28	64.54
RoBERTa (large)	59.14	72.06	74.12
ELECTRA (base)	57.60	70.23	69.00
SpanBERT (base)	55.60	69.51	63.74
SpanBERT (large)	59.09	72.13	75.05

Table 9: Performance of modeling approaches when evaluated on MRQA NewsQA, MRQA NaturalQuestions and MRQA DROP.

	MRQA HotpotQA	QAMR
RaSoR	51.35	51.56
BiDAF	50.94	51.84
DocumentReader	52.74	56.00
DocumentReader (no external features)	52.18	54.14
BiDAF++	53.86	54.69
MnemonicReader	56.13	58.07
MnemonicReader (no external features)	55.60	56.92
QANet	54.16	53.31
FusionNet	57.69	59.14
FusionNet (no external features)	57.38	56.91
BERT (base, uncased)	59.53	64.36
BERT (large, uncased)	61.63	67.51
BERT (large, uncased, whole-word masking)	65.02	71.03
ALBERT (base, V1)	61.65	66.30
ALBERT (xxlarge, V1)	68.17	74.15
RoBERTa (base)	61.19	67.16
RoBERTa (large)	64.58	71.44
ELECTRA (base)	62.58	68.16
SpanBERT (base)	63.89	68.70
SpanBERT (large)	66.60	71.46

Table 10: Performance of modeling approaches when evaluated on MRQA HotpotQA and QAMR.

D.2 Full Results on Existing Cloze Benchmarks

Table 11 and Table 12 show the performance of each modeling approach on each existing cloze benchmark.

	CBT (CN)	CBT (NE)	LAMBADA
RaSoR	53.00	69.85	71.95
BiDAF	52.45	72.75	70.29
DocumentReader	56.55	73.85	74.42
DocumentReader (no external features)	57.15	74.60	74.08
BiDAF++	58.40	77.15	71.95
MnemonicReader	61.45	78.80	74.57
MnemonicReader (no external features)	61.20	77.90	74.55
QANet	57.65	76.95	74.89
FusionNet	65.05	80.25	76.83
FusionNet (no external features)	64.85	79.85	76.92
BERT (base, uncased)	72.40	82.45	84.13
BERT (large, uncased)	76.65	84.55	86.83
BERT (large, uncased, whole-word masking)	79.90	86.90	91.23
ALBERT (base, V1)	70.75	82.70	82.14
ALBERT (xxlarge, V1)	86.90	90.70	94.53
RoBERTa (base)	75.70	84.90	86.48
RoBERTa (large)	82.45	88.60	92.27
ELECTRA (base)	74.20	84.40	86.40
SpanBERT (base)	75.90	85.50	87.10
SpanBERT (large)	80.75	88.80	91.65

Table 11: Performance of modeling approaches when evaluated on CBT (CN), CBT (NE) and LAMBADA.

	CNN (100K Examples)	ReCoRD
RaSoR	74.59	32.97
BiDAF	75.59	30.88
DocumentReader	72.66	29.97
DocumentReader (no external features)	72.38	29.52
BiDAF++	79.20	34.93
MnemonicReader	79.46	39.01
MnemonicReader (no external features)	78.95	37.87
QANet	79.00	33.46
FusionNet	79.05	30.89
FusionNet (no external features)	78.80	28.91
BERT (base, uncased)	79.74	58.45
BERT (large, uncased)	82.54	67.18
BERT (large, uncased, whole-word masking)	82.72	72.85
ALBERT (base, V1)	79.33	56.54
ALBERT (xxlarge, V1)	86.03	81.87
RoBERTa (base)	82.26	68.88
RoBERTa (large)	86.77	77.63
ELECTRA (base)	82.08	69.61
SpanBERT (base)	83.31	69.23
SpanBERT (large)	84.81	77.72

Table 12: Performance of modeling approaches when evaluated on CNN (100K Examples) and ReCoRD.

D.3 Full Results on Existing Synthetic Benchmarks

Table 13 and Table 14 and Table 15 show the performance of each modeling approach on each existing of the bAbI tasks (900 training examples).

