A Dataset of Information-Seeking Questions and Answers Anchored in Research Papers

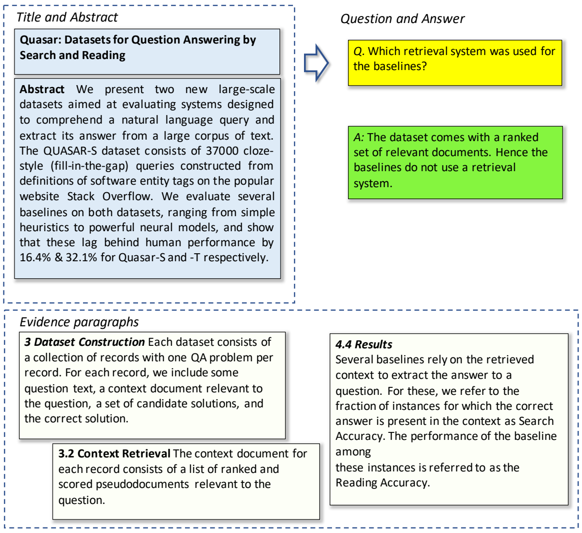
Pradeep Dasigi[♣] **Kyle Lo**[♣] **Iz Beltagy**[♣] **Arman Cohan**[♣]
Noah A. Smith[♢♣] **Matt Gardner**[♣]
[♣]Allen Institute for AI [♢]Paul G. Allen School of CSE, University of Washington
{pradeepd,kylel,beltagy,armanc,noah,mattg}@allenai.org

Abstract

Readers of academic research papers often read with the goal of answering specific questions. Question Answering systems that can answer those questions can make consumption of the content much more efficient. However, building such tools requires data that reflect the difficulty of the task arising from complex reasoning about claims made in multiple parts of a paper. In contrast, existing information-seeking question answering datasets usually contain questions about generic factoid-type information. We therefore present QASPER, a dataset of 5,049 questions over 1,585 Natural Language Processing papers. Each question is written by an NLP practitioner who read only the title and abstract of the corresponding paper, and the question seeks information present in the full text. The questions are then answered by a separate set of NLP practitioners who also provide supporting evidence to answers. We find that existing models that do well on other QA tasks do not perform well on answering these questions, underperforming humans by at least 27 F_1 points when answering them from entire papers, motivating further research in document-grounded, information-seeking QA, which our dataset is designed to facilitate.

1 Introduction

Machines built to assist humans who engage with texts to seek information ought to be designed with an awareness of the information need. Abstractly, the human's need should define the lens through which the system views the text in order to find desired information. Existing information-seeking machine reading datasets (e.g., Kwiatkowski et al., 2019; Clark et al., 2020) have led to significant progress in reading at scale (e.g., Asai et al., 2020; Guu et al., 2020; Liu et al., 2020). However, most of those benchmarks focus on an "open domain" setting where the questions are not anchored in any particular user context. The result is an emphasis on generic factoid questions, rather than the full range of information needs people have.



Quasar: Datasets for Question Answering by Search and Reading

Abstract We present two new large-scale datasets aimed at evaluating systems designed to comprehend a natural language query and extract its answer from a large corpus of text. The QUASAR-S dataset consists of 37000 cloze-style (fill-in-the-gap) queries constructed from definitions of software entity tags on the popular website Stack Overflow. We evaluate several baselines on both datasets, ranging from simple heuristics to powerful neural models, and show that these lag behind human performance by 16.4% & 32.1% for Quasar-S and -T respectively.

Q. Which retrieval system was used for the baselines?

A: The dataset comes with a ranked set of relevant documents. Hence the baselines do not use a retrieval system.

3 Dataset Construction Each dataset consists of a collection of records with one QA problem per record. For each record, we include some question text, a context document relevant to the question, a set of candidate solutions, and the correct solution.

3.2 Context Retrieval The context document for each record consists of a list of ranked and scored pseudodocuments relevant to the question.

4.4 Results
Several baselines rely on the retrieved context to extract the answer to a question. For these, we refer to the fraction of instances for which the correct answer is present in the context as Search Accuracy. The performance of the baseline among
these instances is referred to as the Reading Accuracy.

Figure 1: An example instance taken from QASPER. A **question** about the paper is written after reading only the title and the abstract. To arrive at the **answer**, one finds relevant **evidence**, which can be spread across multiple paragraphs. In this example, to answer the question about "baselines", the reader must realize from evidence from Sections 3 and 4 that "context documents" come pre-ranked in the dataset and the paper's "baselines" select from these "context documents."

We present QASPER,[1] an information-seeking question answering (QA) dataset over academic research papers. Each question is written as a follow-up to the title and abstract of a particular paper, and the answer, if present, is identified in the rest of the paper, along with evidence required to arrive at it. This setup results in questions requiring more complex document-level reasoning than prior datasets, because *(i)* abstracts provide rich prompts for questions that can be asked as follow-up and *(ii)* academic research papers naturally trigger ques-

[1] Loosely derived from *Question Answering over Scientific Research Papers*. The dataset, baseline code, and other information about the project can be found at https://allenai.org/project/qasper.

