Extractive NarrativeQA with Heuristic Pre-Training

Lea Frermann*
School of Computing and Information Systems
The University of Melbourne, Australia
lea.frermann@unimelb.edu.au

Abstract

Although advances in neural architectures for NLP problems and unsupervised pre-training led to impressive improvements on question answering and natural language inference, reasoning over long texts still poses a great challenge. Here, we consider the task of question answering from full narratives (e.g., books or movie scripts), or their summaries, tackling the NarrativeQA challenge (NQA; Kocisky et al. (2018)). We introduce a heuristic *extractive* version of the data set, which allows us to approach the more feasible problem of answer extraction (rather than generation). We develop models for passage retrieval and answer span prediction using this data set. We use pre-trained BERT embeddings for injecting prior knowledge into our system. We show that our setup leads to state of the art performance on *summary-level* QA. On *narrative-level* QA, our model performs competitively on the METEOR metric. We analyze the relative contributions of BERT embeddings and the extractive model setup, and provide a detailed error analysis.

1 Introduction

With recent advances in machine learning techniques, the availability of sizable data sets as well as compute power, natural language processing has made impressive advances across a variety of NLP tasks. A striking gap between machine and human performance, however, remains the ability to *comprehend* text and make *inferences* over multiple pieces of information.

Automatic question answering (QA) from text has received much recent attention as a task designed towards bridging this gap. A variety of question answering tasks and data sets with different levels of difficulty have been proposed recently, ranging from questions paired with short, relevant documents containing immediately inferable answers (SQUAD; Rajpurkar et al. (2016)), over questions to be answered from sets of documents and requiring to connect facts through multi-step inferences (WikiHop; Welbl et al. (2018)) to naturally occurring questions as Google search queries, paired with sets of Wikipedia pages (Natural Questions; Kwiatkowski et al. (2019)).

Common characteristics of those data sets are (1) sets of (question, document, answer)-tuples in the order of tens- to hundreds of thousands training and test examples; (2) *extractive* answers which can be pin-pointed in the reference documents; (3) the reference documents from which answers are derived are of comparatively short length (e.g., an average of 100 tokens per reference for WikiHop, vs 60K tokens in NQA). All recently proposed successful QA systems were trained in a supervised way, heavily relying on the availability of answer-annotated data sets as described above.

In this work we consider the highly challenging task of narrative question answering (NQA), as introduced by Kocisky et al. (2018). In NQA, a system is presented with a question on the plot of a narrative (a book or a movie) and produces a free-text answer given the raw book or movie script text.[1] The data set was created by pairing each original narrative with a human-created summary, and crowd sourcing a large set of of question-answer pairs based on the summary. Questions are derived from the summaries to deliberately avoid answers to be straightforwardly extractable from the full narrative texts.

Several interesting challenges arise in NQA: (1) although answers are typically localized in the summary, the corresponding answer in the book

*Work done while the author was employed at Amazon.

[1] Although the NQA data set includes both books and movie scripts, and we will refer collectively to *books* for simplicity.

Proceedings of the Second Workshop on Machine Reading for Question Answering, pages 172–182
Hong Kong, China, November 4, 2019.

often requires reasoning across paragraphs or even chapters; (2) answers are *abstractive* and as such not necessarily verbatim in the reference documents; (3) the size of the data set, shown in Table 2, is comparatively small making supervised training challenging.

This paper explores the utility of heuristic, but inexpensive training data sets for NQA. We formulate NQA as an extractive question answering task, leveraging the fact that by construction of the data set, answers tend to be extractable locally from the summary text (cf., Table 1 for examples). While ultimately an abstractive system, which synthesizes an answer based on information in the text, is desirable, a conceptually simpler extractive approach can serve as a first and more feasible step towards the goal of answer generation. Our evaluation shows that our extractive system performs competitively on summary- and book-level NQA.

We construct a heuristic extractive NQA data set by leveraging characteristics of the generating process of the original data. Specifically, since question-answer pairs were synthesized based on the summaries, we hypothesize that the answer to a question can typically be found in a single summary sentence (or subspan thereof). We develop heuristics to retrieve those spans.