	bAbI QA #1	bAbI QA #2	bAbI QA #3	bAbI QA #4
RaSoR	100.0	60.0	71.0	81.0
BiDAF	100.0	42.0	53.0	83.0
DocumentReader	100.0	63.0	70.0	100.0
DocumentReader (no external features)	100.0	76.0	93.0	100.0
BiDAF++	100.0	100.0	100.0	78.0
MnemonicReader	100.0	44.0	71.0	100.0
MnemonicReader (no external features)	100.0	100.0	74.0	100.0
QANet	100.0	42.0	39.0	85.0
FusionNet	100.0	84.0	77.0	100.0
FusionNet (no external features)	100.0	100.0	70.0	100.0
BERT (base, uncased)	100.0	80.0	49.0	81.0
BERT (large, uncased)	100.0	63.0	63.0	79.0
BERT (large, uncased, whole-word masking)	100.0	98.0	98.0	91.0
ALBERT (base, V1)	100.0	86.0	85.0	85.0
ALBERT (xxlarge, V1)	100.0	100.0	100.0	100.0
RoBERTa (base)	100.0	73.0	54.0	64.0
RoBERTa (large)	100.0	39.0	53.0	87.0
ELECTRA (base)	100.0	86.0	64.0	100.0
SpanBERT (base)	57.0	9.0	22.0	60.0
SpanBERT (large)	61.0	38.0	9.0	60.0

Table 13: Performance of modeling approaches when evaluated on bAbI QA #1, bAbI QA #2, bAbI QA #3 and bAbI QA #4.

	bAbI QA #5	bAbI QA #11	bAbI QA #12	bAbI QA #13
RaSoR	98.0	100.00	100.0	100.0
BiDAF	95.0	78.00	100.0	95.0
DocumentReader	96.0	100.00	100.0	100.0
DocumentReader (no external features)	97.0	100.00	100.0	100.0
BiDAF++	96.0	100.00	100.0	95.0
MnemonicReader	95.0	100.00	100.0	95.0
MnemonicReader (no external features)	95.0	100.00	100.0	100.0
QANet	95.0	100.00	100.0	95.0
FusionNet	98.0	100.00	100.0	100.0
FusionNet (no external features)	98.0	100.00	100.0	100.0
BERT (base, uncased)	95.0	100.00	100.0	97.0
BERT (large, uncased)	95.0	100.00	100.0	100.0
BERT (large, uncased, whole-word masking)	96.0	100.00	100.0	100.0
ALBERT (base, V1)	95.0	100.00	100.0	100.0
ALBERT (xxlarge, V1)	99.0	100.00	100.0	100.0
RoBERTa (base)	95.0	98.99	89.0	95.0
RoBERTa (large)	98.0	100.00	100.0	95.0
ELECTRA (base)	95.0	100.00	100.0	97.0
SpanBERT (base)	36.0	74.75	75.0	95.0
SpanBERT (large)	43.0	81.82	77.0	95.0

Table 14: Performance of modeling approaches when evaluated on bAbI QA #5, bAbI QA #11, bAbI QA #12 and bAbI QA #13.

Figure 10 shows how well the bAbI tasks (9000) training examples concur with SQuAD.

Table 16 and Table 17 and Table 18 show the performance of each modeling approach on each existing of the bAbI tasks (9000 training examples).

	bAbI QA #14	bAbI QA #15	bAbI QA #16
RaSoR	97.0	73.00	64.0
BiDAF	95.0	66.00	61.0
DocumentReader	96.0	68.00	63.0
DocumentReader (no external features)	99.0	68.00	64.0
BiDAF++	92.0	65.00	61.0
MnemonicReader	99.0	63.00	65.0
MnemonicReader (no external features)	99.0	67.00	65.0
QANet	62.0	64.00	58.0
FusionNet	100.0	69.00	64.0
FusionNet (no external features)	99.0	100.00	64.0
BERT (base, uncased)	84.0	60.56	50.0
BERT (large, uncased)	88.0	56.34	52.0
BERT (large, uncased, whole-word masking)	96.0	100.00	62.0
ALBERT (base, V1)	78.0	60.56	80.0
ALBERT (xxlarge, V1)	100.0	100.00	100.0
RoBERTa (base)	81.0	61.97	47.0
RoBERTa (large)	77.0	100.00	44.0
ELECTRA (base)	87.0	100.00	47.0
SpanBERT (base)	37.0	46.48	36.0
SpanBERT (large)	37.0	59.15	49.0

Table 15: Performance of modeling approaches when evaluated on bAbI QA #14, bAbI QA #15 and bAbI QA #16.