Proceedings of the 2021 Conference of the North American Chapter of the
Association for Computational Linguistics: Human Language Technologies, pages 4599–4610
June 6–11, 2021. ©2021 Association for Computational Linguistics

tions by their target readers that require supporting or refuting claims. This evidence may be spread across the paper, including tables and figures, often resulting in complex entailment problems. The example in Figure 1 illustrates one such case where we need to retrieve information from paragraphs in three different sections to answer the question.

QASPER contains 5,049 questions over 1,585 natural language processing (NLP) papers, asked by regular readers of NLP papers, and answered by a separate set of NLP practitioners. Each paper has an average of 3.2 questions, up to a maximum of 12 questions for a single paper. In addition to providing answers when the questions are answerable, the annotators were asked to select text, tables, or figures as evidence required for answering the questions. 55.5% of the questions require evidence from multiple paragraphs in the paper and 13% require tables or figures. To the best of our knowledge, QASPER is the first QA dataset in the academic research domain focusing on entire papers, and not just abstracts.

To quantify the difficulty of the tasks in QASPER, we apply state-of-the-art document-level Transformer (Vaswani et al., 2017) models to the tasks of selecting evidence and generating answers, and show that the best model performance lags behind humans by 27 F_1 points at answering questions from entire papers, and 32 F_1 points at selecting the paragraphs that provide evidence to answer the questions, indicating that these are both unsolved problems. Additionally, we experiment with oracles that answer questions from gold evidence and find that better pretraining and domain-adaptation might be helpful.

2 Building the QASPER Dataset

We now describe our process for constructing the dataset. We began with a set of open-access NLP papers, recruited NLP practitioners who are regular readers of research papers, and designed two different data collection interfaces: one for collecting follow-up questions given titles and abstracts, and another for obtaining evidence and answers to those questions.

2.1 Papers

We filtered S2ORC (Lo et al., 2020),[2] a collection of machine-readable full text for open-access papers, to *(i)* those from arXiv with an associated LaTeX source file,[3] and *(ii)* are in the computational linguistics domain.[4] We limited our domain to computational linguistics to ensure high quality as we have access to realistic users through our research network; broader domain collection is left to future work and should be enabled by the proof-of-concept of our protocols given in this paper. We used the S2ORC parser (which normalizes multi-file LaTeX sources and resolves comments and macros) to convert LaTeX markup to full text while preserving section and paragraph breaks and math equations. We supplemented the paper text with extracted images of figures and tables associated with their captions; these were crawled from Semantic Scholar.[5] The result of this process was a collection of 18K full text papers for annotation.

2.2 Decoupled Data Collection

To ensure that our questions are realistic, we decoupled the question-writing and question-answering phases. For both tasks we recruited graduate students studying NLP and freelancers practicing NLP through professional networks and Upwork[6]. All the workers were regular readers of NLP papers, and were paid US$25 per hour on average ($20-$40 based on experience). We paid them on a per-hour basis and not a per-question basis to prioritize data quality over quantity. A total of 25 workers wrote questions while 51 answered them.

Questions To ensure that annotators were actually interested in the paper they are reading, we provided them with a lightweight search interface to search papers from the aforementioned collection to focus on their papers of interest. The interface supports entering manual queries and examples of the queries annotators used include general (e.g., "computer vision") or specific (e.g., "question answering", "information extraction") areas of study, specific tasks (e.g., "language identification"), entities (e.g., "bert", "transformers") or concepts (e.g., "commonsense", "interpretability"), or domain specifications (e.g., "medical", "wikipedia"). Annotators also had the option to not enter any search queries; in this case, they were shown random papers. Annotators were displayed only the title and abstracts of relevant papers and asked to

[2] We accessed both release versions `20190928` and `20200705v1`.

[3] LaTeX allows us to avoid quality issues with PDF parsing.

[4] We chose those either tagged with the `cs.CL` arXiv category or published with an ACL Anthology identifier.

[5] http://semanticscholar.org

[6] https://www.upwork.com/

write any number of questions they had about the paper. Annotators were instructed to only write questions that are *not* answerable from the title and abstract but expected to be answered somewhere in the paper. Annotators also provided basic information about their expertise in NLP and how familiar they already were with the paper for which they asked questions. Most workers (about 70%) had some experience in NLP, with 20% having more than five years of experience. A vast majority (94%) of the abstracts were seen by the question-writers for the first time.

Answers Annotators were randomly assigned papers with all the corresponding questions written for that paper. They were shown the paper title, abstract, question, full text, and all associated figures and tables to answer the questions. After reading these, annotators were were asked to:

- Make a binary decision as to whether the question is answerable given the paper.
- If the question is answerable, select the minimal set of *evidence* snippets that contains the answer to the question. This could be (possibly discontiguous) paragraphs from the text and/or figures or tables. Annotators were asked to prioritize text over figures and tables, unless the information required was present only in figures or tables. When multiple paragraphs could serve as evidence, annotators were asked to first prioritize evidence that adequately answered the question, and then paragraphs that occurred earlier in the text.
- If the question is answerable, also provide a concise *answer* to the question. Annotators were also asked to also indicate whether their concise answer was *(i)* extracted from the evidence, *(ii)* "yes" or "no", or *(iii)* abstractively written.