Based on our heuristic extractive data set we train models for two tasks: (1) Question-based sentence retrieval, which, given a question, selects relevant passages for a question (which may serve as input to a sophisticated QA model); and (2) SQUAD-style answer extraction, where the system learns to point to the beginning and end of the answer in the reference text. We train systems for sentence-retrieval and answer extraction on top of pre-trained BERT embeddings (Devlin et al., 2018), which serve as a source of prior knowledge.

We train question answering systems on summary-question-answer tuples, and evaluate the systems on (1) *summary references* and (2) on the *full book text*. Although summaries are required for training, our model can answer questions on unseen test books with no need for a summary.

While a variety of systems has been proposed for summary-level based NQA, the full NQA challenge of answering questions based on the full, raw narrative text has received less attention. Conceptually similar to our approach of deriving heuristics from question-answer-summary tuples, very recent work proposes heuristic generative pre-training directly on book passages (Tay et al., 2019). They use pointer-generator networks (See et al., 2017) which allow to produce an answer by sampling from the vocabulary (*generate*) even when the answer cannot be *pointed* to directly in the context passage.

Our system achieves state-of-the-art results on summary-level answer extraction, and performs competitively on the book-level specifically on METEOR, a semantically informed evaluation metric which scores semantic relevance beyond word overlap.

In summary, our contributions are:

1. Augmentation of existing (sparse) data sets with heuristic, inexpensive and supervised training data, with an application to extractive question answering for NQA

2. State-of-the-art results on the summary level NQA benchmark; and competitive results on the book-level NQA task under the METEOR metric, which takes into account synonymy in addition to word overlap

3. An analysis of common errors shedding light on shortcomings in model performance as well as evaluation

2 Task Description

The NarrativeQA data set (Kocisky et al., 2018) provides a testbed for question answering on raw narrative text. It consists of over 1,567 publicly available full-length narrative documents (books or movie scripts), each paired with a human-created plot summary. For each document a set of question-answer pairs was collected by presenting human annotators with the summary. The annotators generated a set of questions (30 per summary) together with free-text answers (two answers per question, from distinct annotators), for a total of 46,765 question-answer pairs. Considering the variety in question types, narrative styles (books and movie scripts of different genres), sheer length of the documents, and the fact that answers need to be synthesized, this data set is too small to train models in a purely in-domain supervised way.

We address the above challenges in two ways. First, we incorporate prior knowledge in the form of pre-trained word embeddings (Devlin et al., 2018). Second, we recognize that by construction of the data set, answers to questions can generally be localized in the summaries, even though

Q: Why does Nora track Mark down?	**G1:** Malcom's suicide **G2:** to confront him after Malcolm commits suicide
E: Nobody knows the true identity of Hard Harry [...] until Nora Diniro (Mathis), a fellow student, tracks him down and confronts him the day after a student named **Malcolm commits suicide** after Harry attempts to reason with him.	
Q: Why did the couple visit medium Shaun San Dena in Pasadena in 1969?	**G1:** their son has been hearing voices from evil spirits **G2:** because their son was hearing evil spirits voices
E: In 1969 Pasadena, California, a couple seeks the aid of the medium Shaun San Dena (Flor de Maria Chahua) saying their **son (Shiloh Selassie) has been hearing evil spirits' voices** after stealing a silver necklace [...]	
Q: How was Hadley's Hope Colony destroyed?	**G1:** the nuclear blast from the damaged power plant **G2:** an explosion
E: All four escape moments before the station explodes with the colony consumed by the **nuclear blast**.	

Table 1: Example questions (Q) from the NarrativeQA data set, with gold free-text answers (G), the most relevant sentence as automatically extracted from the summary (E) and the most relevant sub-sentence level span (boldface).

the free-text answers are typically not found verbatim in the summary. We leverage this property to construct *extractive* data sets for sentence-level and sub-sentence level answer extraction.

3 Data Sets for Extractive NarrativeQA

We derive data sets for supervised query-based sentence retrieval (Section 3.1), and answer span extraction (Section 3.2).

