The NarrativeQA Reading Comprehension Challenge

Tomáš Kočiský[†‡] **Jonathan Schwarz**[†] **Phil Blunsom**[†‡] **Chris Dyer**[†]
Karl Moritz Hermann[†] **Gábor Melis**[†] **Edward Grefenstette**[†]
[†]DeepMind [‡]University of Oxford
{tkocisky,schwarzjn,pblunsom,cdyer,kmh,melisgl,etg}@google.com

Abstract

Reading comprehension (RC)—in contrast to information retrieval—requires integrating information and reasoning about events, entities, and their relations across a full document. Question answering is conventionally used to assess RC ability, in both artificial agents and children learning to read. However, existing RC datasets and tasks are dominated by questions that can be solved by selecting answers using superficial information (e.g., local context similarity or global term frequency); they thus fail to test for the essential integrative aspect of RC. To encourage progress on deeper comprehension of language, we present a new dataset and set of tasks in which the reader must answer questions about stories by reading entire books or movie scripts. These tasks are designed so that successfully answering their questions requires understanding the underlying narrative rather than relying on shallow pattern matching or salience. We show that although humans solve the tasks easily, standard RC models struggle on the tasks presented here. We provide an analysis of the dataset and the challenges it presents.

Title: Ghostbusters II
Question: How is Oscar related to Dana?
Answer: her son
Summary snippet: ... Peter's former girlfriend Dana Barrett has had a son, Oscar...
Story snippet:

DANA (setting the wheel brakes on the buggy)
Thank you, Frank. I'll get the hang of this eventually.

She continues digging in her purse while Frank leans over the buggy and makes funny faces at the baby, OSCAR, a very cute nine-month old boy.

FRANK (to the baby)
Hiya, Oscar. What do you say, slugger?

FRANK (to Dana)
That's a good-looking kid you got there, Ms. Barrett.

Figure 1: Example question–answer pair. The snippets here were extracted by humans from summaries and the full text of movie scripts or books, respectively, and are *not* provided to the model as supervision or at test time. Instead, the model will need to read the full text and locate salient snippets based solely on the question and its reading of the document in order to generate the answer.

1 Introduction

Natural language understanding seeks to create models that read and comprehend text. A common strategy for assessing the language understanding capabilities of comprehension models is to demonstrate that they can answer questions about documents they read, akin to how reading comprehension is tested in children when they are learning to read. After reading a document, a reader usually can not reproduce the entire text from memory, but often can answer questions about underlying narrative elements of the document: the salient entities, events, places, and the relations between them. Thus, testing understanding requires the creation of questions that examine high-level abstractions instead of just facts occurring in one sentence at a time.

Unfortunately, superficial questions about a document may often be answered successfully (by both humans and machines) using a shallow pattern match-

Transactions of the Association for Computational Linguistics, vol. 6, pp. 317–328, 2018. Action Editor: Katrin Erk.
Submission batch: 6/2017; Revision batch: 9/2017; Published 5/2018.

ing strategies or guessing based on global salience. In the following section, we survey existing QA datasets, showing that they are either too small or answerable by shallow heuristics (Section 2). On the other hand, questions which are not about the surface form of the text, but rather about the underlying narrative, require the formation of more abstract representations about the events and relations expressed in the course of the document. Answering such questions requires that readers integrate information which may be distributed across several statements throughout the document, and generate a cogent answer on the basis of this integrated information. That is, they test that the reader comprehends language, not just that it can pattern match. We present a new task and dataset, which we call NarrativeQA, which will test and reward artificial agents approaching this level of competence (Section 3), and make available online.[1]

The dataset consists of *stories*, which are books and movie scripts, with human written questions and answers based solely on human-generated abstractive *summaries*. For the RC tasks, questions may be answered using just the summaries or the full story text. We give a short example of a sample movie script from this dataset in Figure 1. Fictional stories have a number of advantages as a domain (Schank and Abelson, 1977). First, they are largely self-contained: beyond the basic fundamental vocabulary of English, all of the information about salient entities and concepts required to understand the narrative is present in the document, with the expectation that a reasonably competent language user would be able to understand it.[2] Second, story summaries are abstractive and generally written by independent authors who know the work only as a reader.

2 Review of Reading Comprehension Datasets and Models

There are a large number of datasets and associated tasks available for the training and evaluation of reading comprehension models. We summarize the key features of a collection of popular recent datasets in Table 1. In this section, we briefly discuss the nature and limitations of these datasets and their associated tasks.

MCTest (Richardson et al., 2013) is a collection of short stories, each with multiple questions. Each such question has set of possible answers, one of which is labelled as correct. While this could be used as a QA task, the MCTest corpus is in fact intended as an answer selection corpus. The data is human generated, and the answers can be phrases or sentences. The main limitation of this dataset is that it serves more as a an evaluation challenge than as the basis for end-to-end training of models, due to its relatively small size.