Annotators were allowed to skip any questions they did not feel comfortable answering. Since the answering task is significantly more complex than the question-writing task, we designed interactive tutorials and qualification exams for the workers for this task using CrowdAQ (Ning et al., 2020). Workers who scored well were invited to work on the task. If the test performance indicated that the workers did not have sufficient NLP knowledge, or were not used to reading papers we did not let them work on the task. In cases where the workers misunderstood the task, but had sufficient background knowledge, we provided additional training before letting them work on the task.

3 QASPER Analysis

Table 1 provides representative examples from QASPER categorized by question, answer, and evidence types, which we describe here in greater detail.

Question types We first analyze whether our annotation setup results in questions that are anchored in the context of the papers. To answer this question, we manually[7] categorized a set of 200 questions as being applicable to most papers in the domain (general) vs. being applicable only to the paper that the question is written about (specific). Table 1 shows that most of the questions (67%) are specific to the papers they are written about. This result indicates the advantage of viewing the QASPER task as a question answering problem, instead of an information extraction problem since a fixed schema would not be able to handle the long tail of paper-specific information needs.

Answer types As shown in Table 1, most of the answers in the dataset are extractive. The average length of the extractive answers is 14.4 words (including all spans), and that of abstractive spans is 15.6 words.

Evidence types Evidence can include one or more paragraphs from the paper, a figure, or a table, or a combination of these. Table 1 shows the distribution of these types. Among the answerable questions with text-only evidence, 55.5% of the answers have multi-paragraph evidence (Figure 1 is one example). Unanswerable questions do not have any evidence. Among the answerable ones, (3.0%) have no evidence when the answer is *No*, and the evidence is the *lack* of a mention of something specific. The last question in Table 4 is one example of such a case.

Distribution of evidence paragraphs We perform an analysis to identify the main sections of a paper that contain textual evidence. We assign each evidence paragraph to its containing top-level[8]

[7] Two domain-experts independently judged these, and achieved a Cohen's κ of 0.94.

[8] S2ORC provides section hierarchy derived from LaTeX source

Question	Type	%	Paper(s)
What datasets do they use?	General	33.3%	1; 2; 3
What other political events are included in the database?	Specific	66.7%	1706.01875

Question	Answer	Type	%	Paper
What five dialogue attributes were analyzed?	Model; Confidence; Continuity; Query-relatedness; Repetitiveness; Specificity	Extractive	51.8%	1705.00571
Which neural architecture do they use as a base for their attention conflict mechanisms?	GRU-based encoder, interaction block, and classifier consisting of stacked fully-connected layers.	Abstractive	24.2%	1906.08593
Do they ensure the that the architecture is differentiable everywhere after adding the Hungarian layer?	Yes	Yes/No	13.9%	1712.02555
What language are the captions in?	N/A	Unanswer.	10.2%	1909.09070

Question	Evidence	Type	%	Paper
What new tasks do they use to show the transferring ability of the shared meta-knowledge?	To test the transferability of our learned Meta-LSTM, we also design an experiment, in which we take turns choosing 15 tasks to train our model with multi-task learning, then the learned Meta-LSTM are transferred to the remaining one task. The parameters of transferred Meta-LSTM, $\theta_m^{(s)}$ in Eq.(33), are fixed and cannot be updated on the new task.	Text	81.6%	1802.08969
How much does it minimally cost to fine-tune some model according to benchmarking framework?	Table 1	Table/Figure	11.6%	2002.05829
Do they recommend translating the premise and hypothesis together?	N/A	None	12.8%	2004.04721

Table 1: Examples of questions (top), answers (middle), and evidence (bottom) sampled from QASPER. % are relative frequencies of the corresponding type over all examples in QASPER. The percentages for evidence types sum over 100% due to double-counting of 446 answers with both Table/Figure and Text evidence.

section, and perform some section name normalization. We find that among the frequently used section names such as "Experiments" and "Introduction," there was not a single section name that contained a majority of evidence spans, indicating that the distribution of evidence over section in the paper was more or less uniform.

Inter-annotator agreement 44% of the questions in QASPER have multiple annotated answers. On average, each question is answered by 1.6 annotators (up to a maximum of 6 annotators for the same question). Using these multiple annotations, we compute some measures of agreement between annotators. First, we found that there is a high level of agreement (90%) regarding answerability of questions. Second, we find that annotators agreed on the type of the evidence (text vs. figure) in 84.0% of the cases. Papers often provide the same information both in tables and text, and agreement over the evidence types could be a consequence of our clear annotation guidelines regarding selecting evidence.

Correctness To estimate the correctness of the answer annotations in QASPER, we manually analyzed 100 randomly sampled questions with multiple answer annotations (averaging 2.73 answers per question). We found that 207 (75.8%) of the answers were correct. 98% of the questions had at least one correct answer, and 77% had most of the answers correct.

4 Modeling QASPER

This section explains the task, evaluation metrics, and a model addressing QASPER tasks.

4.1 Task Setup

We formally define the QASPER tasks as follows: Given a paper, and a question about it, the primary task is to determine if the question is answerable, and output a predicted answer, that is one or more spans in the full-text of the paper, *yes*, *no* or other free-form text. A system built for this will be evaluated based on the correctness of the predicted answer measured against the reference answers. Since QASPER also provides labeled evidence for all questions, the system may also use auxiliary supervision provided by the evidence.