3.1 Sentence Retrieval Data Set

For each question, and its corresponding summary, we proceed as follows. We first obtain a relevance score of each summary sentence s to the input question q: we concatenate the question[2] q with both human-created free text answers $a1, a2$,

$$z = [q; a1; a2], \quad (1)$$

and obtain a relevance score of each summary sentence s w.r.t. z by passing both through the Universal Sentence Encoder (USE)[3] (Cer et al., 2018) and computing the cosine similarity between the encodings,

$$rel_z(s) = cos(\text{USE}(z), \text{USE}(s)). \quad (2)$$

We can thus rank summary sentences by their relevance to input qa-pair z. Our method can serve as a sentence or passage retrieval system, providing pre-selected input to a more sophisticated question answering model. Assuming the top-ranked sentence to be the true relevant sentence (and all other sentences to be irrelevant), we train supervised retrieval models given a question as input. We further use sentence relevance scores as a basis for heuristic answer-span annotation as described in the following section. Example questions, together with the most relevant retrieved sentence, are shown in Table 1.

3.2 Answer Span Prediction Data Set

Although sentence retrieval is an important step towards question answering from narratives, ultimately a more flexible answer granularity is desirable. Building on sentence-level relevance scores, given a question-answer pair, we extract the most relevant contiguous word sequence to a question q in the summary. We employ the following backoff strategy:

1. if available, return an exact match of one of the reference answers (if both answer candidates match, choose one at random)
2. if unsuccessful: considering the three most question-relevant sentences as determined by the USE (Section 3.1) find the longest substring bounded by content words in the answers
3. if unsuccessful: considering any sentence in the summary, return the longest substring

[2] We remove the question mark and the first word if it indicates a wh-question.

[3] In preliminary experiments we tested ROUGE-L as an alternative to USD, but found a bias towards mapping to short sentences.

	train	valid	test
# QA-pairs	32,170	3,461	10,557
# documents	1,102	115	355

Table 2: Statistics of the NarrativeQA data set (Kocisky et al., 2018). We obtain a heuristic answer match for each original question, and maintain the original train/valid/test splits.

bounded by content words in the answers

Our resulting dataset of questions paired with answer-annotated summaries containing the answers, allows us to train SQUAD-style answer prediction systems (cf., Section 5; Rajpurkar et al. (2016); Devlin et al. (2018)). Figure 1 shows examples of automatically annotated answer spans in NarrativeQA summaries (boldfaced).

4 Experiment Setup

We train systems for sentence retrieval and answer span prediction on questions paired with answer-annotated summaries, obtained as described in Sections 3.1 and 3.2. We evaluate sentence retrieval and answer span prediction performance on both summary level data, and full narrative texts. We evaluate our extractive model predictions against the original, *abstractive* NarrativeQA gold answers using the evaluation setup proposed in the original paper to ensure comparability.

Our experiments investigate (a) the effectiveness of a heuristic training data set on sentence retrieval and answer span prediction in the context of NQA; (b) the extent of generalization of systems trained on summary data to book full texts; and (c) the utility of prior knowledge in the form of pre-trained word embeddings. We train sentence retrieval and span prediction models on top of pre-trained BERT embeddings (Devlin et al., 2018).

4.1 BERT

BERT embeddings (Devlin et al., 2018) are contextualized word representations, pre-trained on enormous training corpora on unsupervised word- and sentence prediction tasks using bi-directional transformers. They have been shown to encode substantial semantic and syntactic information, and have been efficiently fine-tuned towards a variety of NLP tasks leading to new state-of-the-art results (Devlin et al., 2018). Here, we fine-tune BERT embeddings for NQA sentence retrieval and answer span selection, as described below.

	accuracy	precision	recall	f1
$p_{rel} > 0.5$	0.87	0.88	0.83	0.86

Table 3: Results on summary-level sentence-relevance classification on the NQA test set of 25K question-answer pairs. We set the relevance threshold to $p > 0.5$.

5 Sentence Retrieval

Given a question and a reference text, our models retrieve the most relevant sentences from the reference to the query by computing a relevance score for each sentence in the reference.

