Multi-Relational Question Answering from Narratives: Machine Reading and Reasoning in Simulated Worlds

Anonymous ACL submission

Abstract

Question Answering (QA), as a research field, has primarily focused on either knowledge bases (KBs) or free text as a source of knowledge. These two sources have historically shaped the kinds of questions that are asked over these sources, and the methods developed to answer them. In this work, we look towards a practical use-case of *QA over user-instructed knowledge* that uniquely combines elements of both *structured QA* over knowledge bases, and *unstructured QA* over narrative, introducing the task of *multi-relational QA over personal narrative*. As a first step towards this goal, we make three key contributions: (i) we generate and release TEXTWORLDSQA, a set of five diverse datasets, where each dataset contains dynamic narrative that describes entities and relations in a simulated world, paired with variably compositional questions over that knowledge, (ii) we perform a thorough evaluation and analysis of several state-of-the-art QA models and their variants at this task, and (iii) we release a lightweight Python-based framework we call TEXTWORLDS for easily generating arbitrary additional worlds and narrative, with the goal of allowing the community to create and share a growing collection of diverse worlds as a test-bed for this task.

1 Introduction

Personal devices that interact with users via natural language conversation are becoming ubiquitous (e.g., Siri, Alexa), however, very little of that conversation today allows the user to teach, and then query, new knowledge. Most of the focus in these personal devices has been on Question Answering (QA) over general world-knowledge (e.g., *"who was the president in 1980"* or *"how many ounces are in a cup"*). These devices open a new and exciting possibility of enabling end-users to teach machines in natural language, e.g., by expressing the state of their personal world to its virtual assistant (e.g., via narrative about people and events in that user's life) and enabling the user to ask questions over that personal knowledge (e.g., *"which engineers in the QC team were involved in the last meeting with the director?"*).

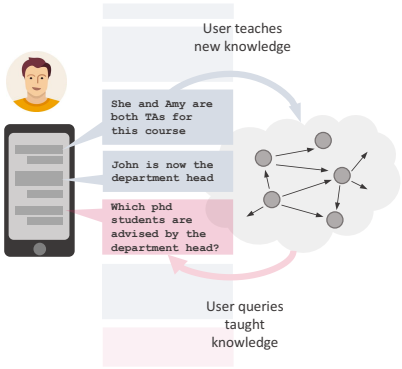

Figure 1: Illustration of our task: relational question answering from dynamic knowledge expressed via personal narrative

This type of questions highlight a unique blend of two conventional streams of research in Question Answering (QA) – QA over *structured* sources such as knowledge bases (KBs), and QA over *unstructured* sources such as free text. This blend is a natural consequence of our problem setting: (i) users may choose to express rich relational knowledge about their world, in turn enabling them to pose complex **compositional** queries (e.g., *"all CS undergrads who took my class last semester"*), while at the same time (ii) personal knowledge generally evolves through

