Looking Beyond the Surface: A Challenge Set for Reading Comprehension over Multiple Sentences

Daniel Khashabi¹, Snigdha Chaturvedi², Michael Roth³, Shyam Upadhyay¹, Dan Roth¹
¹University of Pennsylvania, ²University of California, Santa Cruz, ³Saarland University
{danielkh,shyamupa,danroth}@cis.upenn.edu, snigdha@ucsc.edu, mroth@coli.uni-sb.de

Abstract
We present a reading comprehension challenge in which questions can only be answered by taking into account information from multiple sentences. We solicit and verify questions and answers for this challenge through a 4-step crowdsourcing experiment. Our challenge dataset contains ~6k questions for +800 paragraphs across 7 different domains (elementary school science, news, travel guides, fiction stories, etc) bringing in linguistic diversity to the texts and to the questions wordings. On a subset of our dataset, we found human solvers to achieve an F1-score of 86.4%. We analyze a range of baselines, including a recent state-of-art reading comprehension system, and demonstrate the difficulty of this challenge, despite a high human performance. The dataset is the first to study multi-sentence inference at scale, with an open-ended set of question types that requires reasoning skills.

1 Introduction
Machine Comprehension of natural language text is a fundamental challenge in AI and it has received significant attention throughout the history of AI (Greene, 1959; McCarthy, 1976; Reiter, 1976; Winograd, 1980). In particular, in natural language processing (NLP) it has been studied under various settings, such as multiple-choice Question-Answering (QA) (Green Jr. et al., 1961), Reading Comprehension (RC) (Hirschman et al., 1999), Recognizing Textual Entailment (RTE) (Dagan et al., 2013) etc. The area has seen rapidly increasing interest, thanks to the existence of sizable datasets and standard benchmarks. CNN/Daily Mail (Hermann et al., 2015), SQuAD (Rajpurkar et al., 2016) and NewsQA (Trischler et al., 2016) to name a few, are some of the datasets that were released recently with the goal of facilitating research in machine comprehension. Despite all the excitement fueled by that large data sets and the ability to directly train statistical learning models, current QA systems do not have capabilities comparable to elementary school or younger children (Clark and Etzioni, 2016). For many of these datasets, researchers point out that models neither need to 'comprehend' in order to correctly predict an answer, nor do they learn to 'reason' in a way that generalizes across datasets. For example, Khashabi et al. (2016) showed that adversarial perturbation in candidate answers results in a significant drop in performance of a few state-of-art science QA systems. Similarly, Jia and Liang (2017) show that adding an adversarially selected sentence to the instances in the SQuAD datasets drastically reduces the performance of many of the existing baselines. Chen et al. (2016) show that in the CNN/Daily Mail datasets, “the required reasoning and inference level . . . is quite simple” and that a relatively simple algorithm can get almost close to the upper-bound. We believe that one key reason that simple algorithms can deal with the existing large datasets but, nevertheless, fail at generalization, is that the datasets do not actually require a deep understanding.

We propose to address this shortcoming by developing a reading comprehension challenge in which answering each of the questions requires reasoning over multiple sentences.

There is evidence that answering ‘single-sentence questions’, i.e. questions that can be answered from a single sentence of the given paragraph, is easier than answering multi-sentence questions’, which require multiple sentences to answer a given question. For example, Richardson et al. (2013) released a reading comprehension dataset that contained both single-sentence and multi-sentence questions; models proposed for this task yielded considerably better performance on the single-sentence questions than on the multi-
sentence questions (according to Narasimhan and Barzilay (2015) accuracy of about 83% and 60% on these two types of questions, respectively).

There could be multiple reasons for this. First, multi-sentence reasoning seems to be inherently a difficult task. Research has shown that while complete-sentence construction emerges as early as first grade for many children, their ability to integrate sentences emerges only in fourth grade (Berninger et al., 2011). Answering multi-sentence questions might be more challenging for an automated system because it involves more than just processing individual sentences but rather combining linguistic, semantic and background knowledge across sentences—a computational challenge itself. Despite these challenges, multi-sentence questions can be answered by humans and hence present an interesting yet reasonable goal for AI systems (Davis, 2014).

In this work, we propose a multi-sentence QA challenge in which questions can be answered only using information from multiple sentences. Specifically, we present MultiRC (Multi-Sentence Reading Comprehension)—a dataset of short paragraphs and multi-sentence questions that can be answered from the content of the paragraph. Each question is associated with several choices for answer-options, out of which one or more correctly answer the question. Figure 1 shows two examples from our dataset. Each instance consists of a multi-sentence paragraph, a question, and answer-options. All instances were constructed such that it is not possible to answer a question correctly without gathering information from multiple sentences. Due to space constraints, the figure shows only the relevant sentences from the original paragraph. The entire corpus consists of 871 paragraphs and about ~60k multi-sentence questions.

The goal of this dataset is to encourage the research community to explore approaches that can do more than sophisticated lexical-level matching. To accomplish this, we designed the dataset with three key challenges in mind. (i) The number of correct answer-options for each question is not pre-specified. This removes the over-reliance of current approaches on answer-options and forces them to decide on the correctness of each candidate answer independently of others. In other words, unlike previous work, the task here is not to simply identify the best answer-option, but to evaluate the correctness of each answer-option individually. For example, the first question in Figure 1 can be answered by combining information from sentences 3, 5, 10, 13 and 15. It requires not only understanding that the stalker’s name is Timothy but also that he is the man who Mary had hit. (ii) The correct answer(s) is not required to be a span in the text. For example, the correct answer, A, of the second question in Figure 1 is not present in the paragraph verbatim. It is instead a combination of two spans from 2 sentences: 12 and 13. Such answer-options force models to process and understand not only the paragraph and the question but also the answer-options. (iii) The paragraphs in our dataset have diverse provenance by being extracted from 7 different domains such as news, fiction, historical text etc., and hence are expected to be more diverse in their contents as compared to single-domain datasets. We also expect this to lead to diversity in the types of questions that can be constructed from the passage.

Overall, we introduce a reading comprehension
dataset that significantly differs from most other datasets available today in the following ways:

- ∼6k high-quality multiple-choice RC questions that are generated (and manually verified via crowdsourcing) to require integrating information from multiple sentences.
- The questions are not constrained to have a single correct answer, generalizing existing paradigms for representing answer-options.
- Our dataset is constructed using 7 different sources, allowing more diversity in content, style, and possible question types.
- We show a significant performance gap between current solvers and human performance, indicating an opportunity for developing sophisticated reasoning systems.