Approach Given a large set of sentence-question pairs, we train a relevance prediction model on top of BERT embeddings. Following closely the architecture for BERT-based sentence classification, our system takes as input the BERT-embedded query q concatenated with a single BERT-embedded summary sentence s. The two sequences are separated with a special separation token ($[SEP]$) and pre-pended with another special token $[CLS]$ which will be trained to capture the aggregate sentence pair representation,

$$z = [CLS]enc(q)[SEP]enc(s). \quad (3)$$

The final sentence pair representation $[CLS]$ is passed through a single linear layer followed by a softmax layer to produce an output class ($relevant$ vs $irrelevant$ in our case). We use queries paired with top-ranked summary sentences (Section 3.1) as positive examples, and queries paired with random sentences from the same summary as negative examples, and minimize cross-entropy classification loss.

For each sentence-query pair we obtain a relevance score $\in [0, 1]$, from which we can derive a summary sentence ranking by query relevance. We retrieve the top n most relevant sentences from this ranking for further predictions.

We use the default parameters from the original BERT implementation.[4]

Summary-level results We apply our model to the book summaries from test data set of NarrativeQA. We evaluate the extent to which truly

[4] https://github.com/google-research/bert

	p@1	p@5	MRR
BM25f	10.53	51.42	0.276
BERT	13.80	53.02	0.305

Table 4: Fraction of correct answers contained in the top {1 / 5} answer candidates, and MRR of the correct answer in passages retrieved by the BERT-based retrieval method (BERT) or an IR method (BM25f).

relevant sentences (as extracted by our heuristic method) were assigned a relevance probability $p > 0.5$. Results are shown in Table 3, and show that the model detects the most relevant summary sentence for a question accurately across a variety of metrics.

Book-level results We apply our model to the considerably harder task of NQA on full documents, computing a question-specific relevance score for each sentence in the document. Note that we cannot evaluate retrieval scores directly, because we do not have access to a gold standard of relevant book sentences for a given question. Instead, we treat our system as a passage retrieval model given an input question. As an approximation to the quality of the retrieved passages we compute the extent to which the correct answer is found in the N most frequent answer candidates.[5]

We compare our BERT-retrieval with an IR-style retrieval system (BM25f; Zaragoza et al. (2004)) which retrieves text passages of five consecutive sentences based on word token and character mention overlap with the question. From both systems, we retrieve the 20 most relevant predicted sentences, each in a context of ± 2 sentences.

The results are shown in Table 4. We can observe that BERT-based retrieval outperforms the IR retrieval-based model. We will also incorporate this model as a passage-preselection module for book-level answer span prediction in Section 6.

Qualitatively, we observed that most book sentences receive a very low relevance probability in our BERT-based retrieval system, which makes the model amenable for the task of narrowing down the context to few relevant passages. For example, on average across all books, only 1.4% of all sentences are predicted as relevant with $p >= 0.8$ and 4.3% with $p >= 0.01\%$.

[5]We evaluate our system only in the context of *who?* questions with an entity as answer and consider all book entities as candidate answers. We extract character mentions using the BookNLP pipeline (Bamman et al., 2014).

6 Answer Span Prediction

Given a question and a reference text (summary or full narrative), the task is to predict a contiguous sub-span of arbitrary length in the reference text as the answer to the question.

Approach We fine-tune BERT embeddings for answer extraction, similar to the approach for BERT-based SQUAD question answering in Devlin et al. (2018). Given a query q and a text passage c, we map both to BERT embeddings, and concatenate the embedded representations,

$$z = [CLS]enc(q)[SEP]enc(c). \quad (4)$$

BERT fine-tuning for answer-span prediction involves training a start-vector representation S and an end-vector representation E. The probability of a word $i \in enc(c)$ being the start of the answer is the dot-product between $enc(c)_i$ and S, softmax-normalized over all words in $enc(c)$; and the probability distribution over end tokens is computed analogously. The probability of a span from word i to word j, s.th. $i < j$, is the sum of its start and end position

$$S \times enc(c)_i + E \times enc(c)_j. \quad (5)$$

Pointing to the $[CLS]$ token, the model also has the capacity to predict no answer at all. We use the start and end positions of our heuristic answer spans (Section 3.2) as gold training examples, and maximize the sum of log likelihoods of the start and end position as our training objective.

While we use the whole summaries as contexts for summary-based QA, considering full narrative texts is prohibitive. To this end, we leverage the sentence retrieval model from Section 5 to obtain a subset of relevant sentences. In our experiment we retrieve the 100 most likely sentences given a question, each in a context of ± 2 sentences, resulting in contexts of (up to) 500 sentences per question.