In contrast, CNN/Daily Mail (Hermann et al., 2015), Children's Book Test (CBT) (Hill et al., 2016), and BookTest (Bajgar et al., 2016) each provide large amounts of question–answer pairs. Questions are Cloze-form (predict the missing word) and are produced from either short abstractive summaries (CNN/Daily Mail) or from the next sentence in the document the context was taken from (CBT and BookTest). The tasks associated with these datasets are all selecting an answer from a set of options, which is explicitly provided for CBT and BookTest, and is implicit for CNN/Daily Mail, as the answers are always entities from the document. This significantly favors models that operate by pointing to a particular token (or type). Indeed, the most successful models on these datasets, such as the Attention Sum Reader (AS Reader) (Kadlec et al., 2016), exploit precisely this bias in the data. However, these models are inappropriate for answers requiring synthesis of a new answer. This bias towards answers that are shallowly salient is a more serious limitation of the CNN/Daily Mail dataset, since its context documents are news stories which usually contain a small number of salient entities and focus on a single event.

Stanford Question Answering Dataset (SQuAD) (Rajpurkar et al., 2016) and NewsQA (Trischler et al., 2016) offer a different challenge. A large number of questions and answers are provided for a set of documents, where the answers are *spans* of the context document, i.e. contiguous sequences of words from the document. Although the answers are not

[1] http://deepmind.com/publications

[2] For example, new names and words may be coined by the author (e.g. "muggle" in Harry Potter novels) but the reader need only appeal to the book itself to understand the meaning of these concepts, and their place in the narrative. This ability to form new concepts based on the contexts of a text is a crucial aspect of reading comprehension, and is in part tested as part of the question answering tasks we present.

Dataset	Documents	Questions	Answers
MCTest (Richardson et al., 2013)	660 short stories, grade school level	2640 human generated, based on the document	multiple choice
CNN/Daily Mail (Hermann et al., 2015)	93K+220K news articles	387K+997K Cloze-form, based on highlights	entities
Children's Book Test (CBT) (Hill et al., 2016)	687K of 20 sentence passages from 108 children's books	Cloze-form, from the 21st sentence	multiple choice
BookTest (Bajgar et al., 2016)	14.2M, similar to CBT	Cloze-form, similar to CBT	multiple choice
SQuAD (Rajpurkar et al., 2016)	23K paragraphs from 536 Wikipedia articles	108K human generated, based on the paragraphs	spans
NewsQA (Trischler et al., 2016)	13K news articles from the CNN dataset	120K human generated, based on headline, highlights	spans
MS MARCO (Nguyen et al., 2016)	1M passages from 200K+ documents retrieved using the queries	100K search queries	human generated, based on the passages
SearchQA (Dunn et al., 2017)	6.9m passages retrieved from a search engine using the queries	140k human generated Jeopardy! questions	human generated Jeopardy! answers
NarrativeQA (this paper)	1,572 stories (books, movie scripts) & human generated summaries	46,765 human generated, based on summaries	human generated, based on summaries

Table 1: Comparison of datasets.

just single word/entity answers, many plausible questions for assessing RC cannot be asked because no document span would contain its answer. While they provide a large number of questions, these are from a relatively small number of documents, which are themselves fairly short, thereby limiting the lexical and topical diversity of models trained on this data. While the answers are multi-word phrases, the spans are generally short and rarely cross sentence boundaries. Simple models scoring and/or extracting candidate spans conditioned on the question and superficial signal from the rest of the document do well, e.g., Seo et al. (2016). These models will not trivially generalize to problems where the answers are not spans in the document, supervision for spans is not provided, or several discontinuous spans are needed to generate a correct answer. This restricts the scalability and applicability of models doing well on SQuAD or NewsQA to more complex problems.

MS MARCO dataset (Nguyen et al., 2016) presents a bolder challenge: questions are paired with sets of snippets ("context passages") that contain the information necessary to answer the question and answers are free-form human generated text. However, as no restriction was placed on annotators to prevent them from copying answers from source documents, many answers are in fact verbatim copies of short spans from the context passages. Models that do well on SQuAD (e.g., Wang and Jiang (2016), Weissenborn et al. (2017)), extracting spans or pointing, do well here too, and the same concerns about the general applicability of solutions to this particular dataset to larger reading comprehension problems apply here also, as above.

SearchQA (Dunn et al., 2017) is a recent dataset in which the context for each question is a set of documents retrieved by a search engine using the question as the query. However, in contrast with previous datasets neither questions nor answers were produced by annotating the context documents, but rather the context documents were retrieved after collecting pre-existing question–answer pairs. As such, it is not open to same annotation bias as the datasets discussed above. However, upon examining answers in the Jeopardy data used to construct this dataset, one finds that 80% of answers are bigrams or unigrams, and 99% are 5 tokens or fewer. Of a sample of 100 answers, 72% are named entities, all are short noun-phrases.

Summary of Limitations. We see several limitations of the scope and depth of the RC problems in existing datasets. First, several datasets are small (MCTest) or not overly naturalistic (bAbI (Weston et al., 2015)). Second, in more naturalistic documents, a majority of questions require only a single sentence to locate supporting information for answering (Chen et al., 2016; Rajpurkar et al., 2016). This, we suspect, is largely an artifact of the question generation methodology, in which annotators have created questions from a context document, or where context documents that explicitly answer a question are iden-

tified using a search engine. Although the factoid-like Jeopardy questions of SearchQA also appear to favor questions answerable with local context. Finally, we see further evidence of the superficiality of the questions in the architectures that have evolved to solve them, which tend to exploit span selection based on representations derived from local context and the query (Seo et al., 2016; Wang et al., 2017).