2 Relevant Work

Automated reasoning is arguably one of the major problems in contemporary AI research. Brachman et al. (2005) suggest challenges for developing AI program that can pass the SAT exams. In similar spirit Clark and Etzioni (2016) advocate elementary-school tests as a new test for AI. Davis (2014) proposes hand-construction of multiple-choice challenge sets that are easy for children but difficult for computers. Despite Davis’ claim on simplicity of his target questions, it is not clear how easy it is to generate such questions, as he doesn’t provide any reasonably-sized dataset matching his proposal. Weston et al. (2015) present a relatively small dataset of 10 reasoning categories, and propose to build a system that uses a world model and a linguistic model. The fundamental limitation of the dataset is that it is generated according to a restricted set of reasoning categories, which possibly limits the complexity and diversity of questions.

Some other recent datasets proposed for machine comprehension also pay attention to type of questions and reasoning required. For example, RACE (Lai et al., 2017) attempts to incorporate different types of reasoning phenomena, and MCTest (Richardson et al., 2013) attempted to contain at least 50% multi-sentence reasoning questions. However, since the crowdsourced workers who created the dataset were only encouraged, and not required, to write such questions, it is not clear how many of these questions actually require multi-sentence reasoning (see Sec. 3.5). Similarly, only about 25% of question in the RACE dataset require multi-sentence reasoning as reported in their paper. Remedia (Hirschman et al., 1999) also contains 5 different types of questions (based on question words) but is a much smaller dataset. Other datasets which do not deliberately attempt to include multi-sentence reasoning, like SQuAD (Rajpurkar et al., 2016) and the CNN/Daily Mail dataset (Hermann et al., 2015), suffer from even lower percentage of such questions (12% and 2% respectively (Lai et al., 2017)). There are several other corpora which do not guarantee specific reasoning types, including MS MARCO (Nguyen et al., 2016), WikiQA (Yang et al., 2015), and TriviaQA (Joshi et al., 2017).

The complexity of reasoning required for a reading comprehension dataset would depend on several factors such as the source of questions or paragraphs; the way they are generated; and the order in which they are generated (i.e. questions from paragraphs, or the reverse). Specifically, paragraphs’ source could influence the complexity and diversity of the language of the paragraphs and questions, and hence the required level of reasoning capabilities. Unlike most current datasets which rely on only one or two sources for their paragraphs (e.g. CNN/Daily Mail and SQuAD rely only on news and Wikipedia articles respectively) our dataset uses 7 different domains.

Another factor that distinguishes our dataset from previously proposed corpora is the way answers are represented. Several datasets represent answers as multiple-choices with a single correct answer. While multiple-choice questions are easy to grade, coming up with non-trivial correct and incorrect answers can be challenging. Also, assuming exactly one correct answer (e.g., as in MCTest and RACE) inadvertently changes the task from choosing the correct answer to choosing the most likely answer. Other datasets (e.g MS-MARCO and SQuAD) represent answers as a contiguous substring within the passage. This assumption of the answer being a span of the paragraph, limits the questions to those whose answer is contained verbatim in the paragraph. Unfortunately, it rules out more complicated questions whose answers are only implied by the text and hence require a deeper understanding. Because of these limitations, we designed our dataset to use multiple-choice representations, but without specifying the number of correct answers for each question.
3 Construction of MultiRC

In this section we describe our principles and methodology of dataset collection. This includes automatically collecting paragraphs, composing questions and answer-options through crowd-sourcing platform, and manually curating the collected data. We also summarize a pilot study that helped us design this process, and end with a summary of statistics of the collected corpus.

3.1 Principles of design

Questions and answers in our dataset are designed based on the following key principles:

**Multi-sentenceness.** Questions in our challenge require models to use information from multiple sentences of a paragraph. This is ensured through explicit validation. We exclude any question that can be answered based on a single sentence from a paragraph.

**Open-endedness.** Our dataset is not restricted to questions whose answer can be found verbatim in a paragraph. Instead, we provide a set of hand-crafted answer-options for each question. Notably, they can represent information that is not explicitly stated in the text but is only inferable from it (e.g. implied counts, sentiments, and relationships).

**Answers to be judged independently.** The total number of answer options per question is variable in our data and we explicitly allow multiple correct and incorrect answer options (e.g. 2 correct and 1 incorrect options). As a consequence, correct answers cannot be guessed solely by a process of elimination or by simply choosing the best candidates out of the given options.

 Through these principles, we encourage users to explicitly model the semantics of text beyond individual words and sentences, to incorporate extra-linguistic reasoning mechanisms, and to handle answer options independently of one another.

**Variability.** We encourage variability on different levels. Our dataset is based on paragraphs from multiple domains, leading to linguistically diverse questions and answers. Also, we do not impose any restrictions on the questions, to encourage different forms of reasoning.

3.2 Sources of documents

The paragraphs used in our dataset are extracted from various sources. Here is the complete list of the text types and sources used in our dataset, and the number of paragraphs extracted from each category (indicated in square brackets on the right):

1. News: [121]
   - CNN (Hermann et al., 2015)
   - WSJ (Ide et al., 2008)
   - NYT (Ide et al., 2008)
2. Wikipedia articles [92]
3. Articles on society, law and justice (Ide and Suderman, 2006) [91]
4. Articles on history and anthropology (Ide et al., 2008) [65]
5. Elementary school science textbooks [153]
6. 9/11 reports (Ide and Suderman, 2006) [72]
7. Fiction: [277]
   - Stories from the Gutenberg project
   - Children stories from MCTest (Richardson et al., 2013)
   - Movie plots from CMU Movie Summary corpus (Bamman et al., 2013)

From each of the above-mentioned sources we extracted paragraphs that had enough content. To ensure this we followed a 3-step process. In the first step we selected top few sentences from paragraphs such that they contained k-1.5k characters. To ensure coherence, all sentences were contiguous and extracted from the same paragraph. In this process we also discarded paragraphs that seemed to deviate too much from third person narrative style. For example, while processing Gutenberg corpus we considered files that had at least 5k lines because we found that most of them were short poetic texts. In the second step, we annotated (Khashabi et al., 2018b) the paragraphs and automatically filtered texts using conditions such as the average number of words per sentence; number of named entities; number of discourse connectives in the paragraph. These were designed by the authors of this paper after reviewing a small sample of paragraphs. A complete set of conditions is listed in Table 1. Finally in the last step, we manually verified each paragraph and filtered out the ones that had formatting issues or other concerns that seemed to compromise their usability.