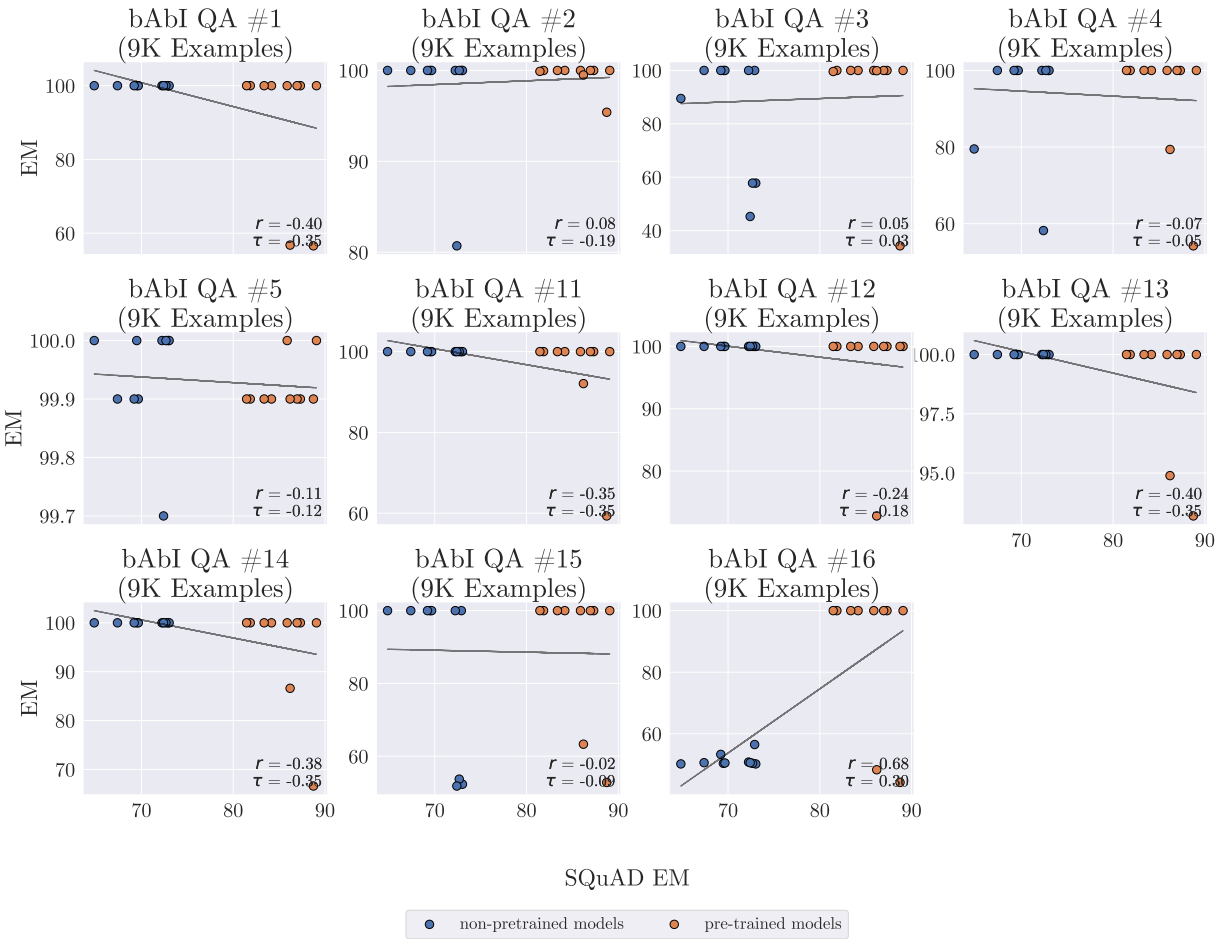

Figure 10: Many modeling approaches perform perfectly on bAbI tasks when training on 9,000 examples, limiting their ability to recapitulate historical modeling progress on SQuAD.

	bAbI QA #1 (9K)	bAbI QA #2 (9K)	bAbI QA #3 (9K)	bAbI QA #4 (9K)
RaSoR	100.00	100.0	89.5	79.50
BiDAF	100.00	100.0	100.0	100.00
DocumentReader	100.00	100.0	100.0	100.00
DocumentReader (no external features)	100.00	100.0	100.0	100.00
BiDAF++	100.00	100.0	100.0	100.00
MnemonicReader	100.00	100.0	57.8	100.00
MnemonicReader (no external features)	100.00	100.0	57.8	100.00
QANet	100.00	80.7	45.3	58.20
FusionNet	100.00	100.0	100.0	100.00
FusionNet (no external features)	100.00	100.0	100.0	100.00
BERT (base, uncased)	100.00	99.9	99.6	100.00
BERT (large, uncased)	100.00	100.0	100.0	100.00
BERT (large, uncased, whole-word masking)	100.00	100.0	100.0	100.00
ALBERT (base, V1)	100.00	100.0	100.0	100.00
ALBERT (xxlarge, V1)	100.00	100.0	100.0	100.00
RoBERTa (base)	100.00	100.0	100.0	100.00
RoBERTa (large)	100.00	100.0	100.0	100.00
ELECTRA (base)	100.00	100.0	100.0	100.00
SpanBERT (base)	56.77	99.5	99.9	79.37
SpanBERT (large)	56.57	95.4	34.3	54.21