Even after this pre-selection, memory constraints prohibit processing of the full contexts, or summary texts. Following Kocisky et al. (2018), we limit context length to a maximum of 384 words, split the original reference documents into multiple such segments, and pass each segment individually as context, and return the most likely

model	BLEU-1	BLEU-4	METEOR	Rouge-L
BiDAF Span Prediction (Kocisky et al., 2018)	33.45	15.69	15.68	36.74
DecaProp (Tay et al., 2018)	42.00	23.42	21.80	44.69
ConZNet (Indurthi et al., 2018)	42.76	22.49	19.24	46.67
BERT SQUAD train	36.22	17.14	23.61	48.58
BERT SQUAD train 31K	40.71	20.60	19.78	45.06
BERT heur	**50.36**	**24.24**	**27.09**	**58.50**

Table 5: Summary-level answer extraction results by previous models and our systems trained on out-of-domain SQUAD data (BERT SQUAD *), and our heuristic data set (BERT heur). All results reported on the NarrativeQA test split.

model	BLEU-1	BLEU-4	METEOR	Rouge-L
BiDAF Span Prediction (Kocisky et al., 2018)	5.68	0.25	3.72	6.22
AS Reader 10 chunks (Kocisky et al., 2018)	**19.09**	1.81	4.29	14.03
AS Reader 20 chunks (Kocisky et al., 2018)	19.06	**2.11**	4.37	14.02
IAL-CL (Tay et al., 2019)$^{(\star)}$	**22.92**	**2.47**	**5.59**	**17.67**
BERT SQUAD train	9.06	1.03	4.29	10.58
BERT SQUAD train 31K	9.23	1.47	3.55	10.29
BERT heur	12.26	2.06	**5.28**	**15.15**

Table 6: Book-level answer extraction results by previous models and our systems trained on out-of-domain SQUAD data (BERT SQUAD *), and our heuristic data set (BERT heur). All results reported on the NarrativeQA test split. $^{(\star)}$: Work developed concurrently with ours; added post acceptance.

span across all passages as an answer. For each test input, we return the most likely non-empty answer candidate returned by the model.

In order to disentangle the contribution of powerful BERT embeddings from the utility of our heuristic training corpus, we also trained an answer extraction model using SQUAD-V2.0 training data (Rajpurkar et al. (2018); BERT SQUAD). We train the models using either the full SQUAD data set, or a random subset of 31,000 training items, comparable in size to our heuristic training data set. On the one hand, this data set is a gold-standard of perfect context-span to answer correspondences. On the other hand, the data stems from a different domain, and thus potentially less informative for the NarrativeQA task.

We evaluate the predicted answers against the human-provided free-text answers using BLEU (Papineni et al., 2002) and METEOR (Banerjee and Lavie, 2005) scores. We report results given (1) summaries as contexts, and (2) the full narrative texts, and compare against previously reported results on the respective tasks.

Summary-level Results Table 5 displays summary-level answer span extraction results for previous models (top), the BERT-based span prediction model trained on SQUAD data (center), and the same model trained on our heuristic extractive NQA corpus (bottom).

BiDAF is a span prediction model, conceptually similar to our own and was used as a baseline method in Kocisky et al. (2018). DecaProp (Tay et al., 2018) is a neural network which, through dense connections between neighboring layers, is designed to distill information from hierarchical passage representations (over words, sentences, and paragraphs). CoZNet (Indurthi et al., 2018) is a neural network architecture designed to 'zoom into' relevant passages of contiguous, long text passages, using co-attention on query and passage and reinforcement learning with answer generation as target. The latter models *generate*, rather than extract, an answer. All models were evaluated against the human free-text answers.

Our model trained on the heuristic data set outperforms all prior work. The model trained on SQUAD data compares poorly against all other

models, demonstrating that the prior information from BERT embeddings by themselves do not automatically lead to improvements on NQA. Interestingly, the SQUAD-data trained model perform better with fewer data (31K) compared with the full training data set, suggesting that fitting the model to SQUAD-data prediction decreases its generalization ability to out-of-domain NQA test data. The strong performance with our heuristic training corpus suggests that a heuristic and potentially noisy in-domain data set is of great utility for summary-level answer span extraction.