[https://www.ck12.org]
pairing it with each of the sentences necessary for the previous step, we create question-sentence pairs by verification can only be answered using more than one question. In a second step, we verify that each question has requirements. Step 2: Verifying multi-sentenceness of questions. Regarding typos and unusual wordings.

Step 2: Generating answer-options. In this step, we collect answer-options that will be shown with each question. Specifically, for each verified question from the previous steps, we ask 3 turkers to write as many correct and incorrect answer options as they can think of. In order to not curb creativity, we do not place a restriction on the number of options they have to write. We explicitly ask turkers to design difficult and non-trivial incorrect answer-options (e.g. if the question is about a person, a non-trivial incorrect answer-option would be other people mentioned in the paragraph).

After this step, we perform a light clean up of the candidate answers by manually correcting minor errors (such as typos), completing incomplete sentences and rephrasing any ambiguous sentences. We further make sure there is not much repetition in the answer-options, to prevent potential exploitation of correlation between some candidate answers in order to find the correct answer. For example, we drop obviously duplicate answer-options (i.e. identical options after lower-casing, lemmatization, and removing stop-words).

Step 4: Verifying quality of the dataset. This step serves as the final quality check for both questions and the answer-options generated in the previous steps. We show each paragraph, its questions, and the corresponding answer-options to 3 turkers, and ask them to indicate if they find any errors (grammatical or otherwise), in the questions and/or answer-options. We then manually review,

<table>
<thead>
<tr>
<th>Condition</th>
<th>bound</th>
</tr>
</thead>
<tbody>
<tr>
<td>Number of sentences</td>
<td>≥ 6 &amp; ≤ 18</td>
</tr>
<tr>
<td>Number of NER(CoNLL) mentions</td>
<td>≥ 2</td>
</tr>
<tr>
<td>Avg. number of NER(CoNLL) mentions</td>
<td>≤ 0.2</td>
</tr>
<tr>
<td>Number of NER(Ontonotes) mentions</td>
<td>≥ 4</td>
</tr>
<tr>
<td>Avg. number of NER(Ontonotes) mentions</td>
<td>≤ 0.25</td>
</tr>
<tr>
<td>Avg. number of words per sentence</td>
<td>≥ 5</td>
</tr>
<tr>
<td>Number of coreference mentions</td>
<td>≥ 3</td>
</tr>
<tr>
<td>Avg. number of coreference mentions</td>
<td>≤ 0.1</td>
</tr>
<tr>
<td>Number of coreference relations</td>
<td>≥ 3</td>
</tr>
<tr>
<td>Avg. number of coreference relations</td>
<td>≤ 0.08</td>
</tr>
<tr>
<td>Number of coreference chains</td>
<td>≥ 2</td>
</tr>
<tr>
<td>Avg. number of coreference chains</td>
<td>≤ 0.1</td>
</tr>
<tr>
<td>Number of discourse markers</td>
<td>≥ 2</td>
</tr>
</tbody>
</table>

Table 1: Bounds used to select paragraphs for dataset creation.

### 3.3 Pipeline of question extraction

In this section, we delineate details of the process for collecting questions and answers. Figure 2 gives a high-level idea of the process. The first two steps deal with creating multi-sentence questions, followed by two steps for construction of candidate answers. Interested readers can find more details on set-ups of each step in Appendix I.

**Step 1: Generating questions.** The goal of the first step of our pipeline is to collect multi-sentence questions. We show each paragraph to 5 turkers and ask them to write 3-5 questions such that: (1) the question is answerable from the passage, and (2) only those questions are allowed whose answer cannot be determined from a single sentence. We clarify this point by providing example paragraphs and questions. In order to encourage turkers to write meaningful questions that fit our criteria, we additionally ask them for a correct answer and for the sentence indices required to answer the question. To ensure the grammatical quality of the questions collected in this step, we limit the turkers to the countries with English as their major language. After the acquisition of questions in this step, we filter out questions which required less than 2 or more than 4 sentences to be answered; we also run them through an automatic spell-checker\(^3\) and manually correct questions regarding typos and unusual wordings.

**Step 2: Verifying multi-sentenceness of questions.** In a second step, we verify that each question can only be answered using more than one sentence. For each question collected in the previous step, we create question-sentence pairs by pairing it with each of the sentences necessary for answering it as indicated in the previous step. For a given question-sentence pair, we then ask turkers to annotate if they could answer the question from the sentence it is paired with (binary annotation). The underlying idea of this step is that a multi-sentence question would not be answerable from a single sentence, hence turkers should not be able to give a correct answer for any of the question-sentence pair. Accordingly, we determine a question as requiring multiple sentences only if the correct answer cannot be guessed from any single question-sentence pair. We collected at least 3 annotations per pair, and to avoid sharing of information across sentences, no two pairs shown to a turker came from the same paragraph. We aggregate the above annotations for each question-answer pair and retain only those questions for which no pair was judged as answerable by a majority of turkers.

**Step 3: Generating answer-options.** In this step, we collect answer-options that will be shown with each question. Specifically, for each verified question from the previous steps, we ask 3 turkers to write as many correct and incorrect answer options as they can think of. In order to not curb creativity, we do not place a restriction on the number of options they have to write. We explicitly ask turkers to design difficult and non-trivial incorrect answer-options (e.g. if the question is about a person, a non-trivial incorrect answer-option would be other people mentioned in the paragraph).

After this step, we perform a light clean up of the candidate answers by manually correcting minor errors (such as typos), completing incomplete sentences and rephrasing any ambiguous sentences. We further make sure there is not much repetition in the answer-options, to prevent potential exploitation of correlation between some candidate answers in order to find the correct answer. For example, we drop obviously duplicate answer-options (i.e. identical options after lower-casing, lemmatization, and removing stop-words).