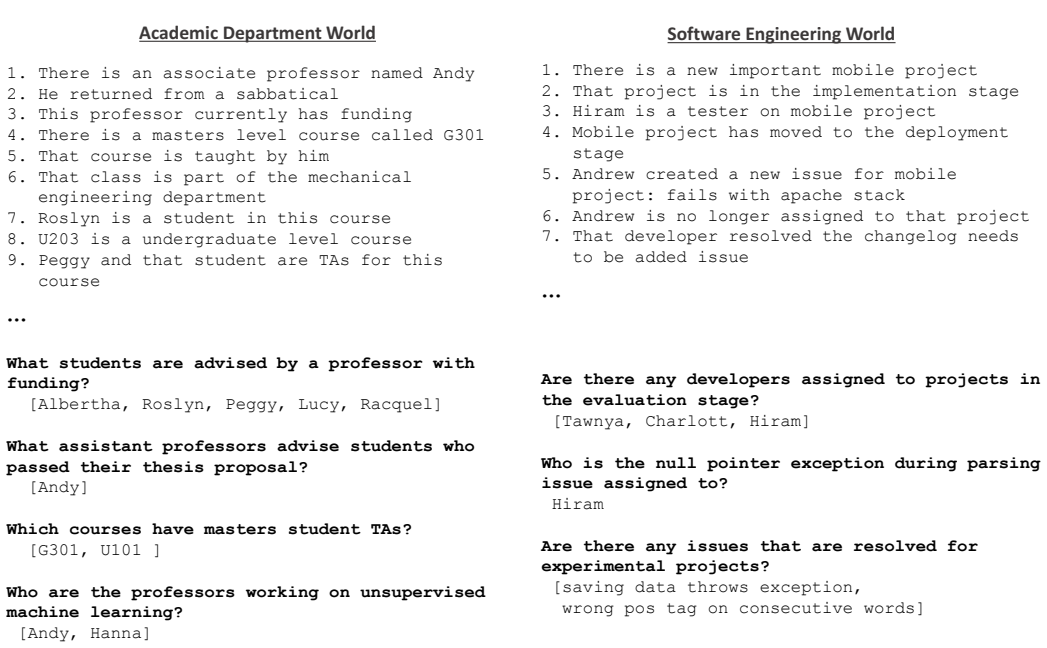

Figure 2: Illustrative snippets from two sample worlds. We aim to generate natural-sounding first-person narratives from five diverse worlds, covering a range of different events, entities and relations.

time and has an open and growing set of relations, making natural language the only practical interface for creating and maintaining that knowledge by non-expert users. In short, the task that we address in this work is: **multi-relational question answering from dynamic knowledge expressed via narrative**.

Although we hypothesize that question-answering over personal knowledge of this sort is ubiquitous (e.g., between a professor and their administrative assistant, or even if just in the user's head), such interactions are rarely recorded, presenting a significant practical challenge to collecting a sufficiently large real-world dataset of this type. At the same time, we hypothesize that the technical challenges involved in developing models for relational question answering from narrative would not be fundamentally impacted if addressed via sufficiently rich, but controlled simulated narratives. Such simulations also offer the advantage of enabling us to directly experiment with stories and queries of different complexity, potentially offering additional insight into the fundamental challenges of this task.

While our problem setting blends the problems of relational question answering over knowledge bases and question answering over text, our hypothesis is that end-to-end QA models may learn to answer such multisentential relational queries, without relying on an intermediate knowledge base representation. In this work, we conduct an extensive evaluation of a set of state-of-the-art end-to-end QA models on our task and analyze their results.

2 Related Work

Question answering has been mainly studied in two different settings: KB-based and text-based. KB-based QA mostly focuses on parsing questions to logical forms (Zelle and Mooney, 1996; Zettlemoyer and Collins, 2012; Berant et al., 2013; Kwiatkowski et al., 2013; Yih et al., 2015) in order to better retrieve answer candidates from a knowledge base. Text-based QA aims to directly answer questions from the input text. This includes works on early information retrieval-based methods (Banko et al., 2002; Ahn et al., 2004) and methods that build on extracted structured representations from both the question and the input text (Sachan et al., 2015; Sachan and Xing, 2016; Khot et al., 2017; Khashabi et al., 2018b). Although these structured presentations make reasoning more effective, they rely on sophisticated NLP pipelines and suffer from error propagation. More recently, end-to-end neural architectures have been successfully applied to text-

based QA, including Memory-augmented neural networks (Sukhbaatar et al., 2015; Miller et al., 2016; Kumar et al., 2016) and attention-based neural networks (Hermann et al., 2015; Chen et al., 2016; Kadlec et al., 2016; Dhingra et al., 2017; Xiong et al., 2017; Seo et al., 2017; Chen et al., 2017). In this work, we focus on QA over text (where the text is generated from a supporting KB) and evaluate several state-of-the-art memory-augmented and attention-based neural architectures on our QA task. In addition, we consider a sequence-to-sequence model baseline (Bahdanau et al., 2015), which has been widely used in dialog (Vinyals and Le, 2015; Ghazvininejad et al., 2017) and recently been applied to generating answer values from Wikidata (Hewlett et al., 2016).