One such auxiliary task is to predict the evidence required for the question. The inputs are the same as that of the primary task, but the outputs are expected to be one or more paragraphs in the full-text, figures, or tables, and they will be evaluated against labeled evidence spans.

Evaluation metrics As an automatic proxy for the measure of correctness of all types of answers, we use the span-level F_1 measure proposed by Rajpurkar et al. (2016). We convert answers that are multiple selected spans into single comma-separated strings. For questions with multiple reference answers, we compute the max span-F_1 of the predictions over all the references. We evaluate the performance of a system over the auxiliary task by computing a F_1 score over the set of paragraphs, figures, and tables chosen by the system against the reference evidence, considering a max when there are multiple references. We refer to these metrics as Answer-F_1 and Evidence-F_1, respectively.

Data splits We split the dataset into train, validation, and test sets, so that each paper appears in only one of them. Our analysis of correctness of annotations presented in Section 3 indicates a high likelihood (98%) of evaluating against a correct reference when evaluation is aggregated over multiple references. Hence we ensure that most of the questions in validation and test sets have multiple references (98% in test, and 74% in validation). This resulted in 2,593, 1,005, and 1,451 questions in the three sets, respectively.

Estimating human performance To estimate an upper bound on model performance given our data splits and metrics, we assess the performance of the workers when evaluated against each other using the same metrics on a sample of the test set. Since model performance is evaluated by aggregating over multiple references, we consider a subset of the test set containing questions with at least three references (40% of the test set), evaluate each reference against the remaining, and compute an average over all such combinations. This procedure estimates the human performance to be 60.9 Answer-F_1, and 71.6 Evidence-F_1. Note that given the disagreements among the workers estimated in Section 3, this is a lower bound on human performance for two reasons: first, because only two annotations are used to compute the metric, while systems are evaluated against all three; and second, because the annotators are NLP practitioners, not expert researchers, and it is likely that an expert would score higher. Hence we report these numbers, along with a breakdown over answer types in Table 2 and Table 3 as human performance lower bounds.

4.2 QASPER Model

We base our model on pretrained Transformer (Vaswani et al., 2017) models which currently produce state-of-the-art results on a majority of QA tasks.[9] Recall that QASPER introduces two main modeling challenges – different answer types and long input documents. First, QASPER includes a variety of answer types, including extractive, abstractive, yes/no, and unanswerable questions, which means a typical span-selection BERT-based QA model (Devlin et al., 2019) is not sufficient to support all these answer types. We address this by converting all answer types into a single task: generating answer text (Raffel et al., 2020; Khashabi et al., 2020).[10] This is a sequence-to-sequence formulation that requires an encoder-decoder Transformer model where the encoder reads the question and the document and the decoder generates the answer text.

Second, research papers are much longer than the typical 512 or 1024 token limit of most BERT-like models, so we need a Transformer model that can process long inputs. We use the Longformer-Encoder-Decoder (LED; Beltagy et al., 2020), an encoder-decoder Transformer model that can efficiently process input sequences thousands of tokens long. With LED's support for input sequence length of 16K tokens, we can encode 99% of the paper full texts in the QASPER dataset without truncation.

Longformer-Encoder-Decoder (LED) LED (Beltagy et al., 2020) is a variant of the original Transformer encoder-decoder model that replaces the Transformer's full self-attention in the encoder with the efficient local+global attention pattern

[9] https://paperswithcode.com/task/question-answering

[10] We tried a model that predicts answer type, then based on the type uses a different head to predict the corresponding answer. This model performed much worse than the proposed seq2seq formulation.

of Longformer. This allows each token to attend to only its local window and a pre-specified set of global locations of interest, thereby scaling self-attention computation linearly with the input size (as opposed to quadratically with full context self-attention). LED has a similar architecture to BART (Lewis et al., 2020) in terms of number of layers and hidden state sizes, with the distinction that it has a larger position embeddings matrix, allowing it to process inputs of up to 16K tokens long (up from 1K tokens in the original BART model). In practice, LED's parameters are initialized from a pretrained BART model, and LED copies BART's position embeddings 16 times to fill the entire 16K position embeddings matrix. For all experiments we use the LED-base sized model, which uses BART-base weights.

Input and Output Encoding For the input, we follow the Longformer QA models (Beltagy et al., 2020) and encode the question and context in one concatenated string with "global attention" over all the question tokens. For the output, all answer types are encoded as single strings. The string is the text of the abstractive answer, a comma separated concatenation of the extractive spans, "Yes", "No", or "Unanswerable".

Evidence extraction To support extracting evidence paragraphs, we prepend each paragraph with a `</s>` token and add a classification head over these tokens on LED's encoder side. We also add Longformer's global attention over these tokens to facilitate direct information flow across the paragraphs. We then train LED using both loss functions (teacher-forced text generation and paragraph classification) in a multi-task training setup. For the answer generation, we use a cross-entropy loss function over the vocabulary. For the evidence paragraph extraction, we use a cross-entropy loss function with binary 0 or 1 gold labels for evidence/non-evidence paragraph. To account for class imbalance, we use loss scaling with weights proportional to the ratio of positive to negative gold paragraphs in the batch, which we found to be crucial for the model to train. One benefit of multi-task training of evidence extraction along with answer selection is that tasks can benefit each other (see Section 5.2).