**Step 4: Verifying quality of the dataset.** This step serves as the final quality check for both questions and the answer-options generated in the previous steps. We show each paragraph, its questions, and the corresponding answer-options to 3 turkers, and ask them to indicate if they find any errors (grammatical or otherwise), in the questions and/or answer-options. We then manually review,
and correct if needed, all erroneous questions and answer-options. This ensures that we have meaningful questions and answer-options. In this step, we also want to verify that the correct (or incorrect) options obtained from Step 3 were indeed correct (or incorrect). For this, we additionally ask the annotators to select all correct answer-options for the question. If their annotations did not agree with the ones we had after Step 3 (e.g., if they unanimously selected an ‘incorrect’ option as the answer), we manually reviewed and corrected (if needed) the annotation.

3.4 Pilot experiments

The 4-step process described above was a result of detailed analysis and substantial refinement after two small pilot studies.

In the first pilot study, we ran a set of 10 paragraphs extracted from the CMU Movie Summary Corpus through our pipeline. Our then pipeline looked considerably different from the one described above. We found the steps that required turkers to write questions and answer-options to often have grammatical errors, possibly because a large majority of turkers were non-native speakers of English. This problem was more prominent in questions than in answer-options. Because of this, we decided to limit the task to native speakers. Also, based on the results of this pilot, we overhauled the instructions of these steps by including examples of grammatically correct—but undesirable (not multi-sentence)—questions and answer-options, in addition to several minor changes.

Thereafter, we decided to perform a manual validation of the verification steps (current Steps 2 and 4). For this, we (the authors of this paper) performed additional annotations ourselves on the data shown to turkers, and compared our results with those provided by the turkers. We found that in the verification of answer-options, our annotations were in high agreement (98%) with those obtained from mechanical turk. However, that was not the case for the verification of multi-sentence questions. We made several further changes to the first two steps. Among other things, we clarified in the instructions that turkers should not use their background knowledge when writing and verifying questions, and also included negative examples of such questions. Additionally, when turkers judged a question to be answerable using a single sentence, we decided to encourage (but not require) them to guess the answer to the question. This improved our results considerably, possibly because it forced annotators to think more carefully about what the answer might be, and whether they actually knew the answer or they just thought that they knew it (possibly because of background knowledge or because the sentence contained a lot of information relevant to the question). Guessed answers in this step were only used to verify the validity of multi-sentence questions. They were not used in the dataset or subsequent steps.

After revision, we ran a second pilot study in which we processed a set of 50 paragraphs through our updated pipeline. This second pilot confirmed that our revisions were helpful, but thanks to its larger size, also allowed us to identify a couple of borderline cases for which additional clarifications were required. Based on the results of the second pilot, we made some additional minor changes and then decided to apply the pipeline for creating the final dataset.

3.5 Verifying multi-sentenceness

While collecting our dataset, we found that, even though Step 1 instructed turkers to write multi-sentence questions, not all generated questions indeed required multi-sentence reasoning. This happened even after clarifications and revisions to the corresponding instructions, and we attribute it to honest mistakes. Therefore, we designed the subsequent verification step (Step 2).

There are other datasets which aim to include multi-sentence reasoning questions, especially MCTest. Using our verification step, we systematically verify their multi-sentenceness. For this, we conducted a small pilot study on about 60 multi-sentence questions from MCTest. As for our own verification, we created question-sentence pairs for each question and asked annotators to judge whether they can answer a question from the single sentence shown. Because we did not know
which sentences contain information relevant to a question, we created question-sentence pairs using all sentences from a paragraph. After aggregation of turker annotations, we found that about half of the questions annotated as multi-sentence could be answered from a single sentence of the paragraph. This study, though performed on a subset of the data, underscores the necessity of rigorous veriﬁcation step for multi-sentence reasoning when studying this phenomenon.

3.6 Statistics on the dataset

We now provide a brief summary of MultiRC. Overall, it contains roughly ∼ 6k multi-sentence questions collected for about +800 paragraphs. The median number of correct and total answer options for each question is 2 and 5, respectively. Additional statistics are given in Table 2.

In Step 1, we also asked annotators to identify sentences required to answer a given question. We found that answering each question required 2.4 sentences on average. Also, required sentences are often not contiguous, and the average distance between sentences is 2.4. Next, we analyze the types of questions in our dataset. Figure 4 shows the count of ﬁrst word(s) for our questions. We can see that while the popular question words (What, Who, etc.) are very common, there is a wide variety in the ﬁrst word(s) indicating a diversity in question types. About 28% of our questions require binary decisions (true/false or yes/no).

We randomly selected 60 multi-sentence questions from our corpus and asked two independent annotators to label them with the type of reasoning phenomenon required to answer them. During this process, the annotators were shown a list of common reasoning phenomena (shown below), and they had to identify one or more of the phenomena relevant to a given question. The list of phenomena shown to the annotators included the following categories: mathematical and logical reasoning, spatio-temporal reasoning, list/enumeration, coreference resolution (including implicit references, abstract pronouns, event coreference, etc.), causal relations, paraphrases and contrasts (including lexical relations such as synonyms, antonyms), commonsense knowledge, and ‘other’. The categories were selected after a manual inspection of a subset of questions by two of the authors. The annotation process revealed that answering questions in our corpus requires a broad variety of reasoning phenomena. The left plot in Figure 3 provides detailed results.

The ﬁgure shows that a large fraction of questions require coreference resolution, and a more careful inspection revealed that there were different types of coreference phenomena at play here. To investigate these further, we conducted a follow-up experiment in which manually annotated all questions that required coreference resolution into ﬁner categories. Speciﬁcally, each question was shown to two annotators who were asked to select one or more of the following categories: entity coreference (between two entities), event coreference (between two events), set inclusion coreference (one item is part of or included in the other) and ‘other’. Figure 3 (right) shows the results of this experiment. We can see that, as expected, entity coreference is the most common type of coreference phenomena needed in our corpus. However, a signiﬁcant number of questions also require other types of coreference resolution. We provide some examples of questions along with the required reasoning phenomena in Appendix II.