3 NarrativeQA: A New Dataset

In this section, we introduce our new dataset, NarrativeQA, which addresses many of the limitations identified in existing datasets.

3.1 Desiderata

From the above discussed features and limitations, we define our desiderata as follows. We wish to construct a dataset with a large number of question–answer pairs based on either a large number of supporting documents or from a smaller collection of large documents. This permits the training of neural network-based models over word embeddings and provides decent lexical coverage and diversity. The questions and answers should be natural, unconstrained, and human generated; and answering questions should frequently require reference to several parts or a larger span of the context document rather than superficial representations of local context. Furthermore, we want annotators to express, in their own words, higher-level relations between entities, places, and events, rather than copy short spans of the document.

Furthermore, we want to evaluate models both on the fluency and correctness of generated free-form answers, and as an answer selection problem, which requires the provision of sensible distractors to the correct answer. Finally, the scope and complexity of the QA problem should be such that current models struggle, while humans are capable of solving the task correctly, so as to motivate further research into the development of models seeking human reading comprehension ability.

3.2 Data Collection Method

We will consider complex, self-contained narratives as our documents/stories. To make the annotation tractable and lead annotators towards asking non-localized questions, we will only provide them human written summaries of the stories for generating the question–answer pairs.

We present both books and movie scripts as stories in our dataset. Books were collected from Project Gutenberg[3] and movie scripts are scraped from the web.[4] We matched our stories with plot summaries from Wikipedia using titles and verified the matching with help from human annotators. The annotators were asked to determine if both the story and the summary refer to a movie or a book (as some books are made into movies), or if they are the same part in a series produced in the same year. In this way we obtained 1,567 stories. This provides with a smaller set of documents, compared to the other datasets, but the documents are long which provides us with good lexical coverage and diversity. The bottleneck for obtaining a larger number of publicly available stories was finding corresponding summaries.

Annotators on Amazon Mechanical Turk were instructed to write 10 question–answer pairs each based solely on a given summary. Reading and annotating summaries is tractable unlike writing questions and answers based on the full stories, and moreover, as the annotators never see the full stories we are much less likely to get questions and answers which are extracted from a localized context.

Annotators were instructed to imagine that they are writing questions to test students who have read the full stories but not the summaries. We required questions that are specific enough, given the length and complexity of the narratives, and to provide a diverse set of questions about characters, events, why this happened, and so on. Annotators were encouraged to use their own words and we prevented them from copying.[5] We asked for answers that are grammatical, complete sentences, and explicitly allowed short answers (one word, or a few-word phrase, or a short sentence) as we think that answering with a full sentence is frequently perceived as artificial when asking about factual information. Annotators were asked to avoid extra, unnecessary information in the question or the answer, and to avoid yes/no questions

[3] http://www.gutenberg.org/

[4] Mainly from http://www.imsdb.com/, but also http://www.dailyscript.com/, http://www.awesomefilm.com/.

[5] This was done both through instructions and Javascript hard limitations on the annotation site.

or questions about the author or the actors.

About 30 question–answer pairs per summary were obtained. The result is a collection of human written natural questions and answers. As we have multiple questions per summary/story, this allows us to consider answer selection (from among the 30) as a simpler version of the QA rather than answer generation from scratch. Answer selection (Hewlett et al., 2016) and multiple-choice question answering (Richardson et al., 2013; Hill et al., 2016) are frequently used.

We additionally collected a second reference answer for each question by asking annotators to judge whether a question is answerable, given the summary, and provide an answer if it was. All but 2.3% of the questions were judged as answerable.

3.3 Core Statistics

We collected 1,567 stories, evenly split between books and movie scripts. We partitioned the dataset into non-overlapping training, validation, and test portions, along stories/summaries. See Table 2 for detailed statistics.

The dataset contains 46,765 question–answer pairs. The questions are grammatical questions written by human annotators, that average 9.8 tokens in length, and are mostly formed as 'WH'-questions (see Table 3). We categorized a sample of 300 questions in Table 4. We observed a good variety of question types. An interesting category are questions which ask for something related to, or occurring together, before, or after with an event, of which there are about 15%.

Answers in the dataset are human written natural answers that are short, averaging 4.73 tokens, but are not restricted to spans from the documents. There are answers that appear as spans of the summaries and the stories, 44.05% and 29.57%, respectively. As expected, lower proportion of answers are spans on stories compared to summaries on which they were constructed.

3.4 Tasks

We present tasks varying in their scope and complexity: we consider either the summary or the story as context, and for each we evaluate answer generation and answer selection.

The task of answering questions based on summaries is similar in scope to previous datasets. However, summaries contain more complex relationships and timelines than news articles or short paragraphs from the web and thus provide a task different in nature. We hope that NarrativeQA will motivate the design of architectures capable of modeling such relationships. This setting is similar to the previous tasks in that the questions and answers were constructed based on these supporting documents.

The full version of NarrativeQA requires reading and understanding entire stories (i.e., books and movie scripts). At present, this task is intractable for existing neural models out of the box. We further discuss the challenges and possible approaches in the following sections.