Table 16: Performance of modeling approaches when evaluated on bAbI QA #1 (9K Examples), bAbI QA #2 (9K Examples), bAbI QA #3 (9K Examples) and bAbI QA #4 (9K Examples).

	bAbI QA #5 (9K)	bAbI QA #11 (9K)	bAbI QA #12 (9K)	bAbI QA #13 (9K)
RaSoR	100.0	100.00	100.0	100.00
BiDAF	99.9	100.00	100.0	100.00
DocumentReader	99.9	100.00	100.0	100.00
DocumentReader (no external features)	99.9	100.00	100.0	100.00
BiDAF++	100.0	100.00	100.0	100.00
MnemonicReader	100.0	100.00	100.0	100.00
MnemonicReader (no external features)	100.0	100.00	100.0	100.00
QANet	99.7	100.00	100.0	100.00
FusionNet	100.0	100.00	100.0	100.00
FusionNet (no external features)	100.0	100.00	100.0	100.00
BERT (base, uncased)	99.9	100.00	100.0	100.00
BERT (large, uncased)	99.9	100.00	100.0	100.00
BERT (large, uncased, whole-word masking)	99.9	100.00	100.0	100.00
ALBERT (base, V1)	99.9	100.00	100.0	100.00
ALBERT (xxlarge, V1)	100.0	100.00	100.0	100.00
RoBERTa (base)	99.9	100.00	100.0	100.00
RoBERTa (large)	99.9	100.00	100.0	100.00
ELECTRA (base)	100.0	100.00	100.0	100.00
SpanBERT (base)	99.9	92.08	72.8	94.89
SpanBERT (large)	99.9	59.32	100.0	93.19

Table 17: Performance of modeling approaches when evaluated on bAbI QA #5 (9K Examples), bAbI QA #11 (9K Examples), bAbI QA #12 (9K Examples) and bAbI QA #13 (9K Examples).

	bAbI QA #14 (9K)	bAbI QA #15 (9K)	bAbI QA #16 (9K)
RaSoR	100.0	100.00	50.2
BiDAF	100.0	100.00	50.6
DocumentReader	100.0	100.00	50.5
DocumentReader (no external features)	100.0	100.00	53.3
BiDAF++	100.0	100.00	50.4
MnemonicReader	100.0	52.30	50.2
MnemonicReader (no external features)	100.0	53.70	50.4
QANet	100.0	51.80	50.6
FusionNet	100.0	100.00	56.5
FusionNet (no external features)	100.0	100.00	50.8
BERT (base, uncased)	100.0	100.00	100.0
BERT (large, uncased)	100.0	100.00	100.0
BERT (large, uncased, whole-word masking)	100.0	100.00	100.0
ALBERT (base, V1)	100.0	100.00	100.0
ALBERT (xxlarge, V1)	100.0	100.00	100.0
RoBERTa (base)	100.0	100.00	100.0
RoBERTa (large)	100.0	100.00	100.0
ELECTRA (base)	100.0	100.00	100.0
SpanBERT (base)	86.6	63.30	48.3
SpanBERT (large)	66.6	52.78	44.2

Table 18: Performance of modeling approaches when evaluated on bAbI QA #14 (9K Examples), bAbI QA #15 (9K Examples) and bAbI QA #16 (9K Examples).

E FuzzySyntheticQA Construction Details

Figure 11 provides an overview of the construction of FuzzySyntheticQA.