Note that our model scores higher than the human results reported in (Kocisky et al., 2018), where the automatic evaluation metrics were computed by evaluating one human annotation against the other. By extracting the answer string from the summary, our system is frequently in agreement with at least one human annotator; however, as humans were allowed to provide free-text answers, the two annotations often do not match exactly, resulting in overly pessimistic automatic scores. We discuss shortcomings of automatic evaluation metrics like BLEU in the context of NarrativeQA in more detail in Section 7.

Q1	Who is Mark Hunter?
G	he is a high school student in Phoenix
E	high school student (✓)
Q2	Why do more students tune into Mark's show?
G	Mark talks about what goes on at school and in the community
E	speaks his mind (✓)
Q3	Why do Faulkland and Julia always fight?
G	he thinks she's unfaithful
E	jealous suspicion. He is constantly fretting himself about her fidelity (✓)
Q4	Who was Murphy's ghost?
G	Cooper from the future
E	a poltergeist (✗)

Figure 1: Example questions (Q) with gold (G) and top-ranking model-extracted answer (E) from the book summaries. ✓: correct; ✗: incorrect.

Book-level Results Although a range of prior models have been proposed for summary-level QA, the only prior work that tackles the full NarrativeQA task has been developed concurrently with our work (IAL-CL; Tay et al. (2019)). IAL-CL is a pipelined approach of tfidf/cosine similarity-based passage retrieval pointer-generator networks for question answering model, together with sophisticated block-based alignment (IAL) strategy, trained with curriculum learning (CL). We also compare against the most competitive systems described in the original paper (Kocisky et al., 2018).

All results are shown in Table 6. We compare our own model trained on the heuristic training corpus (bottom), against another span prediction model, Bi-Directional Attention Flow (BiDAF; Seo et al. (2016)), as reported in Kocisky et al. (2018), as well as their most competitive model, an adaptation of the Attention Sum Reader (Kadlec et al., 2016) (AS Reader). AS Reader follows an encoder-decoder architecture with attention, where the decoder is an LSTM sequence decoder which can synthesize an answer (rather than extract). Both prior models are combined with a passage pre-selection method (similar to our own), which is based on tf-idf based cosine similarity of answers (for training sets) and questions for (test sets). Like for the summary-level task, we compare our architecture fine-tuned on quality out-of-domain training data (SQUAD).

Tay et al. (2019) achieve the most competitive results across the board. Our model outperforms the conceptually similar span extraction model (BiDAF). The AS Reader performs similarly to our model, with the ranking depending on the metric used. Our model outperforms previous systems in terms of METEOR score. METEOR includes synonym matching and as such recognizes semantically similar predictions to the gold standard. The error analysis (Section 7), provides a variety of examples which demonstrate that model predictions are indeed often correct, despite having little word overlap with the gold standard. Like in the summary-level evaluation, models trained on our own corpus outperform the SQUAD-based models, suggesting again the utility of training on easily obtainable, inexpensive but heuristic in-domain data.

7 Error Analysis

We inspect a variety of examples on both summary- and narrative level QA to shed light on shortcomings of the model and evaluation. We show qualitative support for our model's discrepancy in METEOR and BLEU performance (Ta-