We require the use of metrics for generated text. We evaluate using BLEU-1, BLEU-4 (Papineni et al., 2002), Meteor (Denkowski and Lavie, 2011), and ROUGE-L (Lin, 2004), using two references for each question,[6] except for the human baseline where we evaluate one reference against the other. We also evaluate our models using a ranking metric. This allows us to evaluate how good our model is at reading comprehension regardless of how good it is at generating answers. We rank answers for questions associated with the same summary/story and compute the mean reciprocal rank (MRR).[7]

4 Baselines and Oracles

In this section, we show that NarrativeQA presents a challenging problem for current approaches to reading comprehension by evaluating several baselines based on information retrieval (IR) techniques and neural models. Since neural models use quite different processes for generating answers (e.g., predicting a single word or entity, selecting a span of the document context, or open generation of the answer sequence), we present results on each. We also report the human performance by scoring the second reference answer against the first.

[6] We lowercase both the candidates and the references and remove the end of sentence marker and the final full stop.

[7] MRR is the mean over examples of $1/r$, where $r \in \{1, 2, \ldots\}$ is the rank of the correct answer among candidates.

	train	valid	test
# documents	1,102	115	355
... books	548	58	177
... movie scripts	554	57	178
# question–answer pairs	32,747	3,461	10,557
Avg. #tok. in summaries	659	638	654
Max #tok. in summaries	1,161	1,189	1,148
Avg. #tok. in stories	62,528	62,743	57,780
Max #tok. in stories	430,061	418,265	404,641
Avg. #tok. in questions	9.83	9.69	9.85
Avg. #tok. in answers	4.73	4.60	4.72

Table 2: NarrativeQA dataset statistics.

First token	Frequency
What	38.04%
Who	23.37%
Why	9.78%
How	8.85%
Where	7.53%
Which	2.21%
How many/much	1.80%
When	1.67%
In	1.19%
OTHER	5.57%

Table 3: Frequency of first token of the question in the training set.

Category	Frequency
Person	30.54%
Description	24.50%
Location	9.73%
Why/reason	9.40%
How/method	8.05%
Event	4.36%
Entity	4.03%
Object	3.36%
Numeric	3.02%
Duration	1.68%
Relation	1.34%

Table 4: Question categories on a sample of 300 questions from the validation set.

4.1 Simple IR Baselines

We consider basic IR baselines which retrieve an answer by selecting a span of tokens from the context document based on a similarity measure between the candidate span and a query. We compare two queries: the question and (as an oracle) the gold standard answer. The answer oracle provides an upper bound on the performance of span retrieval models, including the neural models discussed below. When using the question as the query, we obtain generalization results of IR methods. Test set results are computed by extracting either 4-gram, 8-gram, or full-sentence spans according to the best performance on the validation set.[8]

We consider three similarity metrics for extracting spans: BLEU-1, ROUGE-L, and the cosine similarity between bag-of-words embedding of the query and the candidate span using pre-trained GloVe word embeddings (Pennington et al., 2014).

4.2 Neural Benchmarks

As a first benchmark we consider a simple bi-directional LSTM sequence to sequence (Seq2Seq) model (Sutskever et al., 2014) predicting the answer directly from the query. Importantly, we provide no context information from either summary or story. Such a model might classify the question and predict an answer of a similar topic or category.

[8]Note that we do not consider the span's context when computing the MRR for IR baselines, as the candidate spans (i.e. all answers to questions on the story) are given and simply ranked by their similarity to the query.

Previous reading comprehension tasks such as CNN/Daily Mail motivated models constrained to predicting a single token from the input sequence. The AS Reader (Attention Sum Reader (Kadlec et al., 2016)) considers the entire context and predicts a distribution over unique word types. We adapt the model for sequence prediction by using an LSTM sequence decoder and choosing a token from the input at each step of the output sequence.

As a span-prediction model we consider a simplified version of the Bi-Directional Attention Flow network (Seo et al., 2016). We omit the character embedding layer and learn a mapping from words to a vector space rather than making use of pre-trained embeddings; and we use a single layer bi-directional LSTM to model interactions among context words conditioned on the query (modelling layer). As proposed, we adopt the output-layer tailored for span-prediction and leave the rest unchanged. It was not our aim to use the state-of-the-art model for other datasets but rather to provide a strong benchmark.

Span prediction models can be trained by obtaining supervision on the training set from the oracle IR model. We use start and end indices of the span achieving the highest ROUGE-L score with respect to the reference answers as labels on the training set. The model is then trained to predict these spans by maximizing the probability of the indices.

4.3 Neural Benchmarks on Stories

The design of the NarrativeQA dataset makes the straight-forward application of the existing neural ar-