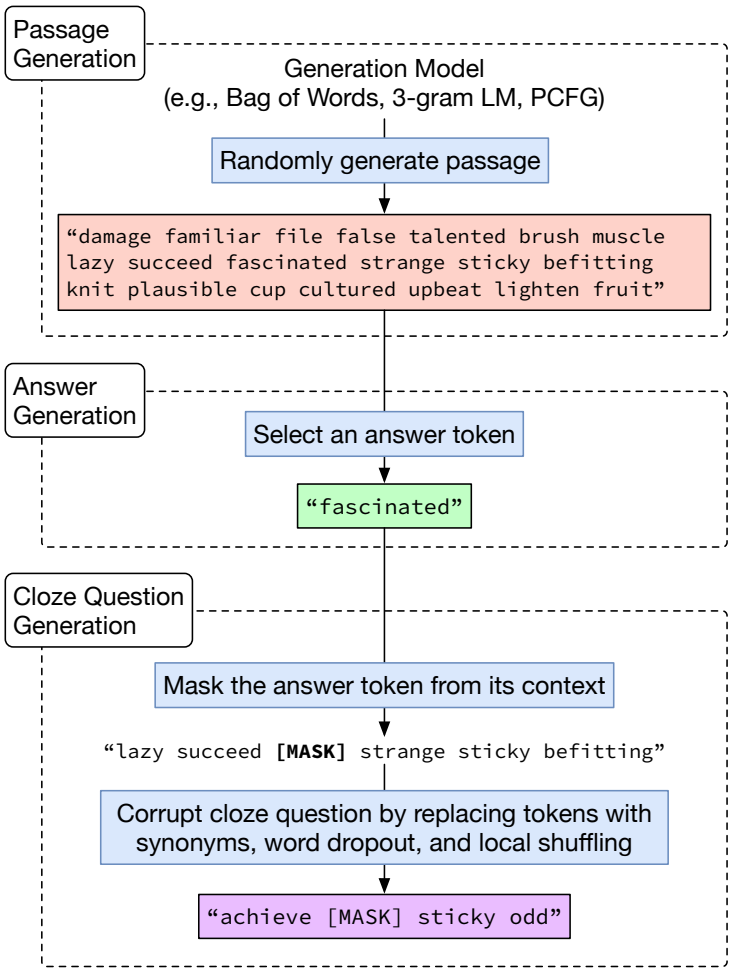

Figure 11: Constructing a FuzzySyntheticQA example by generating a passage, answer, and cloze question.

To efficiently replace tokens with related tokens, we consider each token's 100 *approximate* nearest neighbors as replacement candidates. In particular, we use Annoy (Bernhardsson and the Annoy development team, 2020) to perform the approximate nearest neighboor look-ups. Similarities are derived from the Euclidean distance of normalized vectors between two tokens.

F Full Results on FuzzySyntheticQA

Figure 12 shows that changing the passage generation method in FuzzySyntheticQA has a minimal effect on concurrence. We experiment with generating passages from a 3-gram language model, a probabilistic context-free grammar, a large neural language model (GPT-2 1.5B; Radford et al., 2019), and by taking real Wikipedia paragraphs.

The 3-gram language model is trained with maximum likelihood estimation on WikiText-103 (Merity et al., 2017). The PCFG is trained with maximum likelihood estimation on the Penn Treebank (Marcus et al., 1993). Lastly, we take GPT-2 1.5B generations from the officially-released output samples (github.com/openai/gpt-2-output-dataset; generated with top-k truncated sampling with k = 40).

Table 19 and Table 20 show the performance of each modeling approach on each of our constructed synthetic fuzzy pattern-matching benchmarks.

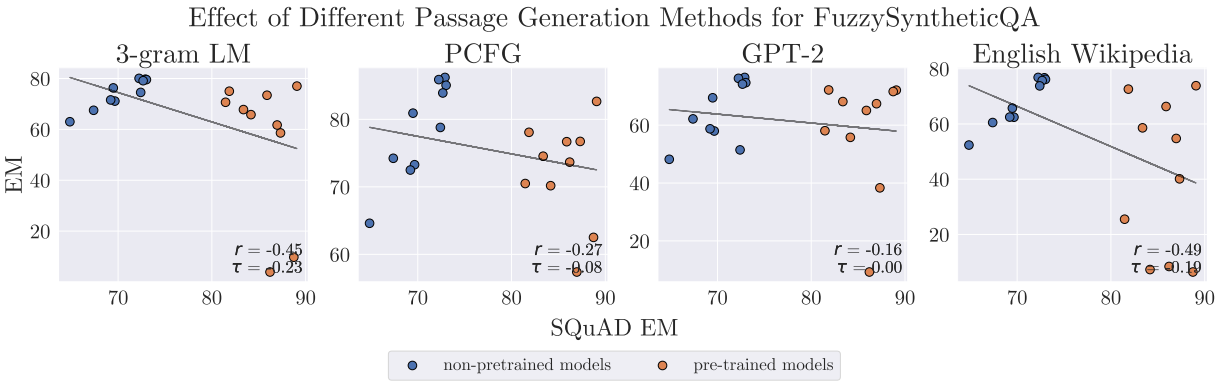

Figure 12: Even with progressively more natural passages, FuzzySyntheticQA continues to have low overall concurrence with SQuAD—this low concurrence is not trivially caused by the lack of natural passages, and simply making our passages more closely resemble natural language will not yield high concurrence.