5 Experiments

We evaluate model performance on question answering and evidence selection tasks, and compare them to estimated lower bounds on human performance. These human performance estimates are calculated by comparing the answers of questions for which we have multiple human annotations. For each question, we choose one annotation as if it were a prediction, and evaluate it against the rest of the annotations, and consider as human performance the average over all annotations chosen as predictions. We restrict our experiments to the subset of questions in QASPER that can be answered from text in the paper, ignoring those that require figures or tables as evidence (13% of the dataset; see Section 3) to avoid having to deal with multimodal inputs. We leave multimodal question answering to future work.

5.1 Training Details

We train all models using the Adam optimizer (Kingma and Ba, 2014) and a triangular learning rate scheduler (Howard and Ruder, 2018) with 10% warmup. To determine number of epochs, peak learning rate, and batch size, we performed manual hyperparameter search on a subset of the training data. We searched over $\{1, 3, 5\}$ epochs with learning rates $\{1e^{-5}, 3e^{-5}, 5e^{-5}, 9e^{-5}\}$, and found that smaller batch sizes generally work better than larger ones. Our final configuration was 10 epochs, peak learning rate of $5e^{-5}$, and batch size of 2, which we used for all reported experimental settings. When handling full text, we use gradient checkpointing (Chen et al., 2016) to reduce memory consumption. We run our experiments on a single RTX 8000 GPU, and each experiment takes 30–60 minutes per epoch.

5.2 Results

Question answering Table 2 shows the overall performance of the LED-base model[11] on question answering, as well as the performance breakdown on the different answer types. The table also compares LED-base variants when the input is heuristically limited to smaller parts of the paper (i.e., no context, abstract, introduction). We generally observe that, by using more context, the performance improves. Specifically, as we observe in row 5 encoding the entire context results in significant overall performance improvement ($\Delta = +9.5$) over the best heuristic ("introduction"). This signifies the importance of encoding the entire paper. Comparing rows 4 and 5, we observe that using the

[11] We trained an LED-large model as well, but it performed much worse than the base model on the QA task.

Input	Extractive		Abstractive		Yes/No		Unanswerable		Overall	
	Dev.	Test	Dev.	Test	Dev.	Test	Dev.	Test	Dev.	Test
Q only	4.60	5.91	6.06	7.38	69.05	66.36	58.43	66.67	17.81	22.48
Q+Abstract	6.69	7.97	7.50	8.25	69.05	63.43	51.14	62.50	18.60	22.30
Q+Introduction	4.40	6.60	2.52	3.16	65.87	67.28	71.00	78.07	18.30	24.08
Q+Full Text	26.07	30.96	16.59	15.76	67.48	70.33	28.57	26.21	29.05	32.80
Q+Full Text w/ scaff.	24.62	29.97	13.86	15.02	63.64	68.90	38.89	44.97	28.01	33.63
Human (lower bound)	-	58.92	-	39.71	-	78.98	-	69.44	-	60.92

Table 2: LED-base and lower-bound human performance on answering questions in QASPER, measured in Answer-F_1. The top three rows are heuristic baselines that try to predict answers without encoding entire papers. *w/ scaff.* refers to the inclusion of the evidence selection scaffold during training.

evidence prediction as a multi-task scaffolding objective helps, improving the results by $\Delta = +0.8$ points.

Evidence selection Table 3 illustrates the evidence selection performance of the LED-large and LED-base models compared with simpler baselines. We observe that LED variants outperform the simple TF-IDF baseline but there still remains a large gap to human performance.

Varying amounts of training Figure 2 shows the learning curve that measures the validation Answer-F_1 and Evidence-F_1 of the LED-base variants based on training data size. The learning curve suggests that performance has not reached a plateau, and future data collection could be useful.

Answer prediction from gold evidence To better isolate the question answering (as opposed to evidence selection) task performance, we perform oracle experiments where models are given the gold evidence. For these experiments, we are able to use larger (T5-large; Raffel et al., 2020) or better task-adapted pretrained models (UnifiedQA-large; Khashabi et al., 2020), which perform significantly better in the oracle setting. We did not use them in the non-oracle setting, however, as Longformer versions of these models are not available, and LED's ability to handle the full document without the need for a pipelined retrieval system was more important. These experiments show that (1) the human lower bound is in fact a lower bound, as large models exceed it for span answers in this setting; (2) the majority of the large headroom in the non-oracle setting can be closed with better evidence selection; and (3) research into making large pretrained models able to better scale to long documents would be beneficial.