Q5	What is Tom trying to desperately get working?
G	his latest invention
E	a photo telephone (✓)
C	I 'm trying to make **a photo telephone**. I have the telephone part down Pat, but I can't see anything of the photo image.
Q6	What is Dubuches passion besides painting?
G	music
E	music (✓)
C	his landscapes were at least conscientiously painted, excellent in intention; **but his real passion was music**, a madness for music, a cerebral bonfire which set him on a level with the wildest of the band.
Q7	How does Claude die?
G	he hangs himself
E	knocked down by a little hand-truck (✗)
C	**Claude**, stepping back, **narrowly escaped being knocked down by a little hand-truck** which two big full-bearded fellows brought up
Q8	How does Mr. Peters feel about Tom not allowing him access to the patents?
G	he gets angry
E	embarrassed (✗)
C	"Who are these men?" **Tom felt a little embarrassed** over what he had said. "Couldn't he make him pay?" asked Tom.
Q9	What were Owen and George found doing the morning after Brenda died?
G	they were laughing and singing
E	singing snatches of songs (**?**)
C	George and Owen were **singing snatches of songs** and gibbering like two great apes.
Q10	How long does Jimmy Gator have to live?
G	just a few months
E	two months (✓)
C	[[Jimmy Gator]]: I have about **two months**, I have no time.
Q11	What happened to the doctor who was called to the house after Mortimer's brothers went insane?
G	he collapsed at the sight of Brenda's body
E	fell into a chair (✓)
C	you will recollect that Mortimer Tregennis, in describing the episode of his last visit to his brother's house, remarked that **the doctor on entering the room fell into a chair?**
Q12	How does Linda try to commit suicide?
G	by overdosing on Earls prescription medication
E	swallows a whole bottle of pills (✓)
C	**Linda** takes some pills. Then she takes some more... Then she takes some more... Then **she swallows a whole bottle of pills**... She drinks from a small bottle of vodka
Q13	What is ironic about Donnie's teeth being knocked out when he falls from the pole?
G	he no longer has to worry about getting braces on his teeth
E	Donnie's mouth is full of blood and his teeth (✗)
C	he.he . **Donnie's mouth is full of blood and his teeth are broken** ... [[Donnie]]: My teeff ... My teeef .... [[Jim Kurring]]: You 're ok

Figure 2: Example questions (Q), gold (G) and extracted (E) answers, and local extraction contexts (C) for NQA on full narrative texts. Correct (✓), incorrect (✗) or undecidable (**?**) answers.

ble 6), with model predictions frequently *paraphrasing* gold answers. Furthermore, incorrect answer predictions are often still topically relevant to the question, which highlights a need for models that go beyond word co-occurrence based prior knowledge (as obtained through pre-trained embeddings like BERT).

Figure 1 displays example questions with gold and model predicted answers from the summaries as reference documents. Example Q1 shows a case where the correct answer is conceptually simple and easily extractable. In examples Q2 and Q3, answers are complex concepts as indicated by the more verbose human and model-produced answers. Still, the model predictions are correct in both cases. For Example Q4 the model prediction is incorrect, even though the predicted span is clearly semantically related to the question.

We show questions with gold and model answers based on passages from the full narrative in Figure 2. We also include the local context from which the model answer was extracted (the full context is up to 500 sentences long). Examples Q5, Q6, Q10, Q11 and Q12 are predicted correctly. Note that some predicted answers have very little lexical overlap with the gold answer, although the prediction is correct as supported by the context. Example Q7 illustrates a case where the model-predicted answer is wrong, however, the proposed passage refers to a situation which is similar to the correct answer (nearly escaping a potentially deadly situation, rather than real death of the same person). Example Q8 is a wrong prediction, a result of confusing semantic roles of the participants. Example Q9 seems to be correct, however, from the context it is not clear whether the extracted passage indeed refers to *the morning after brenda died*. Example Q13 shows another wrong prediction, however, the extracted context is arguably semantically relevant to the query.

Overall, the error analysis suggests that purely data-driven models tend to overly rely on surface semantic similarity and local contexts. We also find that automatic evaluation scores like BLEU and METEOR, which rely on word overlap, are overly conservative regarding the output of our model. A series of recent papers discussed problems of comparing models on abstractive NLI tasks using automatic metrics as the ones listed above (Novikova et al., 2017; Chaganty et al., 2018). While there is decent agreement between human and automatic judgments on bad model outputs, disagreements tend to be substantial on good outputs. Our analysis provides further support for these observations.

8 Conclusion

Answering questions on the basis of long and comples texts is a major challenge even for the most advanced NLP methods. While the NarrativeQA data set provides an excellent benchmark for this task, it is comparatively small, and not designed for developing *extractive* question answering models, an arguably more straightforward task compared to extractive Q&A. We heuristically constructed an extractive summary-level Q&A data set and showed that it can be used to train accurate sentence- and span-level answer extraction systems from summary text. We also applied our models to full book text and showed that it outperforms IR-based retrieval systems when incorporated in a entity classification network.