There are numerous datasets available for evaluating the capabilities of QA systems. For example, MCTest (Richardson et al., 2013) contains comprehension questions for fictional stories. Allen AI Science Challenge (Clark, 2015) contains science questions that can be answered with knowledge from text books. RACE (Lai et al., 2017) is an English exam dataset for middle and high school Chinese students. MULTIRC (Khashabi et al., 2018a) is a dataset that focuses on evaluating multi-sentence reasoning skills. These datasets all require humans to carefully design multiple-choice questions and answers, so that certain aspects of the comprehension and reasoning capabilities are properly evaluated. As a result, it is difficult to collect them at scale. Furthermore, as the knowledge required for answering each question is not clearly specified in these datasets, it can be hard to identify the limitations of QA systems and propose improvements.

Weston et al. (2015) proposes to use synthetic QA tasks (the BABI dataset) to better understand the limitations of QA systems. BABI builds on a simulated physical world similar to interactive fiction (Montfort, 2005) with simple objects and relations and includes 20 different reasoning tasks. Various types of end-to-end neural networks (Sukhbaatar et al., 2015; Lee et al., 2015; Peng et al., 2015) have demonstrated promising accuracies on this dataset. However, the performance can hardly translate to real-world QA datasets, as BABI uses a small vocabulary (150 words) and short sentences with limited language variations (e.g., nesting sentences, coreference). A more sophisticated QA dataset with a supporting KB is WIKIMOVIES (Miller et al., 2016), which contains 100k questions about movies, each of them is answerable by using either a KB or a Wikipedia article. However, WIKIMOVIES is highly domain-specific, and similar to BABI, the questions are designed to be in simple forms with little compositionality and hence limit the difficulty level of the tasks.

Our dataset differs in the above datasets in that (i) it contains five different realistic domains permitting cross-domain evaluation to test the ability of models to generalize beyond a fixed set of KB relations, (ii) it exhibits rich referring expressions and linguistic variations (vocabulary much larger than the BABI dataset), (iii) questions in our dataset are designed to be deeply compositional and can cover multiple relations mentioned across multiple sentences.

Other large-scale QA datasets include Cloze-style datasets such as CNN/Daily Mail (Hermann et al., 2015), Children's Book Test (Hill et al., 2015), and Who Did What (Onishi et al., 2016); datasets with answers being spans in the document, such as SQuAD (Rajpurkar et al., 2016), NewsQA (Trischler et al., 2016), and TriviaQA (Joshi et al., 2017); and datasets with human generated answers, for instance, MS MARCO (Nguyen et al., 2016) and SearchQA (Dunn et al., 2017). One common drawback of these datasets is the difficulty in accessing a system's capability of integrating information across a document context. Kočiskỳ et al. (2017) recently emphasized this issue and proposed NarrativeQA, a dataset of fictional stories with questions that reflect the complexity of narratives: characters, events, and evolving relations. Our dataset contains similar narrative elements, but it is created with a supporting KB and hence it is easier to analyze and interpret results in a controlled setting.

3 TEXTWORLDS: Simulated Worlds for Multi-Relational QA from Narratives

In this work, we synthesize narratives in five diverse worlds, each containing a thousand narratives and where each narrative describes the evolution of a simulated user's world from a first-person perspective. In each narrative, the simulated user may introduce new facts, update existing facts or express a state change (e.g., *"Homework 3 is now due on Friday"* or *"Samantha passed her the-*

Statistics	Value
# of total stories	5,000
# of total questions	1,207,022
Avg. # of entity mentions (per story)	217.4
Avg. # of correct answers (per question)	8.7
Avg. # of facts in stories	100
Avg. # of tok. in stories	837.5
Avg. # of tok. in questions	8.9
Avg. # of tok. in answers	1.5
Vocabulary size (tok.)	1,994
Vocabulary size (entity)	10,793

Table 1: TEXTWORLDSQA dataset statistics

sis defense"). Each narrative is interleaved with questions about the current state of the world, and questions range in complexity depending on the amount of knowledge that needs to be integrated to answer them. This allows us to benchmark a range of QA models at their ability to answer questions that require different extents of relational reasoning to be answered.