Model	Evidence F_1	
	Dev.	Test
LED-base	23.94	29.85
LED-large	31.25	39.37
TF-IDF	10.68	9.20
Random paragraph	2.09	1.30
First paragraph	0.71	0.34
Human (lower bound)	-	71.62

Table 3: Model and lower-bound human performance on selecting evidence for questions in QASPER

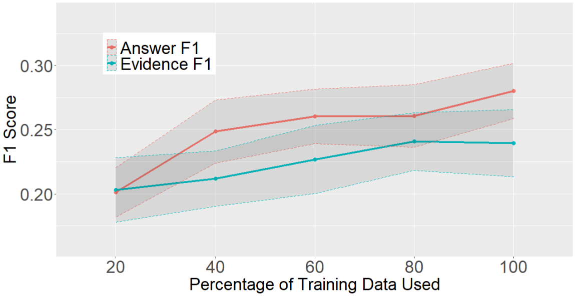


Figure 2: Learning curves showing Answer-F_1 and Evidence-F_1 on the dev. set while varying training data size.

Error analysis To gain insight into the model's errors, we sample 67 test questions with predicted Answer-F_1 scores below 0.10 from the LED model trained with evidence prediction scaffolding. We remove four cases in which the predicted answers are actually correct. Examining gold answers of the remaining 63, we find 31 are extractive, 24 are abstractive, 3 are "yes", 3 are "no," and 2 are unanswerable. We observe that LED often predicts shorter spans than the gold answers (9.5 words shorter than gold counterparts, on average). Focusing only on the 55 questions with either extractive or abstractive gold answers, we manually categorize error types in Table 5.

Model	Answer F_1		
	Span	Abstractive	Overall
LED-base	54.20	24.95	44.96
T5-large	65.59	29.11	60.03
UnifiedQA-large	67.23	28.92	61.39

Table 4: Model performance on the QASPER test set on answering questions given gold evidence. We do not show performance on *Yes/No* and *Unanswerable* types because they can be trivially predicted to a large extent from the absence of gold evidence.

6 Related Work

Information-Verifying QA A large body of work on question answering follows the *information-verifying* paradigm where the writer of the question already knows its answer, and the questions are written solely for evaluating the knowledge or understanding capabilities of machines. Some examples include SQuAD (Rajpurkar et al., 2016), TriviaQA (Joshi et al., 2017), NarrativeQA (Kočiský et al., 2018), WikiHop (Welbl et al., 2018), HotpotQA (Yang et al., 2018), CoQA (Reddy et al., 2019), DROP (Dua et al., 2019), QUOREF (Dasigi et al., 2019). Most datasets for QA on academic research papers also fall within the information-verifying paradigm as they automatically construct QA examples using extracted entities and relations and structured knowledge resources, like DrugBank. Some examples include emrQA (Pampari et al., 2018), BioRead (Pappas et al., 2018), BioMRC (Pappas et al., 2020), MedHop (Welbl et al., 2018). While these datasets enabled significant progress in machine comprehension, they include biases in questions that may not reflect real-world settings (Kwiatkowski et al., 2019).

Information-Seeking QA in General Domain Recognizing this challenge, others have followed an *information-seeking* paradigm where the writer of questions is genuinely interested in finding the answer to the question, or at least does not have access to the answer. Examples of such datasets include WikiQA (Yang et al., 2015), NewsQA (Trischler et al., 2017), MsMarco (Campos et al., 2016), QuAC (Choi et al., 2018), Natural Questions (Kwiatkowski et al., 2019), TyDiQA (Clark et al., 2020), and IIRC (Ferguson et al., 2020). Unlike QASPER, Natural Questions and TyDiQA[12] questions are not grounded in any contexts, and the associated documents are linked to the questions after they are written. In contrast, QASPER's questions are real follow-up questions about a paper that a reader of appropriate domain expertise would have after reading the title and the abstract. The priming lets the readers ask detailed questions that are specific to the papers in context, those that require a deeper understanding of the contexts, like those shown in Figure 1 and Table 1. QuAC used similar data collection method but with focus on entities, which QASPER does not impose.

Domain-Specific Information-seeking QA Some work has been done on information-seeking QA on academic research papers. PubmedQA (Jin et al., 2019) derives Yes/No/Maybe questions from PubMed paper titles answered from the conclusion sections of the corresponding abstracts. BioAsq benchmarks (Balikas et al., 2013; Nentidis et al., 2018; Krallinger et al., 2020) focus on open-domain QA over PubMed abstracts. Like QASPER, BioAsq answers can take different forms (e.g., yes/no, extracted span(s)). QASPER differs from BioAsq in that questions are grounded in a single paper of interest. Furthermore, QASPER uses the paper full text, not just the abstract. To the best of our knowledge, QASPER is the first information-seeking QA dataset in a computer science domain, while most prior work using academic research papers has been in biomedicine. Furthermore, with over 5K annotated questions, QASPER is also larger than other comparable human-annotated QA datasets – PubmedQA and BioAsq contain 1K and 3.2K questions, respectively. Finally, QASPER poses a challenging full document-level task while other related datasets are abstract-level. Beyond the domain of academic research, realistic QA datasets have also been built in the privacy policy domain (Ravichander et al., 2019; Ahmad et al., 2020). These tasks are similar to our evidence selection task.

7 Conclusion

We presented QASPER, an information-seeking QA dataset over NLP research papers. With natural questions asked as follow-up to titles and abstracts, the task presented by QASPER requires evidence from multiple paragraphs and/or figures and tables within the full text of the papers. Our empirical

[12] TyDiQA uses short snippets to prime annotators to write questions of interest, but the annotation process does not require workers to write questions grounded in those snippets.