Model	Validation / Test				
	BLEU-1	BLEU-4	Meteor	ROUGE-L	MRR
IR Baselines					
BLEU-1 given question (1 sentence)	10.48/10.75	3.02/ 3.34	11.93/12.33	14.34/14.90	0.176/0.171
ROUGE-L given question (8-gram)	11.74/11.01	2.18/ 1.99	7.05/ 6.50	12.58/11.74	0.168/0.161
Cosine given question (1 sentence)	7.49/ 7.51	1.88/ 1.97	10.18/10.35	12.01/12.28	0.170/0.171
Random rank					0.133/0.133
Neural Benchmarks					
Seq2Seq (no context)	16.10/15.89	1.40/ 1.26	4.22/ 4.08	13.29/13.15	0.211/0.202
Attention Sum Reader	23.54/23.20	5.90/ 6.39	8.02/ 7.77	23.28/22.26	**0.269/0.259**
Span Prediction	**33.45/33.72**	**15.69/15.53**	**15.68/15.38**	**36.74/36.30**	—
Oracle IR Models					
BLEU-1 given answer (ans. length)	54.60/55.55	26.71/27.78	31.32/32.08	58.90/59.77	1.000/1.000
ROUGE-L given answer (ans. length)	52.94/54.14	27.18/28.18	30.81/31.50	59.09/59.92	1.000/1.000
Cosine given answer (ans. length)	46.69/47.95	24.25/25.25	27.02/27.81	44.64/45.66	0.836/0.838
Human (given summaries)	44.24/44.43	18.17/19.65	23.87/24.14	57.17/57.02	—

Table 5: Experiments on summaries. Higher is better for all metrics. Sections 4.1 and 4.2 explain the IR and neural models, respectively.

chitectures computationally infeasible, as this would require running an recurrent neural network on sequences of hundreds of thousands of time steps or computing a distribution over the entire input for attention, as is common.

We split the task into two steps: first, we retrieve a small number of relevant passages from the story using an IR system; second, we apply one of the neural models on the resulting document. The question becomes the query for retrieval. This IR problem is much harder than traditional document retrieval, as the documents, the passages here, are very similar, and the question is short and entities mentioned likely occur many times in the story.

Our retrieval system considers chunks of 200 words from the story and computes representations for all chunks and the query. We then select a varying number of such chunks based on their similarity to the query. We experiment with different representations and similarity measures in Section 5. Finally, we concatenate the selected chunks in the correct temporal order and insert delimiters between them to obtain a much shorter document. For span prediction models, we then further select a span from the retrieved chunks as described in Section 4.2.

5 Experiments

In this section, we describe the data preparation methodology we used, and the experimental results on the summary-reading task as well as the full story task.

5.1 Data Preparation

The provided narratives contain a large number of named entities (such as names of characters or places). Inspired by Hermann et al. (2015), we replace such entities with markers, such as `@entity42`. These markers are permuted during training and testing so that none of their embeddings learn a specific entity's representation. This allows us to build representations for entities from stories that were never seen in training, since they are given a specific identifier (to differentiate them from other entities in the document) from a set of generic identifiers re-used across documents. Entities are replaced according to a simple heuristic based on a capital first character and the respective word not appearing in lowercase.

5.2 Reading Summaries Only

Reading comprehension of summaries is similar to a number of previous reading comprehension tasks where questions were constructed based on the context document. However, plot summaries tend to

Model	Validation / Test				
	BLEU-1	**BLEU-4**	**Meteor**	**ROUGE-L**	**MRR**
IR Baselines					
BLEU-1 given question (8-gram)	6.73/ 6.52	0.30/ 0.34	3.58/ 3.35	6.73/ 6.45	0.176/0.171
ROUGE-L given question (1 sentence)	5.78/ 5.69	0.25/ 0.32	3.71/ 3.64	6.36/ 6.26	0.168/0.161
Cosine given question (8-gram)	6.40/ 6.33	0.28/ 0.29	3.54/ 3.28	6.50/ 6.43	0.171/0.171
Random rank					0.133/0.133
Neural Benchmarks					
Attention Sum Reader given 1 chunk	16.95/16.08	1.26/1.08	3.84/3.56	12.12/11.94	0.164/0.161
Attention Sum Reader given 2 chunks	18.54/17.76	0.0/1.1	4.2/4.01	13.5/12.83	0.169/0.169
Attention Sum Reader given 5 chunks	18.91/18.36	1.37/1.64	4.48/4.24	14.47/13.4	0.171/0.173
Attention Sum Reader given 10 chunks	**20.0/19.09**	2.23/1.81	4.45/4.29	**14.47/14.03**	0.182/0.177
Attention Sum Reader given 20 chunks	19.79/19.06	**1.79/2.11**	**4.6/4.37**	14.86/14.02	**0.182/0.179**
Span Prediction	5.82/5.68	0.22/0.25	3.84/3.72	6.33/6.22	—
Oracle IR Models					
BLEU-1 given answer (ans. length)	41.81/42.37	7.03/ 7.70	19.10/19.52	46.40/47.15	1.000/1.000
ROUGE-L given answer (ans. length)	39.17/39.50	7.81/ 8.46	18.13/18.55	48.91/49.94	1.000/1.000
Cosine given answer (4-gram)	38.21/38.92	7.78/ 8.43	12.58/12.60	31.24/31.70	0.842/0.845
Human (given summaries)	44.24/44.43	18.17/19.65	23.87/24.14	57.17/57.02	—

Table 6: Experiments on full stories. Each chunk contains 200 tokens. Higher is better for all metrics. Sections 4.1 and 4.2 explain the IR and neural models, respectively. Note that the human scores are based on answering questions given summaries, same as in Table 5.

contain more intricate event time lines and a larger number of characters, and in this sense, are more complex to follow than news articles or paragraphs from Wikipedia. See Table 5 for the results.