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td># of paragraphs</td>
<td>871</td>
</tr>
<tr>
<td># of questions</td>
<td>9,872</td>
</tr>
<tr>
<td># of multi-sentence questions</td>
<td>5,825</td>
</tr>
<tr>
<td>avg # of candidates (per question)</td>
<td>5.44</td>
</tr>
<tr>
<td>avg # of correct answers (per question)</td>
<td>2.58</td>
</tr>
<tr>
<td>avg paragraph length (in sentences)</td>
<td>14.3 (4.1)</td>
</tr>
<tr>
<td>avg paragraph length (in tokens)</td>
<td>263.1 (92.4)</td>
</tr>
<tr>
<td>avg question length (in tokens)</td>
<td>10.9 (4.8)</td>
</tr>
<tr>
<td>avg answer length (in tokens)</td>
<td>4.7 (5.5)</td>
</tr>
<tr>
<td>% of yes/no/true/false questions</td>
<td>27.57%</td>
</tr>
<tr>
<td>avg # of sent. used for questions</td>
<td>2.37 (0.63)</td>
</tr>
<tr>
<td>avg distance between sent. used</td>
<td>2.4 (2.58)</td>
</tr>
<tr>
<td>% of correct answers verbatim in paragraph</td>
<td>54.96%</td>
</tr>
<tr>
<td>% of incorrect answers verbatim in paragraph</td>
<td>25.84%</td>
</tr>
</tbody>
</table>

Table 2: Various statistics of our dataset. Figures in parentheses represent standard deviation.

4 Analysis

In this section, we provide a quantitative analysis of several baselines for our challenge.

Evaluation Metrics. We deﬁne precision and recall for a question $q$ as: $\text{Pre}(q) = \frac{|A(q)\cap A(q)|}{|A(q)|}$ and $\text{Rec}(q) = \frac{|A(q)\cap A(q)|}{|A(q)|}$, where $A(q)$ and $A(q)$ are the sets of correct and selected answer-options.

---

4We will also release the 3.7k questions that did not pass Step 2. Though not multi-sentence questions, they could be a valuable resource on their own.

5The annotations were adjudicated by two authors of this paper.
We define (macro-average) $F_1_m$ as the harmonic mean of average-precision $\text{avg}_{q \in Q}(\text{Pre}(q))$ and average-recall $\text{avg}_{q \in Q}(\text{Rec}(q))$ with $Q$ as the set of all questions.

Since by design, each answer-option can be judged independently, we consider another metric, $F_1_a$, evaluating binary decisions on all the answer-options in the dataset. We define $F_1_a$ to be the harmonic mean of $\text{Pre}(Q)$ and $\text{Rec}(Q)$, with $\text{Pre}(Q) = |A(Q) \cap \hat{A}(Q)| / |A(Q)|; \ A(Q) = \bigcup_{q \in Q} A(q)$; and similar definitions for $\hat{A}(Q)$ and $\text{Rec}(Q)$.

4.1 Baselines

Human. Human performance provides us with an estimate of the best achievable results on datasets. Using mechanical turk, we ask 4 people (limited to native speakers) to solve our data. We evaluate score of each label by averaging the decision of the individuals.

Random. To get an estimate on the lower-bound we consider a random baseline, where each answer option is selected as correct with a probability of 50% (an unbiased coin toss). The numbers reported for this baseline represent the expected outcome (statistical expectation).

IR (information retrieval baseline). This baseline selects answer-options that best match sentences in a text corpus (Clark et al., 2016). Specifically, for each question $q$ and answer option $a_i$, the IR solver sends $q + a_i$ as a query to a search engine (we use Lucene) on a corpus, and returns the search engine’s score for the top retrieved sentence $s$, where $s$ must have at least one non-stopword overlap with $q$, and at least one with $a_i$.

We create two versions of this system. In the first variation IR(paragraphs) we create a corpus of sentences extracted from all the paragraphs in the dataset. In the second variation, IR(web) in addition to the knowledge of the paragraphs, we use extensive external knowledge extracted from the web (Wikipedia, science textbooks and study guidelines, and other webpages), with $5 \times 10^{10}$ tokens (280GB of plain text).

SurfaceLR (logistic regression baseline). As a simple baseline that makes use of our small training set, we reimplemented and trained a logistic regression model using word-based overlap features. As described in (Merkhofer et al., 2018), this baseline takes into account the lengths of a text, question and each answer candidate, as well as indicator features regarding the (co-)occurrences of any words in them.

SemanticILP (semi-structured baseline). This state-of-the-art solver, originally proposed for science questions and biology tests, uses a semi-structured representation to formalize the scoring problem as a subgraph optimization problem over multiple layers of semantic abstrac-
Table 3: Performance comparison for different baselines tested on a subset of our dataset (in percentage). There is a significant gap between the human performance and current statistical methods.

<table>
<thead>
<tr>
<th></th>
<th>Dev F1m</th>
<th>Dev F1a</th>
<th>Test F1m</th>
<th>Test F1a</th>
</tr>
</thead>
<tbody>
<tr>
<td>Random</td>
<td>44.3</td>
<td>43.8</td>
<td>47.1</td>
<td>47.6</td>
</tr>
<tr>
<td>IR(paragraphs)</td>
<td>64.3</td>
<td>60.0</td>
<td>54.8</td>
<td>53.9</td>
</tr>
<tr>
<td>SurfaceLR</td>
<td>66.1</td>
<td>63.7</td>
<td>66.7</td>
<td>63.5</td>
</tr>
<tr>
<td>Human</td>
<td>86.4</td>
<td>83.8</td>
<td>84.3</td>
<td>81.8</td>
</tr>
</tbody>
</table>

4.2 Results

To get a sense of our dataset’s hardness, we evaluate both human performance and multiple computational baselines. Each baseline scores an answer-option with a real-valued score, which we threshold to decide whether an answer option is selected or not, where the threshold is tuned on the development set. Table 3 shows performance results for different baselines. The significantly high human performance shows that humans do not have much difficulties in answering the questions. Similar observations can be made in Figure 5 where we plot $\text{avg}_{q \in Q}(\text{Pre}(q))$ vs. $\text{avg}_{q \in Q}(\text{Rec}(q))$, for different threshold values.

5 Conclusion

In this paper we have presented MultiRC, a reading comprehension dataset in which questions require reasoning over multiple sentences to be answered. Our dataset contains ~ 6k questions extracted from about +800 paragraphs. For each question, it contains multiple answer-options out of which one or more can be correct. The paragraphs (and questions) originate from different domains and hence are amenable to a wide variety and complexity of required reasoning phenomena. We found human performance on this corpus to be about 88% while state-of-the-art machine comprehension models do not exceed a F1-score of 60%.