The set of worlds that we simulate as part of this work are as follows:

1. MEETING WORLD: This world describes situations related to professional meetings, e.g., meetings being set/cancelled, people attending meetings, topics of meetings.

2. HOMEWORK WORLD: This world describes situations from the first-person perspective of a student, e.g., courses taken, assignments in different courses, deadlines of assignments.

3. SOFTWARE ENGINEERING WORLD: This world describes situations from the first-person perspective of a software development manager, e.g., task assignment to different project team members, stages of software development, bug tickets.

4. ACADEMIC DEPARTMENT WORLD: This world describes situations from the first-person perspective of a professor, e.g., teaching assignments, faculty going/returning from sabbaticals, students from different departments taking/dropping courses.

5. SHOPPING WORLD: This world describes situations about a person shopping for various occasions, e.g., adding items to a shopping list, purchasing items at different stores, noting where items are on sale.

3.1 Narrative

Each world is represented by a set of entities $\mathcal{E}$ and a set of unary, binary or ternary relations $\mathcal{R}$. Formally, a single step in one simulation of a world involves a combination of instantiating new entities and defining new (or mutating existing) relations between entities. Practically, we implement each world as a collection of classes and methods, with each step of the simulation creating or mutating class instances by sampling entities and methods on those entities. By design, these classes and methods are easy to extend, to either enrich existing worlds or create new ones. Each simulation step is then expressed as a natural language statement, which is added to the narrative. In the process of generating a natural language expression, we employ a rich mechanism for generating anaphora, such as *"meeting with John about the performance review"* and *"meeting that I last added"*, in addition to simple pronoun references. This allows us to generate more natural and flowing narratives. These references are generated and composed automatically by the underlying TEXTWORLDS framework, significantly reducing the effort needed to build new worlds. Furthermore, all generated stories also provide additional annotation that maps all entities to underlying gold-standard KB ids, allowing to perform experiments that provide models with different degrees of access to the "simulation oracle".

We generate 1,000 narratives within each world, where each narrative consists of 100 sentences, plus up to 300 questions interleaved randomly within the narrative. See Figure 1 for two example narratives. Each story in a given world samples its entities from a large general pool of entity names collected from the web (e.g., *people names*, *university names*). Although some entities do overlap between stories, each story in a given world contains a unique flow of events and entities involved in those events. See Table 1 for the data statistics.

3.2 Questions

Formally, questions are queries over the knowledge-base in the state defined up to the point when the question is asked in the narrative. In the narrative, the questions are expressed in natural language, employing the same anaphora mechanism used in generating the narrative (e.g., *"who is attending the last meeting I added?"*).

We categorize each question according to their

Dataset	Questions			
	Single Entity/Relation	Multiple entities		
		Single relation	Two relations	Three relations
MEETING	57,590 (41.16%)	46,373 (33.14%)	30,391 (21.72%)	5,569 (3.98%)
HOMEWORK	45,523 (24.10%)	17,964 (9.51%)	93,669 (49.59%)	31,743 (16.80%)
SOFTWARE	47,565 (20.59%)	51,302 (22.20%)	66,026 (28.58%)	66,150 (28.63%)
ACADEMIC	46,965 (24.81%)	54,581 (28.83%)	57,814 (30.53%)	29,982 (15.83%)
SHOPPING	111,522 (26.25%)	119,890 (28.22%)	107,418 (25.29%)	85,982 (20.24%)
All	309,165 (26.33%)	290,110 (24.71%)	355,318 (30.27%)	219,426 (18.69%)

Table 2: Dataset statistics by question type.

compositionality, broadly, as a proxy to the amount of information that needs to be integrated across the whole narrative in order to answer it. We categorize each question in our dataset into one of the following four categories:

Single Entity/Single Relation Answers to these questions are a single entity, e.g. *"what is John's email address?"*, or expressed in lambda-calculus notation:

$$\lambda x.\texttt{EmailAddress}(\texttt{John}, x)$$

The answers to these questions are found in a single sentence in the narrative, although it is possible that the answer may change through the course of the narrative (e.g., *"John's new office is GHC122"*).