Given that questions were constructed based on the summaries, we expected that both neural models and span-selection models would perform well. This is indeed the case, with the neural span prediction model significantly outperforming all other proposed methods. However, significant room remains for improvement when compared with the oracle and human scores.

Both the plain sequence-to-sequence model and the AS Reader, successfully applied to the CNN/DailyMail reading comprehension task, also performed well on this task. We observe that the AS Reader tends to copy subsequent tokens from the context, thus behaving like a span prediction model. An additional inductive bias results in higher performance for the span prediction model. Similar observations between AS Reader and span models have also been made by Wang and Jiang (2016).

Note that we have tuned each model separately on the development set twice: once selecting the best model based on ROUGE-L, reporting the first four metrics, and a second time selecting based on the MRR.

5.3 Reading Full Stories Only

Table 6 summarizes the results on the full NarrativeQA task, where the context documents are full stories. As expected (and desired), we observe a decline in performance of the span-selection oracle IR model, compared to the results on summaries. This is unsurprising as the questions were constructed on summaries and confirms the initial motivation for designing this task. As previously, we considered all spans of a given length across the entire story for this model. For short answers of one or two words—typically main characters in a story—the candidate (i.e., the closest span to the reference answer) is easily found due to being mentioned throughout the text. For longer answers it becomes much less likely, compared to the summaries, that a high-scoring span can be found in the story. Note that this distinguishes NarrativeQA from many of the reviewed datasets.

In our IR plus neural two-step approach to the task, we first retrieve relevant chunks of the stories and then apply existing reading comprehension models. We use the questions to guide the IR system for chunk

extraction, with the results of the standalone IR baselines giving an indication of the difficulty of this aspect of the task. The retrieval quality has a direct effect on the performance of all neural models—a challenge which models on summaries are not presented with. We considered several approaches to chunk selection: we retrieve chunks based on the highest ROUGE-L or BLEU-1 scoring span with respect to the question in the story; comparing topic distributions from an LDA model (Blei et al., 2003) between questions and chunks according to their symmetric Kullback–Leibler divergence. Finally, we also consider the cosine similarity of TF-IDF representations. We found that this approach led to the best performance of the subsequently applied model on the validation set, irrespective of the number of chunks. Note that we used the answer as the query on the training, and the question for the validation and test.

Given the retrieved chunks, we experimented with several neural models using them as context. The AS Reader, which was the better-performing model on the summaries task, underperforms the simple no-context Seq2Seq baseline (shown in Table 5) in terms of MRR. While it does slightly better on the other metrics, it clearly fails to make use of the retrieved context to gain a distinctive margin over the no-context Seq2Seq model. Increasing the number of retrieved chunks, and thereby recall of possibly relevant parts of the story, had only a minor positive effect. The span prediction model—which here also uses selected chunks for context—does especially poorly in this setup. While this model provided the best neural results on the summaries task, we suspect that its performance was particularly badly hurt by the fact that there is so little lexical and grammatical overlap between the source of the questions (summaries) and the context provided (stories). As with the AS Reader, we observed no significant differences for varying numbers of chunks.

These results leave a large gap in human performance, highlighting the success of our design objective to build a task that is realistic and straightforward for humans while very difficult for current reading comprehension models.

Title: Armageddon 2419 A.D.
Question: In what year did Rogers awaken from his deep slumber?
Answer: 2419
Summary snippet: . . . Rogers remained in sleep for 492 years. He awakes in 2419 and,. . .
Story snippet: I should state therefore, that I, Anthony Rogers, am, so far as I know, the only man alive whose normal span of eighty-one years of life has been spread over a period of 573 years. To be precise, I lived the first twenty-nine years of my life between 1898 and 1927; the other fifty-two since 2419. The gap between these two, a period of nearly five hundred years, I spent in a state of suspended animation, free from the ravages of katabolic processes, and without any apparent effect on my physical or mental faculties. When I began my long sleep, man had just begun his real conquest of the air. . .

Figure 2: Example question–answer pair with snippets from the summary and the story.

6 Qualitative Analysis and Challenges

We find that the proposed dataset meets the desiderata we set out in Section 3.1. In particular, we constructed a dataset with a number of long documents, characterized by good lexical coverage and diversity. The questions and answers are human generated and natural sounding; and, based on a small manual examination of ‘Ghostbusters II’, ‘Airplane’, ‘Jacob’s Ladder’, only a small number of questions and answers are shallow paraphrases of sentences in the full document. Most questions require reading segments at least several paragraphs long, and in some cases even multiple segments spread throughout the story.

Computational challenges identified in Section 5.3 naturally suggest a retrieval procedure as the first step. We found that the retrieval is challenging, even for humans not familiar with the presented narrative. In particular, the task often requires referring to larger parts of the story, in addition to knowing at least some background about entities. This makes the search procedure, based on only a short question, a challenging and interesting task in itself.

We show example question–answer pairs in Figures 1, 2, 3. These examples were chosen from a small set of manually annotated question–answer pairs to be representative of this collection. In partic-

ular, the examples show that larger parts of the story are required to answer questions. Figure 3 shows that while the relevant paragraph depicting the injury appears early on, it is not until the next snippet (which appears at the end of the narrative) that the lethal consequences of the injury are revealed. This illustrates an iterative reasoning process as well as extremely long temporal dependencies that we encountered during manual annotation. As shown in Figure 1, reading comprehension on movie scripts requires an understanding of the written dialogue. This is a challenge as dialogue is typically non-descriptive, whereas the questions were asked based on descriptive summaries, requiring models to "read between the lines".