6 Acknowledgement

The authors would like to thank all the contributors to the project. This work was supported by Contract HR0011-15-2-0025 with the US Defense Advanced Research Projects Agency (DARPA). Approved for Public Release, Distribution Unlimited. The views expressed are those of the authors and do not reflect the official policy or position of the Department of Defense or the U.S. Government. This work was partly funded by grants from the German Research Foundation (DFG EXC 284 and RO 4848/1-1), by the Allen Institute for Artificial Intelligence (allenai.org); by Google; and by IBM-ILLINOIS Center for Cognitive Computing Systems Research (C3SR)—a research collaboration as part of the IBM AI Horizons Network.
References


Daniel Khaskhabi, Mark Sammons, Ben Zhou, Tom Redman, Christos Christodoulopoulos, Vivek Srikumar, Nicholas Rizzolo, Lev Ratinov, Guanheng Luo, Quang Do, Chen-Tse Tsai, Subhro Roy, Stephen Mayhew, Zhili Feng, John Wieting, Xiaodong Yu, Yungjoo Song, Shashank Gupta, Shyam Upadhyay, Naveen Arivazhagan, Quang Ning, Shaoshi Ling, and Dan Roth. 2018b. CogCompNLP:


Appendix I: More details on mechanical turk interface for the various steps of the question and answer-options creation process
### Instructions

You need to read a passage and construct questions about the information presented in the passage. For each question, you also need to provide the correct answer. The question should be answerable from the passage. **Most importantly, only those questions are allowed whose answer cannot be determined from a single sentence.**

Below are a few example passages and sample questions (and answers).

#### Paragraph:

**Sent 1:** A Republican bid to keep the government open past Friday includes no new money for the construction of a border wall along the U.S.-Mexico border; according to several congressional aides familiar with ongoing talks.

**Sent 2:** GOP leaders submitted the new offer Tuesday afternoon in an effort to appease Democrats, whose votes are needed to avert a shutdown of federal agencies, several House and Senate aides said.

**Sent 3:** However, Republicans also insisted on increases in border security and defense spending, including money to repair existing fencing and new surveillance technology to patrol the nearly 2,000-mile border.

**Sent 4:** Democrats have indicated that they would support such appropriations so long as no money goes toward an actual wall.

#### Question:**

Who has to be appeased to keep the government open?

**Correct Answer:** Democrats

**Sentences needed to answer this question:**

1. 2. 3. 4

**Explanation:** This is a multi-sentence question because, “appease Democrats” is mentioned in Sent 2, while “keep the government open” is part of Sent 1.

#### Question:**

When was the republican bid submitted?

**Correct Answer:** Tuesday afternoon

**Sentences needed to answer this question:**

1. 2. 3. 4

**Explanation:** Republican bid is mentioned in Sent 1 while Tuesday afternoon occurs in Sent 2.

#### Question:**

Who does not want a border wall?

**Correct Answer:** Democrats

**Sentences needed to answer this question:**

1. 2. 3. 4

**Explanation:** Democrats will be appeased by the new offer (Sent 2). The new offer includes no money for the wall (Sent 1). This implies that Democrats don’t want the wall.

#### Example of a bad question:

**Paragraph:**

**Sent 1:** Obama was born on August 4, 1961, at Kapiolani Maternity & Gynecological Hospital in Honolulu, Hawaii.

**Sent 2:** He is the only President to have been born in Hawaii.

**Sent 3:** He was born to a white mother and a black father.

**Sent 4:** His mother, Ann Dunham (1942-1995), was born in Wichita, Kansas, of mostly English descent, with some German, Irish, Scottish, Swiss, and Welsh ancestry.

**Question:**

How old was Obama’s mother when he was born?

**Correct Answer:** 19

**Sentences needed to answer this question:**

1. 2. 3. 4

**Explanation:** Obama was born in 1961. His mother was born in 1942. This question reasoning about time across Sentences 1 and 4 (1961-1942=19).

**Question:**

Where was Ann living in August, 1961?

**Correct Answer:** Hawaii

**Sentences needed to answer this question:**

1. 2. 3. 4

**Explanation:** Ann is Obama’s mother (Sent 4). She must have been there when Obama was born (Sent 1). Obama was born in Hawaii (Sent 2).

**Question:**

Was one of Ann’s children a future president?

**Correct Answer:** Yes

**Sentences needed to answer this question:**

1. 2. 3. 4

**Explanation:** Ann is Obama’s mother (Sent 4). Obama was a future president (Sent 2).

---

- For the paragraph below, create at least 3 questions that require multiple sentences (examples above).
- If you create more than 3 questions, you’ll get bonus rewards.
- For each generated question you write the answer in the space followed by the question, as well as denoting what sentences are required for answering it.
- Items with * are required.
Figure 6: Instructions and Interface for the Step 1 of the dataset creation process. In this step, we request turkers to write a question and its correct answer for a given paragraph. We emphasize the requirement for multi-sentence questions, and additionally ask them to indicate which sentence ids would be needed for answering the composed question. We also provide two example paragraphs with 3 multi-sentence questions each and the corresponding answers. We also provide an explanation indicating why these questions fit our criteria. Lastly, we provide an example of bad question before asking them to compose questions. We ask them to write 3-5 questions per paragraph but the figure only shows the form for one question.
Figure 7: Instructions and Interface for the Step 2 of the dataset creation process. This step was included to ensure that the questions collected from the previous step are indeed multi-sentence. In this step we show a sentence and a question to the turkers and ask them if they can answer the question using only the information present in the shown sentence. In the instructions, we discourage them from using background knowledge since some of our paragraphs are from non-fictional sources. We also provide examples along with an explanation. The bottom part of the figure shows the form for one sentence-question pair.
Instructions

Writing Answer Options for a Reading Comprehension Task

You are designing a reading comprehension test for your students. Students will read a paragraph, a question based on the paragraph; and a few options for its answer. They have to choose the correct answer from the options. For the test to make sense, you want the correct answer to be part of the options. However, you don't want the test to be too easy! To make the test difficult for the students, you want to add incorrect options that might be confused with the correct answer to the list of options. A student who tries to answer a question without reading the paragraph carefully might choose the incorrect option! In this task, your goal is to write the list of options that will be shown to your students.