On book-level QA, our model achieves competitive METEOR results. Our results and error analysis suggest that pure word overlap-based evaluation methods can lead to misleading results. The model produced answers were often correct despite lacking lexical overlap with the gold answers. Word overlap-based methods like BLEU or METEOR are agnostic of such hits. METEOR, in contrast takes synonymy into account, and our methods outperformed previous systems in this metric. Our observation follows recent published work on evaluating abstractive NLI systems (Chaganty et al., 2018). Concurrently with improving NLI methodology, it is worth investing in the development of evaluation methods that reflect progress faithfully.

We believe that general, prior knowledge is necessary for successful narrative understanding. We incorporated prior knowledge through pre-trained BERT embeddings, and used heuristic but inexpensive data for supervised training. We hope that our approach opens up avenues for more sophisticated data creation methods for future work, including background knowledge and better models of the full stories.

Acknowledgments

Many thanks to Alex Klementiev, Stefanos Angelidis, Roi Blanco, Lluiz Marquez, Diego Marcheggiani and Hugo Zaragoza for helpful feedback.

References

David Bamman, Ted Underwood, and Noah A. Smith. 2014. 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, Baltimore, Maryland. Association for Computational Linguistics.

Satanjeev Banerjee and Alon Lavie. 2005. METEOR: An automatic metric for MT evaluation with improved correlation with human judgments. In *Proceedings of the ACL Workshop on Intrinsic and Extrinsic Evaluation Measures for Machine Translation and/or Summarization*, pages 65–72, Ann Arbor, Michigan. Association for Computational Linguistics.

Daniel Cer, Yinfei Yang, Sheng-yi Kong, Nan Hua, Nicole Limtiaco, Rhomni St John, Noah Constant, Mario Guajardo-Cespedes, Steve Yuan, Chris Tar, et al. 2018. Universal sentence encoder. *arXiv preprint arXiv:1803.11175*.

Arun Chaganty, Stephen Mussmann, and Percy Liang. 2018. The price of debiasing automatic metrics in natural language evalaution. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 643–653, Melbourne, Australia. Association for Computational Linguistics.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2018. Bert: Pre-training of deep bidirectional transformers for language understanding. *arXiv preprint arXiv:1810.04805*.

Sathish Reddy Indurthi, Seunghak Yu, Seohyun Back, and Heriberto Cuayahuitl. 2018. Cut to the chase: A context zoom-in network for reading comprehension. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 570–575, Brussels, Belgium. Association for Computational Linguistics.

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 (Volume 1: Long Papers)*, pages 908–918, Berlin, Germany. Association for Computational Linguistics.

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

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

Jekaterina Novikova, Ondřej Dušek, Amanda Cercas Curry, and Verena Rieser. 2017. Why we need new evaluation metrics for NLG. In *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing*, pages 2241–2252, Copenhagen, Denmark. Association for Computational Linguistics.

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 of the Association for Computational Linguistics*, pages 311–318, Philadelphia, Pennsylvania, USA. Association for Computational Linguistics.

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

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.

Abigail See, Peter J. Liu, and Christopher D. Manning. 2017. Get to the point: Summarization with pointer-generator networks. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1073–1083, Vancouver, Canada. Association for Computational Linguistics.

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

Yi Tay, Anh Tuan Luu, Siu Cheung Hui, and Jian Su. 2018. Densely connected attention propagation for reading comprehension. In S. Bengio, H. Wallach, H. Larochelle, K. Grauman, N. Cesa-Bianchi, and R. Garnett, editors, *Advances in Neural Information Processing Systems 31*, pages 4906–4917. Curran Associates, Inc.

Yi Tay, Shuohang Wang, Anh Tuan Luu, Jie Fu, Minh C. Phan, Xingdi Yuan, Jinfeng Rao, Siu Cheung Hui, and Aston Zhang. 2019. Simple and effective curriculum pointer-generator networks for reading comprehension over long narratives. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 4922–4931, Florence, Italy. Association for Computational Linguistics.

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.

Hugo Zaragoza, Nick Craswell, Michael J Taylor, Suchi Saria, and Stephen E Robertson. 2004. Microsoft cambridge at trec 13: Web and hard tracks. In *TREC*, volume 4, pages 1–1.