We expect that understanding narratives as complex as those presented in NarrativeQA will require transferring text understanding capability from other supervised learning tasks.

7 Related Work

This paper is the first large-scale question answering dataset on full-length books and movie scripts. However, although we are the first to look at the QA task, learning to understand books through other modeling objectives has become an important subproblem in NLP. These include high level plot understanding through clustering of novels (Frermann and Szarvas, 2017) or summarization of movie scripts (Gorinski and Lapata, 2015), to more fine grained processing by inducing character types (Bamman et al., 2014b; Bamman et al., 2014a), understanding relationships between characters (Iyyer et al., 2016; Chaturvedi et al., 2017), or understanding plans, goals, and narrative structure in terms of abstract narratives (Schank and Abelson, 1977; Wilensky, 1978; Black and Wilensky, 1979; Chambers and Jurafsky, 2009). In computer vision, the MovieQA dataset (Tapaswi et al., 2016) fulfills a similar role as NarrativeQA. It seeks to test the ability of models to comprehend movies via question answering, and part of the dataset includes full length scripts.

8 Conclusion

We have introduced a new dataset and a set of tasks for training and evaluating reading comprehension systems, borne from an analysis of the limitations

Title: Jacob's Ladder
Question: What is the fatal injury that Jacob sustains which ultimately leads to his death ?
Answer: A bayonete stabbing to his gut.
Summary snippet: A terrified Jacob flees into the jungle, only to be bayoneted in the gut by an unseen assailant.
[. . .]
In a wartime triage tent in 1971, military doctors fruitlessly treating Jacob reluctantly declare him dead
Story snippet: As he spins around one of the attackers jams all eight inches of his bayonet blade into Jacob's stomach. Jacob screams. It is a loud and piercing wail.
[. . .]
Int. Vietnam Field Hospital - Day
A doctor leans his head in front of the lamp and removes his mask. His expression is somber. He shakes his head. His words are simple and final.
DOCTOR
He's gone.
Cut to Jacob Singer . . .
The doctor steps away. A nurse rudely pulls a green sheet up over his head. The doctor turns to one of the aides and throws up his hands in defeat.

Figure 3: Example question–answer pair with snippets from the summary and the story.

of existing datasets and tasks. While our QA task resembles tasks provided by existing datasets, it exposes new challenges because of its domain: fiction. Fictional stories—in contrast to news stories—are self-contained and describe a richer set of entities, events, and the relations between them. We have a range of tasks, from simple (which require models to read *summaries* of books and movie scripts, and generate or rank fluent English answers to human-generated questions) to more complex (which require models to read the full *stories* to answer the questions, with no access to the summaries).

In addition to the issue of scaling neural models to large documents, the larger tasks are significantly more difficult as questions formulated based on one or two sentences of a summary might require appealing to possibly discontiguous sentences or paragraphs from the source text. This requires potential solutions to these tasks to jointly model the process of searching for information (possibly in several steps) to serve as support for generating an answer, alongside the

process of generating the answer entailed by said support. End-to-end mechanisms for both searching for information, such as attention, do not scale beyond selecting words or n-grams in short contexts such as sentences and small documents. Likewise, neural models for mapping documents to answers, or determining entailment between supporting evidence and a hypothesis, typically operate on the scale of sentences rather than sets of paragraphs.

We have provided baseline and benchmark results for both sets of tasks, demonstrating that while existing models give sensible results out of the box on summaries, they do not get any traction on the book-scale tasks. Having given a quantitative and qualitative analysis of the difficulty of the more complex tasks, we suggest research directions that may help bridge the gap between existing models and human performance. Our hope is that this dataset will serve not only as a challenge for the machine reading community, but as a driver for the development of a new class of neural models which will take a significant step beyond the level of complexity which existing datasets and tasks permit.

References

Ondrej Bajgar, Rudolf Kadlec, and Jan Kleindienst. 2016. Embracing data abundance: Booktest dataset for reading comprehension. *CoRR*, arXiv:1610.00956.

David Bamman, Brendan O'Connor, and Noah A. Smith. 2014a. Learning latent personas of film characters. In *Proceedings of the Annual Meeting of the Association for Computational Linguistics*, page 352.

David Bamman, Ted Underwood, and Noah A. Smith. 2014b. A Bayesian mixed effects model of literary character. In *Proceedings of the 52nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 370–379.

John B. Black and Robert Wilensky. 1979. An evaluation of story grammars. *Cognitive Science*, 3(3):213–229.

David M. Blei, Andrew Y. Ng, and Michael I. Jordan. 2003. Latent dirichlet allocation. *Journal of Machine Learning Research*, 3(Jan):993–1022.

Nathanael Chambers and Dan Jurafsky. 2009. Unsupervised learning of narrative schemas and their participants. In *Proceedings of the Joint Conference of the 47th Annual Meeting of the ACL and the 4th International Joint Conference on Natural Language Processing of the AFNLP: Volume 2 - Volume 2*, ACL '09, pages 602–610.