The questions showed to you in this HIT were written by people and might sometimes contain typing mistakes or other kinds of errors. In such cases, you could additionally check the box underneath the question indicating the “question is bad”, and also correct it if possible. The answer-options you write would be in response to the corrected questions.

Below we show some good and bad examples of correct and incorrect options. We also show examples of bad questions.

Correct Options: A correct option is one that correctly and completely answers the question. Keep in mind that for any question, there can be multiple correct answers. A bad correct option addresses the question only partially or incorrectly. We want to include only good correct options in the list of options. Here are some good and bad examples of correct options:

Paragraph:
Sent 1: Obama was born on August 4, 1961, at Kapiolani Maternity & Gynecological Hospital in Honolulu, Hawai'i.
Sent 2: He is the only President to have been born in Hawai'i.
Sent 3: He was born to a white mother and a black father.
Sent 4: His mother, Ann Dunham (1942-1996), was born in Wichita, Kansas, of mostly English descent, with some German, Irish, Scottish, Swiss, and Welsh ancestry.

Question: How old was Obama's mother when he was born?
Good Correct options (expressed via “/”): 19 / almost twenty
Bad Correct option: 20
Bad Correct option: older than twenty
Explanation: Obama was born in 1991. His mother was born in 1942 and 1942–19 = 19. The good options (“19” and “almost twenty”) answer the question correctly. The bad options give an incorrect answer.

Question: Where was Ann living in August, 1961?
Good Correct options (expressed via “/”): Honolulu // Hawai'i
Bad Correct option: Kansas
Bad Correct option: Scotland
Explanation: Ann is Obama’s mother (Sent 4). She must have been there when Obama was born (Sent 1). Obama was born in Hawai'i (Sent 2). Hawai'i is also in Honolulu. So both Hawai'i and Honolulu answer the question correctly. Kansas and Scotland don’t answer the question correctly.

Incorrect Options: An incorrect option will be closely related to the content of the paragraph and/or the question but will not truthfully answer the question. They should not be trivial, in the sense that they should not be easily eliminated as the answer without reading the paragraph. They are presented as options to confuse your students and to make the reading comprehension task challenging for them. Here are some good and bad examples of incorrect options:

Paragraph:
Sent 1: It was hot that day.
Sent 2: The temperature on the wall of the backyard was showing something over 100 F.
Sent 3: Meanwhile Tom, at home, was trying finish the remainder of carrots from last night, and packing for his trip to Chicago tomorrow.
Sent 4: As employees of the Art Museum, Tom and his older cousin often had to travel to Chicago.

Question: What was the outside temperature when Tom was eating carrots?
Good Incorrect options (expressed via “/”): Not very hot // Far below 100 F
Bad Incorrect option: Carrots
Bad Incorrect option: Wall
Explanation: The good options appear in the paragraph (at least partially). They are also related to the content of the question (temperature), but do not answer the question correctly. Your student will have to carefully read the paragraph to identify that these don’t answer the question correctly. The bad options appear in the paragraph (e.g. Wall) and sometimes also in the question (e.g. carrots) but they are trivially incorrect. A student can easily determine that they don’t answer the question correctly without even reading the paragraph.

Question: What did Tom not visit?
Good Incorrect options (expressed via “/”): Tom’s older cousin // Art Museum of Chicago
Bad Incorrect option: A flower
Bad Incorrect option: Two Ozma bags
Explanation: The good options are clearly related to the content of the paragraph and question, but do not answer the question correctly. Your student will have to carefully read the paragraph to identify that these don’t answer the question correctly. The bad options are not in the paragraph at all, and don’t challenge the student’s comprehension skills.

Bad Questions: Sometimes a question could have grammatical errors, typing mistakes, etc. or might not make sense in the context of the paragraph (for instance, it might not be related to the content of the paragraph or not be answerable at all). In such cases, please mark the question as a bad question. If the question contains minor errors please correct it in the boxed space provided underneath the question, otherwise leave the boxed space empty. Here are some examples of bad questions:

Paragraph:
Sent 1: It was hot that day.
Sent 2: The temperature on the wall of the backyard was showing something over 100 F.
Sent 3: Meanwhile Tom, at home, was trying finish the remainder of carrots from last night, and packing for his trip to Chicago tomorrow.
Sent 4: As employees of the Art Museum, Tom and his older cousin often had to travel to Chicago.

Question: What was the temperature outside when Tom was eating carrots?
Is this a bad question?
Corrected Question, if bad: What was the temperature outside when Tom was eating carrots?
Explanation: It is easy to identify that there is a typing mistake, and Tom should be Tom.

Question: When did Tom walk out of the party?
Is this a bad question?
Corrected Question, if bad: The question doesn’t make sense for this paragraph. The paragraph does not discuss Tom attending any party. So, this is a bad question and cannot be corrected.

For each of the following questions, write good correct and incorrect options that would be shown to your students. Additionally, if the question is bad please indicate that and correct it if possible.
Figure 8: Instructions and Interface for the Step 3 of the dataset creation process. This step was included to collect correct and in-correct answer options for the questions in our dataset. In the instructions we define what we mean by correct and incorrect answer options—correct options answer the given question correctly and completely, whereas incorrect options do not answer the question but are non-trivial and cannot be easily eliminated as an incorrect answer without reading the passage. We also show examples of good and bad inputs for both along with explanations. The bottom part of the figure shows the form for entering correct and incorrect answer-options for one paragraph-question pair.
Figure 9: Instructions and Interface for the Step 4 of the dataset creation process. In this step we ask the annotators to read the questions and answer-options, and indicate if they find any errors in them. We also ask them to select all answer-options that correctly answer the question (illustrated through examples). Any possibly discrepancies or errors were later reviewed and corrected manually. The bottom part of the figure shows a sample form for one question.
## Appendix II: Examples from the corpus

<table>
<thead>
<tr>
<th>Sentence 1</th>
<th>Sentence 2</th>
<th>Sentence 3</th>
<th>Sentence 4</th>
<th>Sentence 5</th>
</tr>
</thead>
<tbody>
<tr>
<td>Dirk Diggler was born as Steven Samuel Adams on April 15, 1961 outside of Saint Paul, Minnesota.</td>
<td>His parents were a construction worker and a boutique shop owner who attended church every Sunday and believed in God.</td>
<td>Looking for a career as a male model, Diggler dropped out of school at age 16 and left home.</td>
<td>He was discovered at a falafel stand by Jack Horner.</td>
<td>Diggler met his friend, Reed Rothchild, through Horner in 1979 while working on a film.</td>
</tr>
</tbody>
</table>

**Question:** How old was Dirk when he met his friend Reed?  
A) 18  
B) 16  
C) 22

**Reasoning needed:** Temporal and mathematical reasoning  
One needs to identify that Dirk was born in 1961 (Sent 1) and he met Reed in 1979 (Sent 5). At that time he must have been 18 (=1979-1961) years old.