Error	%	Example question	Gold	Predicted
Incorrectly predicts unanswerable	34.5%	How is the text segmented?	"dividing documents into chunks before processing"	Unanswerable
Lacks domain knowledge	23.6%	What other scenarios can the bias mitigation methods be applied to?	"sentiment analysis, other identity problems like racial"	GRU
		What baselines is the neural relation extractor compared to?	Berant et al. (2013), Yao and Van Durme (2014), Xu et al. (2014), Berant and Liang (2014), ...	Freebase, MCCNN
		†What hand-crafted features are used?	"position of sentence, sentence length, tense, qualifying adjectives, meta-discourse features"	the Minimum word count is 40, The number of threads to run in parallel is 4 and the context window is 10
Uninformative	20.0%	What do they mean by intrinsic geometry of spaces of learned representations?	"the inferred embedding space creates a globally consistent structured prediction of the ontology, rather than local relation predictions"	intrinsic geometry
		How does the proposed training framework mitigate the bias pattern?	by balancing or, smoothing the artifacts across different classes by assigning specific weights for every sample	By minimizing the impact of the bias pattern on the dataset
Not comprehensive	7.3%	Which metrics were considered?	"ter, bleu, rouge, nist, lepor, cider, meteor, Semantic Similarity (sim), readability and grammaticality"	Grammar-based metrics (GBMs)
		Was permission sought from the bipolar patients to use this data?	For Twitter and Reddit users, implicit consent is assumed to use their public tweets. Blog users are contacted to obtain consent for using their texts.	No
Lacks specificity	7.3%	What are the performance metrics?	"Rouge-1, Rouge-2 and Rouge-4 recall"	Rouge scores
		What supervised machine learning models do they use?	"ZeroR, Naïve Bayes, J48, and random forest"	Weka classifiers
Lacks numeracy	7.3%	How many tags are included in the ENE tag set?	"200 fine-grained categories"	1

Table 5: Error analysis of our best model (LED from row 5 from Table 2) on 55 test examples with low F_1 score (excluding those with "yes," "no," or "unanswerable" gold answers). "Quotations" denote extractive gold answers. We note *Lacks domain knowledge* errors are not always solved by better entity type resolution (see †).

results show plenty of room for improvement when compared to the estimated human performance, and suggest that QASPER could serve as a test-bed for evaluating document-grounded QA research.

Ethical Considerations

We present a new dataset that uses papers authored by other researchers. To adhere to copyright, we have restricted ourselves to arXiv papers released under a CC-BY-* license, as identified via Unpaywall, which was used in the S2ORC (Lo et al., 2020) dataset construction. Due to our choice to use arXiv as the source of papers, QASPER is almost entirely an English-language dataset, and QA systems built on QASPER would not be expected to work well on non-English language research papers.

We have determined the amount we paid the annotators to be well-above the minimum wage in our local area. While we do collect information about annotator background in NLP and familiarity with the papers they are annotating, we have not collected personal identifiable information without their permission except for payment purposes, and do not include any such information in the released dataset.

References

Wasi Ahmad, Jianfeng Chi, Yuan Tian, and Kai-Wei Chang. 2020. PolicyQA: A reading comprehension dataset for privacy policies. In *Findings of the Association for Computational Linguistics: EMNLP 2020*, pages 743–749, Online. Association for Computational Linguistics.

Akari Asai, Kazuma Hashimoto, Hannaneh Hajishirzi, Richard Socher, and Caiming Xiong. 2020. Learning to retrieve reasoning paths over wikipedia graph for question answering. In *International Conference on Learning Representations*.

Georgios Balikas, Ioannis Partalas, Aris Kosmopoulos, Sergios Petridis, Prodromos Malakasiotis, Ioannis Pavlopoulos, Ion Androutsopoulos, Nicolas Baskiotis, Eric Gaussier, Thierry Artieres, et al. 2013. Evaluation framework specifications. *Project deliverable D*, 4.

Iz Beltagy, Matthew E. Peters, and Arman Cohan. 2020. Longformer: The long-document transformer. *ArXiv*, abs/2004.05150.

Daniel Fernando Campos, T. Nguyen, M. Rosenberg, Xia Song, Jianfeng Gao, Saurabh Tiwary, Rangan Majumder, L. Deng, and Bhaskar Mitra. 2016. Ms marco: A human generated machine reading comprehension dataset. *ArXiv*, abs/1611.09268.

Tianqi Chen, Bing Xu, Chiyuan Zhang, and Carlos Guestrin. 2016. Training deep nets with sublinear memory cost. *ArXiv*, abs/1604.06174.

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

Jonathan H. Clark, Eunsol Choi, Michael Collins, Dan Garrette, Tom Kwiatkowski, Vitaly Nikolaev, and Jennimaria Palomaki. 2020. TyDi QA: A benchmark for information-seeking question answering in typologically diverse languages. *Transactions of the Association for Computational Linguistics*, 8:454–470.