	Synthetic Fuzzy Pattern-Matching	3-gram LM Synthetic Fuzzy Pattern-Matching	PCFG Synthetic Fuzzy Pattern-Matching
RaSoR	37.01	63.00	64.60
BiDAF	38.62	67.50	74.23
DocumentReader	49.32	71.11	73.28
DocumentReader (no external features)	49.24	71.57	72.49
BiDAF++	56.89	76.30	80.92
MnemonicReader	61.50	79.56	85.05
MnemonicReader (no external features)	61.24	79.13	83.91
QANet	59.60	74.53	78.80
FusionNet	64.71	79.72	86.21
FusionNet (no external features)	63.80	80.05	85.89
BERT (base, uncased)	4.51	70.65	70.49
BERT (large, uncased)	40.11	65.79	70.17
BERT (large, uncased, whole-word masking)	0.70	58.60	76.73
ALBERT (base, V1)	44.28	75.00	78.08
ALBERT (xxlarge, V1)	53.79	77.01	82.66
RoBERTa (base)	44.92	67.78	74.54
RoBERTa (large)	0.49	61.71	57.38
ELECTRA (base)	44.85	73.42	76.69
SpanBERT (base)	0.74	3.92	73.66
SpanBERT (large)	0.40	9.74	62.51

Table 19: Performance of modeling approaches when evaluated on Synthetic Fuzzy Pattern-Matching, 3-gram LM Synthetic Fuzzy Pattern-Matching and PCFG Synthetic Fuzzy Pattern-Matching.

	GPT-2 Synthetic Fuzzy Pattern-Matching	English Wikipedia Synthetic Fuzzy Pattern-Matching
RaSoR	48.20	52.37
BiDAF	62.16	60.52
DocumentReader	57.97	62.45
DocumentReader (no external features)	58.73	62.50
BiDAF++	69.45	65.74
MnemonicReader	74.67	76.15
MnemonicReader (no external features)	74.18	75.71
QANet	51.45	73.79
FusionNet	76.48	76.73
FusionNet (no external features)	76.17	76.85
BERT (base, uncased)	58.07	25.52
BERT (large, uncased)	55.78	7.29
BERT (large, uncased, whole-word masking)	38.34	40.13
ALBERT (base, V1)	72.16	72.62
ALBERT (xxlarge, V1)	72.09	73.86
RoBERTa (base)	68.14	58.60
RoBERTa (large)	67.41	54.76
ELECTRA (base)	65.07	66.33
SpanBERT (base)	9.26	8.40
SpanBERT (large)	71.61	6.40

Table 20: Performance of modeling approaches when evaluated on GPT-2 Synthetic Fuzzy Pattern-Matching and English Wikipedia Synthetic Fuzzy Pattern-Matching.

G WikidataSyntheticQA Construction Details

Figure 13 summarizes the data generation procedure for WikidataSyntheticQA.

Inverses of Properties. Some of our generated questions use the inverse relationships between two properties. To obtain the inverse relationship for a given property, we first retrieve its list of property constraints by using Wikidata property P2302 (property constraint). If Q21510855 (inverse constraint) is present, we then retrieve the corresponding property of this inverse relationship. If the inverse constraint is not present, we check the corresponding property of P7087 (inverse label item), which outputs the item with a label of the inverse relationship of the property.

Entity Hyponyms. Some of our generated questions replace entities with their hyponyms. To obtain the hyponyms for a given entity, we retrieve any object entities of the P31 (instance of) and P279 (subclass of) properties.