<table>
<thead>
<tr>
<th>Sent 1: The hijackers attacked at 9:28.</th>
<th>Sent 2: While traveling 35,000 feet above eastern Ohio, United 93 suddenly dropped 700 feet.</th>
<th>Sent 3: Eleven seconds into the descent, the FAA's air traffic control center in Cleveland received the first of two radio transmissions from the aircraft.</th>
<th>Sent 4: During the first broadcast, the captain or first officer could be heard declaring “Mayday” amid the sounds of a physical struggle in the cockpit.</th>
<th>Sent 5: The second radio transmission, 35 seconds later, indicated that the fight was continuing.</th>
</tr>
</thead>
<tbody>
<tr>
<td>Sent 6: The captain or first officer could be heard shouting: “Hey get out of here-get out of here-get out of here.”</td>
<td>Sent 7: On the morning of 9/11, there were only 37 passengers on United 93-33 in addition to the 4 hijackers.</td>
<td>Sent 8: This was below the norm for Tuesday mornings during the summer of 2001.</td>
<td>Sent 9: But there is no evidence that the hijackers manipulated passenger levels or purchased additional seats to facilitate their operation.</td>
<td>Sent 10: The terrorists who hijacked three other commercial flights on 9/11 operated in five-man teams.</td>
</tr>
<tr>
<td>Sent 11: They initiated their cockpit takeover within 30 minutes of takeoff.</td>
<td>Sent 12: On Flight 93, however, the takeover took place 46 minutes after takeoff and there were only four hijackers.</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

**Question:** Which two factors were different between the three other hijacked planes and United 93?  
A) The amount of time that passed before the takeover started  
B) United 93 took longer and had less hijackers  
C) The airline operating the planes  
D) The weather and fuel used by the airplane  
E) The navigation system used by the planes

**Reasoning needed:** Discourse relation (contrast)  
One needs to identify that the discourse marker however in Sent 12 indicates a contrast relation between Flight 93 and the flights mentioned in Sent 10. Also, only in Sent 12 indicates that the number of hijackers were fewer than in the contrasted other flights.

<table>
<thead>
<tr>
<th>Sent 1: The hijackers attacked at 9:28.</th>
<th>Sent 2: While traveling 35,000 feet above eastern Ohio, United 93 suddenly dropped 700 feet.</th>
<th>Sent 3: Eleven seconds into the descent, the FAA's air traffic control center in Cleveland received the first of two radio transmissions from the aircraft.</th>
<th>Sent 4: During the first broadcast, the captain or first officer could be heard declaring “Mayday” amid the sounds of a physical struggle in the cockpit.</th>
<th>Sent 5: The second radio transmission, 35 seconds later, indicated that the fight was continuing.</th>
</tr>
</thead>
<tbody>
<tr>
<td>Sent 6: The captain or first officer could be heard shouting: “Hey get out of here-get out of here-get out of here.”</td>
<td>Sent 7: On the morning of 9/11, there were only 37 passengers on United 93-33 in addition to the 4 hijackers.</td>
<td>Sent 8: This was below the norm for Tuesday mornings during the summer of 2001.</td>
<td>Sent 9: But there is no evidence that the hijackers manipulated passenger levels or purchased additional seats to facilitate their operation.</td>
<td>Sent 10: The terrorists who hijacked three other commercial flights on 9/11 operated in five-man teams.</td>
</tr>
<tr>
<td>Sent 11: They initiated their cockpit takeover within 30 minutes of takeoff.</td>
<td>Sent 12: On Flight 93, however, the takeover took place 46 minutes after takeoff and there were only four hijackers.</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

**Question:** What was below average for this particular day?  
A) the number of passengers in the first class.  
B)* the number of passengers on board.  
C) the number of hijackers  
D) the amount of air traffic in the skies  
E) the temperature

**Reasoning needed:** Event coreference  
One needs to identify that This in Sent 8 co-refers to (event of) number of passengers in Sent 7. Note that Sent 12 contains only four hijackers and understanding that only indicates a smaller number of entities than expected (as in previous question), might mislead a system into believing that (C) is the correct answer.

---

**Figure 10:** Examples from our MultiRC corpus. Each example shows relevant excerpts from a paragraph; multi-sentence question; and corresponding answer-options. The correct answer(s) is indicated by a *. Each example also contains a list of reasoning phenomenon needed to answer that question.
Suddenly, Amy screamed.
I whirled around and threw up my arm just in time to knock Dvorov away.
His eyes were glowing red, and he had bared a set of fangs that could probably take my hand off.

On the floor was Dvorov’s head, separated from his body by several feet and a growing puddle of dark blood.
Nepthys stood over him with a blue sword in his hand.
It flashed in the strobe where it wasn’t streaked with Dvorov’s blood.

**Question:** Who cut off Dvorov’s head?
A) Nepthys  
B) the wolf

**Reasoning needed:** Implicit semantic role (agent) and causation
From sentences 8–10, indicate that Nepthys stood over Dvorov’s (dead) body with a sword which was streaked with Dvorov’s blood. Therefore, one can infer that the blood on the sword must have been caused by the action of killing of Dvorov and that Nepthys is the implicit agent of this action.

**Question:** What was the weapon used to separate Dvorov’s head from his body?
A) a hook  
B) a blue sword  
C) a sword

**Reasoning needed:** Implicit semantic role (instrument) and causation
From sentences 8–10, indicate that Nepthys stood over Dvorov’s (dead) body with a sword which was streaked with Dvorov’s blood. Therefore, one can infer that the blood on the sword must have been caused by the separation of Dvorov’s head from his body and that the sword must have been the implicit instrument of this action.

Figure 11: More example questions and corresponding reasoning phenomena from our MultiRC corpus.