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

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

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

James Ferguson, Matt Gardner, Hannaneh Hajishirzi, Tushar Khot, and Pradeep Dasigi. 2020. IIRC: A dataset of incomplete information reading comprehension questions. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 1137–1147, Online. Association for Computational Linguistics.

Kelvin Guu, Kenton Lee, Zora Tung, Panupong Pasupat, and Ming-Wei Chang. 2020. Realm: Retrieval-augmented language model pre-training. *ArXiv*, abs/2002.08909.

Jeremy Howard and Sebastian Ruder. 2018. Universal language model fine-tuning for text classification. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 328–339, Melbourne, Australia. Association for Computational Linguistics.

Qiao Jin, Bhuwan Dhingra, Zhengping Liu, William Cohen, and Xinghua Lu. 2019. PubMedQA: A dataset for biomedical research question answering. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 2567–2577, Hong Kong, China. Association for Computational Linguistics.

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

Daniel Khashabi, Sewon Min, Tushar Khot, Ashish Sabharwal, Oyvind Tafjord, Peter Clark, and Hannaneh Hajishirzi. 2020. UNIFIEDQA: Crossing format boundaries with a single QA system. In *Findings of the Association for Computational Linguistics: EMNLP 2020*, pages 1896–1907, Online. Association for Computational Linguistics.

Diederik P Kingma and Jimmy Ba. 2014. Adam: A method for stochastic optimization. *ArXiv*, abs/1412.6980.

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

Martin Krallinger, Anastasia Krithara, A. Nentidis, G. Paliouras, and Marta Villegas. 2020. Bioasq at clef2020: Large-scale biomedical semantic indexing and question answering. *Advances in Information Retrieval*, 12036:550 – 556.

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

Mike Lewis, Yinhan Liu, Naman Goyal, Marjan Ghazvininejad, Abdelrahman Mohamed, Omer Levy, Veselin Stoyanov, and Luke Zettlemoyer. 2020. BART: Denoising sequence-to-sequence pre-training for natural language generation, translation, and comprehension. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 7871–7880, Online. Association for Computational Linguistics.

Dayiheng Liu, Yeyun Gong, Jie Fu, Yu Yan, Jiusheng Chen, Daxin Jiang, Jiancheng Lv, and Nan Duan. 2020. RikiNet: Reading Wikipedia pages for natural question answering. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 6762–6771, Online. Association for Computational Linguistics.

Kyle Lo, Lucy Lu Wang, Mark Neumann, Rodney Kinney, and Daniel Weld. 2020. S2ORC: The semantic scholar open research corpus. In *Proceedings of the Association for Computational Linguistics (ACL)*.

Anastasios Nentidis, Anastasia Krithara, Konstantinos Bougiatiotis, Georgios Paliouras, and Ioannis Kakadiaris. 2018. Results of the sixth edition of the BioASQ challenge. In *Proceedings of the 6th BioASQ Workshop A challenge on large-scale biomedical semantic indexing and question answering*, pages 1–10, Brussels, Belgium. Association for Computational Linguistics.

Qiang Ning, Hao Wu, Pradeep Dasigi, Dheeru Dua, Matt Gardner, Robert L. Logan IV, Ana Marasović, and Zhen Nie. 2020. Easy, reproducible and quality-controlled data collection with CROWDAQ. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing: System Demonstrations*, pages 127–134, Online. Association for Computational Linguistics.

Anusri Pampari, Preethi Raghavan, Jennifer Liang, and Jian Peng. 2018. emrQA: A large corpus for question answering on electronic medical records. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 2357–2368, Brussels, Belgium. Association for Computational Linguistics.

Dimitris Pappas, Ion Androutsopoulos, and Haris Papageorgiou. 2018. BioRead: A new dataset for biomedical reading comprehension. In *Proceedings of the Eleventh International Conference on Language Resources and Evaluation (LREC 2018)*, Miyazaki, Japan. European Language Resources Association (ELRA).

Dimitris Pappas, Petros Stavropoulos, Ion Androutsopoulos, and Ryan McDonald. 2020. BioMRC: A dataset for biomedical machine reading comprehension. In *Proceedings of the 19th SIGBioMed Workshop on Biomedical Language Processing*, pages 140–149, Online. Association for Computational Linguistics.

Colin Raffel, Noam Shazeer, Adam Roberts, Katherine Lee, Sharan Narang, Michael Matena, Yanqi Zhou, Wei Li, and Peter J. Liu. 2020. Exploring the limits of transfer learning with a unified text-to-text transformer. *Journal of Machine Learning Research*, 21(140):1–67.

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

Abhilasha Ravichander, Alan W Black, Shomir Wilson, Thomas Norton, and Norman Sadeh. 2019. Question answering for privacy policies: Combining computational and legal perspectives. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 4947–4958, Hong Kong, China. Association for Computational Linguistics.

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

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

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Ł ukasz Kaiser, and Illia Polosukhin. 2017. Attention is all you need. In *Advances in Neural Information Processing Systems*, volume 30. Curran Associates, Inc.

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

Yi Yang, Wen-tau Yih, and Christopher Meek. 2015. WikiQA: A challenge dataset for open-domain question answering. In *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing*, pages 2013–2018, Lisbon, Portugal. Association for Computational Linguistics.

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