Multi-Entity/Single Relation Answers to these questions can be multiple entities but involve a single relation, e.g., *"Who is enrolled in the Math class?"*, or expressed in lambda calculus notation:

$$\lambda x.\texttt{TakingClass}(x, \texttt{Math})$$

Unlike the previous category, answers to these questions can be sets of entities.

Multi-Entity/Two Relations Answers to these questions can be multiple entities and involve two relations, e.g., *"Who is enrolled in courses that I am teaching?"*, or expressed in lambda calculus:

$$\lambda x.\exists y.\texttt{EnrolledInClass}(x, y) \\ \wedge \texttt{CourseTaughtByMe}(y)$$

Multi-Entity/Three Relations Answers to these questions can be multiple entities and involve three relations, e.g., *"Which undergraduates are enrolled in courses that I am teaching?"*, or expressed in lambda calculus notation:

$$\lambda x.\exists y.\texttt{EnrolledInClass}(x, y) \\ \wedge \texttt{CourseTaughtByMe}(y) \\ \wedge \texttt{Undergrad}(x)$$

In the data that we generate, answers to questions are always sets of spans in the narrative (the reason for this constraint is for easier evaluation of several existing machine-reading models; this assumption can easily be relaxed in the simulation). In all of our evaluations, we will partition our results by one of the four question categories listed above, which we hypothesize correlates with the difficulty of a question.

4 Methods

We develop several baselines for our QA task, including a logistic regression model and four different neural network models: Seq2Seq (Bahdanau et al., 2015), MemN2N (Sukhbaatar et al., 2015), BiDAF (Seo et al., 2017), and DrQA (Chen et al., 2017). These models generate answers in different ways, e.g., predicting a single entity, predicting spans of text, or generating answer sequences. Therefore, we implement two experimental settings: ENTITY and RAW. In the ENTITY setting, given a question and a story, we treat all the entity spans in the story as candidate answers, and the prediction task becomes a classification problem. In the RAW setting, a model needs to predict the answer spans. For logistic regression and MemN2N, we adopt the ENTITY setting as they are naturally classification models. This ideally provides an upper bound on the performance when considering answer candidate generation. For all the other models, we can apply the RAW setting.

4.1 Logistic Regression

The logistic regression baseline predicts the likelihood of an answer candidate being a true answer. For each answer candidate e and a given question, we extract the following features: (1) The frequency of e in the story; (2) The number of words within e; (3) Unigrams and bigrams within e; (4) Each non-stop question word combined with each non-stop word within e; (5) The average minimum

distance between each non-stop question word and e in the story; (6) The common words (excluding stop words) between the question and the text surrounding of e (within a window of 10 words); (7) Sum of the frequencies of the common words to the left of e, to the right e, and both. These features are designed to help the model pick the correct answer spans. They have shown to be effective for answer prediction in previous work (Chen et al., 2016; Rajpurkar et al., 2016).

We associate each answer candidate with a binary label indicating whether it is a true answer. We train a logistic regression classifier to produce a probability score for each answer candidate. During test, we search for an optimal threshold that maximizes the F1 performance on the validation data. During training, we optimize the cross-entropy loss using Adam (Kingma and Ba, 2014) with an initial learning rate of 0.01. We use a batch size of $10,000$ and train with 5 epochs. Training takes roughly 10 minutes for each domain on a Titan X GPU.

4.2 Seq2Seq

The seq2seq model is based on the sequence to sequence model presented in (Bahdanau et al., 2015), which includes an attention model. Bahdanau et al. (Bahdanau et al., 2015) have used this model to build a neural based machine translation performing at the state-of-the-art. We adopt this model to fit our own domain by including a preprocessing step in which all facts are concatenated with a dedicated token, while eliminating all previously asked questions, and the current question is added at the end of the list of facts. The answers are treated as a sequence of words. We use word embeddings (Zou et al., 2013), as it was shown to improve accuracy. We use 3 GRU (Cho et al., 2014) connected layers, each with a capacity of 256. Our batch size was set to 16. We use gradient descent with an initial learning rate of 0.5 and a decay factor of 0.99. We iterated on the data for a total of $50,000$ steps (roughly 5 epochs). The training process for each domain took approximately 48 hours on a Titan X GPU.