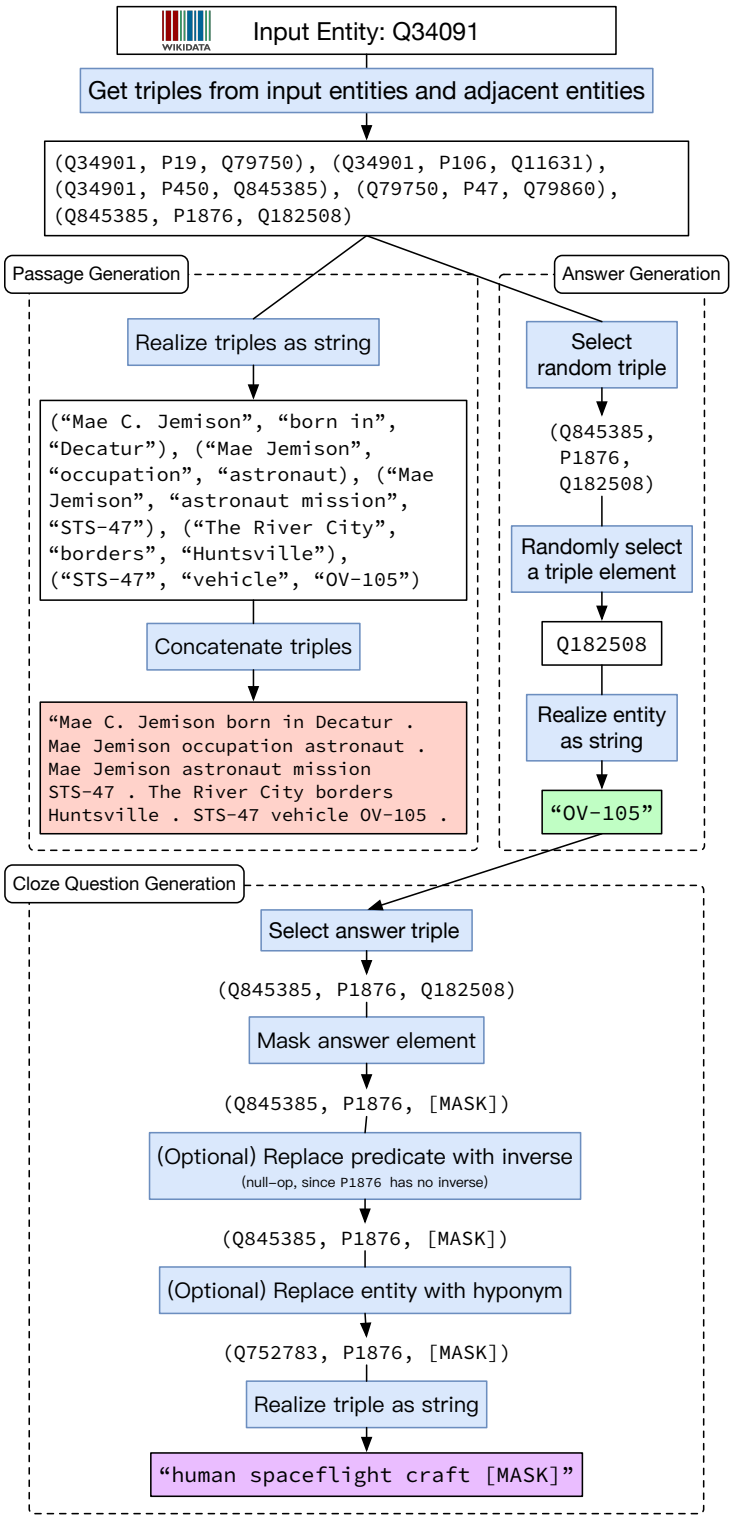

Figure 13: Constructing a WikidataSyntheticQA example by generating a passage, answer, and cloze question.

H Full Results on WikidataSyntheticQA

Table 21 shows the performance of each modeling approach on WikidataSyntheticQA.

	Synthetic Wikidata
RaSoR	63.67
BiDAF	68.69
DocumentReader	67.66
DocumentReader (no external features)	68.03
BiDAF++	70.43
MnemonicReader	75.04
MnemonicReader (no external features)	74.31
QANet	73.12
FusionNet	74.52
FusionNet (no external features)	73.90
BERT (base, uncased)	73.68
BERT (large, uncased)	78.01
BERT (large, uncased, whole-word masking)	81.56
ALBERT (base, V1)	77.23
ALBERT (xxlarge, V1)	86.29
RoBERTa (base)	77.75
RoBERTa (large)	82.79
ELECTRA (base)	76.86
SpanBERT (base)	78.50
SpanBERT (large)	84.26

Table 21: Performance of modeling approaches when evaluated on Synthetic Wikidata.

I Full Results on Subsampled SQuAD

Table 22 and Table 23 show the performance of each modeling approach on subsamples of the SQuAD benchmark.

	SQuAD 1.1		
	All	1K Examples	10K Examples
RaSoR	64.86	15.52	49.44
BiDAF	67.39	7.96	48.54
DocumentReader	69.66	34.66	56.42
DocumentReader (no external features)	69.21	30.69	54.82
BiDAF++	69.49	18.62	57.48
MnemonicReader	73.02	30.67	58.91
MnemonicReader (no external features)	72.67	29.46	57.79
QANet	72.41	7.18	48.15
FusionNet	72.90	37.52	59.97
FusionNet (no external features)	72.24	35.55	58.69
BERT (base, uncased)	81.46	31.80	70.34
BERT (large, uncased)	84.17	49.08	75.47
BERT (large, uncased, whole-word masking)	87.32	69.19	81.78
ALBERT (base, V1)	81.86	57.57	74.55
ALBERT (xxlarge, V1)	89.07	76.36	86.19
RoBERTa (base)	83.37	55.01	77.30
RoBERTa (large)	86.96	62.64	82.56
ELECTRA (base)	85.88	62.05	78.31
SpanBERT (base)	86.20	65.80	80.72
SpanBERT (large)	88.74	75.00	85.06

Table 22: Performance of modeling approaches when evaluated on SQuAD, SQuAD (1K Examples) and SQuAD (10K Examples).