Snigdha Chaturvedi, Mohit Iyyer, and Hal Daumé III. 2017. Unsupervised learning of evolving relationships between literary characters. In *Association for the Advancement of Artificial Intelligence*.

Danqi Chen, Jason Bolton, and Christopher D. Manning. 2016. A thorough examination of the CNN/Daily Mail reading comprehension task. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics*.

Michael Denkowski and Alon Lavie. 2011. Meteor 1.3: Automatic Metric for Reliable Optimization and Evaluation of Machine Translation Systems. In *Proceedings of the EMNLP 2011 Workshop on Statistical Machine Translation*, pages 85–91.

Matthew Dunn, Levent Sagun, Mike Higgins, Ugur Guney, Volkan Cirik, and Kyunghyun Cho. 2017. SearchQA: A new Q&A dataset augmented with context from a search engine. *arXiv:1704.05179v2*.

Lea Frermann and György Szarvas. 2017. Inducing semantic micro-clusters from deep multi-view representations of novels. In *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing*, pages 1874–1884.

Philip John Gorinski and Mirella Lapata. 2015. Movie script summarization as graph-based scene extraction. In *Proceedings of the 2015 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 1066–1076, May–June.

Karl Moritz Hermann, Tomáš Kočiský, Edward Grefenstette, Lasse Espeholt, Will Kay, Mustafa Suleyman, and Phil Blunsom. 2015. Teaching machines to read and comprehend. In *Advances in Neural Information Processing Systems*, pages 1693–1701.

Daniel Hewlett, Alexandre Lacoste, Llion Jones, Illia Polosukhin, Andrew Fandrianto, Jay Han, Matthew Kelcey, and David Berthelot. 2016. WikiReading: A novel large-scale language understanding task over wikipedia. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1535–1545.

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

Mohit Iyyer, Anupam Guha, Snigdha Chaturvedi, Jordan Boyd-Graber, and Hal Daumé III. 2016. Feuding families and former friends: Unsupervised learning for dynamic fictional relationships. In *Proceedings of the 2016 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 1534–1544.

Rudolf Kadlec, Martin Schmid, Ondřej Bajgar, and Jan Kleindienst. 2016. Text understanding with the attention sum reader network. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics*, pages 908–918.

Chin-Yew Lin. 2004. ROUGE: A package for automatic evaluation of summaries. In *Proc. ACL Workshop on Text Summarization Branches Out*.

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

Kishore Papineni, Salim Roukos, Todd Ward, and Wei-Jing Zhu. 2002. BLEU: A method for automatic evaluation of machine translation. In *Proceedings of the 40th Annual Meeting on Association for Computational Linguistics*, pages 311–318.

Jeffrey Pennington, Richard Socher, and Christopher D. Manning. 2014. GloVe: Global vectors for word representation. In *Processing of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 1532–1543.

Pranav Rajpurkar, Jian Zhang, Konstantin Lopyrev, and Percy Liang. 2016. SQuAD: 100,000+ questions for machine comprehension of text. In *Proceedings of Empirical Methods in Natural Language Processing (EMNLP)*.

Matthew Richardson, Christopher JC Burges, and Erin Renshaw. 2013. MCTest: A challenge dataset for the open-domain machine comprehension of text. In *Proceedings of the 2013 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 193–203.

Roger C. Schank and Robert P. Abelson. 1977. *Scripts, Plans, Goals and Understanding: an Inquiry into Human Knowledge Structures*. L. Erlbaum, Hillsdale, NJ.

Minjoon Seo, Aniruddha Kembhavi, Ali Farhadi, and Hannaneh Hajishirzi. 2016. Bidirectional attention flow for machine comprehension. *arXiv:1611.01603*.

Ilya Sutskever, Oriol Vinyals, and Quoc V Le. 2014. Sequence to sequence learning with neural networks. In *Proceedings of Conference on the Advances in Neural Information Processing Systems*, pages 3104–3112.

Makarand Tapaswi, Yukun Zhu, Rainer Stiefelhagen, Antonio Torralba, Raquel Urtasun, and Sanja Fidler. 2016. MovieQA: Understanding stories in movies through question-answering. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 4631–4640.

Adam Trischler, Tong Wang, Xingdi Yuan, Justin Harris, Alessandro Sordoni, Philip Bachman, and Kaheer Suleman. 2016. NewsQA: A machine comprehension dataset. *CoRR*, arXiv:1611.09830.

Shuohang Wang and Jing Jiang. 2016. Machine comprehension using match-LSTM and answer pointer. *arXiv:1608.07905*.

Wenhui Wang, Nan Yang, Furu Wei, Baobao Chang, and Ming Zhou. 2017. Gated self-matching networks for reading comprehension and question answering. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistcs*, pages 189–198.

Dirk Weissenborn, Georg Wiese, and Laura Seiffe. 2017. FastQA: A simple and efficient neural architecture for question answering. *CoRR*, arXiv:1703.04816v1.

Jason Weston, Antoine Bordes, Sumit Chopra, and Tomas Mikolov. 2015. Towards AI-complete question answering: A set of prerequisite toy tasks. *CoRR*, arXiv:1502.05698.

R. Wilensky. 1978. Why John married Mary: Understanding stories involving recurring goals. *Cognitive Science*, 2(3):235–266.