	SQuAD 1.1		
	20K Examples	40K Examples	60K Examples
RaSoR	55.13	60.37	62.95
BiDAF	57.29	62.35	65.25
DocumentReader	61.84	65.45	68.27
DocumentReader (no external features)	59.66	64.47	67.09
BiDAF++	62.25	66.42	68.62
MnemonicReader	64.74	69.09	70.86
MnemonicReader (no external features)	63.71	68.65	70.32
QANet	61.02	66.55	69.74
FusionNet	64.74	69.14	70.98
FusionNet (no external features)	63.28	67.98	69.93
BERT (base, uncased)	74.84	78.24	80.05
BERT (large, uncased)	79.27	81.83	83.25
BERT (large, uncased, whole-word masking)	84.47	85.78	86.75
ALBERT (base, V1)	77.05	79.95	81.02
ALBERT (xxlarge, V1)	86.91	88.02	88.63
RoBERTa (base)	79.56	81.62	82.37
RoBERTa (large)	84.26	86.37	87.18
ELECTRA (base)	81.75	83.95	85.01
SpanBERT (base)	82.54	84.17	85.39
SpanBERT (large)	86.21	87.33	87.82

Table 23: Performance of modeling approaches when evaluated on SQuAD (20K Examples), SQUAD (40K Examples) and SQuAD (60K Examples).

ACL 2023 Responsible NLP Checklist

A For every submission:

☑ A1. Did you describe the limitations of your work?
Yes, at the very end of the paper in an unmarked section.

☐ A2. Did you discuss any potential risks of your work?
Not applicable. Left blank.

☑ A3. Do the abstract and introduction summarize the paper's main claims?
Section 1

☒ A4. Have you used AI writing assistants when working on this paper?
Left blank.

B ☑ Did you use or create scientific artifacts?

Yes, we used code published by prior researchers for training and evaluating QA models they had proposed. We also used existing datasets. See section 2 and 3.

☑ B1. Did you cite the creators of artifacts you used?
Yes, see section 2 and 3.

☐ B2. Did you discuss the license or terms for use and / or distribution of any artifacts?
Not applicable. Left blank.

☐ B3. Did you discuss if your use of existing artifact(s) was consistent with their intended use, provided that it was specified? For the artifacts you create, do you specify intended use and whether that is compatible with the original access conditions (in particular, derivatives of data accessed for research purposes should not be used outside of research contexts)?
Not applicable. Left blank.

☐ B4. Did you discuss the steps taken to check whether the data that was collected / used contains any information that names or uniquely identifies individual people or offensive content, and the steps taken to protect / anonymize it?
Not applicable. Left blank.

☐ B5. Did you provide documentation of the artifacts, e.g., coverage of domains, languages, and linguistic phenomena, demographic groups represented, etc.?
Not applicable. Left blank.

☐ B6. Did you report relevant statistics like the number of examples, details of train / test / dev splits, etc. for the data that you used / created? Even for commonly-used benchmark datasets, include the number of examples in train / validation / test splits, as these provide necessary context for a reader to understand experimental results. For example, small differences in accuracy on large test sets may be significant, while on small test sets they may not be.
Not applicable. Left blank.

C ☑ Did you run computational experiments?

Yes, sections 3 and 4.

☑ C1. Did you report the number of parameters in the models used, the total computational budget (e.g., GPU hours), and computing infrastructure used?
Yes, Appendix A.

The Responsible NLP Checklist used at ACL 2023 is adopted from NAACL 2022, with the addition of a question on AI writing assistance.

☑ C2. Did you discuss the experimental setup, including hyperparameter search and best-found hyperparameter values?
Yes, Appendix A.

☑ C3. Did you report descriptive statistics about your results (e.g., error bars around results, summary statistics from sets of experiments), and is it transparent whether you are reporting the max, mean, etc. or just a single run?
Yes, Appendix A.

☑ C4. If you used existing packages (e.g., for preprocessing, for normalization, or for evaluation), did you report the implementation, model, and parameter settings used (e.g., NLTK, Spacy, ROUGE, etc.)?
Yes, Appendix A.

D ☒ Did you use human annotators (e.g., crowdworkers) or research with human participants?

Left blank.

☐ D1. Did you report the full text of instructions given to participants, including e.g., screenshots, disclaimers of any risks to participants or annotators, etc.?
No response.

☐ D2. Did you report information about how you recruited (e.g., crowdsourcing platform, students) and paid participants, and discuss if such payment is adequate given the participants' demographic (e.g., country of residence)?
No response.

☐ D3. Did you discuss whether and how consent was obtained from people whose data you're using/curating? For example, if you collected data via crowdsourcing, did your instructions to crowdworkers explain how the data would be used?
No response.

☐ D4. Was the data collection protocol approved (or determined exempt) by an ethics review board?
No response.

☐ D5. Did you report the basic demographic and geographic characteristics of the annotator population that is the source of the data?
No response.