4.3 MemN2N

End-To-End Memory Network (MemN2N) is a neural architecture that encodes both long-term and short-term context into a memory and iteratively reads from the memory (i.e., multiple hops) relevant information to answer a question (Sukhbaatar et al., 2015). It has been shown to be effective for a variety of question answering tasks (Weston et al., 2015; Sukhbaatar et al., 2015; Hill et al., 2015).

In this work, we directly apply MemN2N to our task with a small modification. Originally, MemN2N was designed to produce a single answer for a question, so at the prediction layer, it uses softmax to select the best answer from the answer candidates. In order to account for multiple answers for a given question, we modify the prediction layer to apply the logistic function and optimize the cross entropy loss instead. For training, we use the implementation setting in a publicly available MemN2N package [1]. We train the model for 100 epochs and it takes about 2 hours for each domain on a Titan X GPU.

4.4 BiDAF-M

BiDAF (Bidirectional Attention Flow Networks) (Seo et al., 2017) is one of the top-performing models on the span-based question answering dataset SQuAD (Rajpurkar et al., 2016). We reimplement BiDAF with simplified parameterizations and change the prediction layer so that it can predict multiple answer spans.

Specifically, we encode the input story $\{x_1, ..., x_T\}$ and a given question $\{q_1, ..., q_J\}$ at the character level and the word level, where the character level uses CNNs and the word level uses pre-trained word vectors. The concatenation of the character and word embeddings are passed to a bidirectional LSTM to produce a contextual embedding for each word in the story context and in the question. Then, we apply the same bidirectional attention flow layer to model the interactions between the context and question embeddings, producing question-aware feature vectors for each word in the context, denoted as $\mathbf{G} \in \mathbb{R}^{d_g \times T}$. $\mathbf{G}$ is then fed into a bidirectional LSTM layer to obtain a feature matrix $\mathbf{M}_1 \in \mathbb{R}^{d_1 \times T}$ for predicting the start offset of the answer span, and $\mathbf{M}_1$ is then passed into another bidirectional LSTM layer to obtain a feature matrix $\mathbf{M}_2 \in \mathbb{R}^{d_2 \times T}$ for predicting the end offset of the answer span. We then compute two probability scores for each word i in the narrative: $\mathbf{p}^{start} = \text{sigmoid}(\mathbf{w}_1^T[\mathbf{G}; \mathbf{M}_1])$ and $\mathbf{p}^{end} = \text{sigmoid}(\mathbf{w}_2^T[\mathbf{G}; \mathbf{M}_1; \mathbf{M}_2])$, where $\mathbf{w}_1$ and $\mathbf{w}_2$ are trainable weights. The training objec-

[1] https://github.com/domluna/memn2n

Within-World	MEETING	HOMEWORK	SOFTWARE	DEPARTMENT	SHOPPING	Avg. F1
Logistic Regression	50.1	55.7	60.9	55.9	61.1	56.7
Seq2Seq	22.5	32.6	16.7	39.1	31.5	28.5
MemN2N	55.4	46.6	69.5	67.3	46.3	57.0
BiDAF-M	81.8	76.9	68.4	68.2	68.7	72.8
DrQA-M	81.2	83.6	79.1	76.4	76.5	79.4
Cross-World	MEETING	HOMEWORK	SOFTWARE	DEPARTMENT	SHOPPING	Avg. F1
Logistic Regression	9.0	9.1	11.1	9.9	7.2	9.3
Seq2Seq	8.8	3.5	1.9	5.4	2.6	4.5
MemN2N	23.6	2.9	4.7	14.6	0.07	9.2
BiDAF-M	34.0	6.9	16.1	22.2	3.9	16.6
DrQA-M	46.5	12.2	23.1	28.5	9.3	23.9

Table 3: F_1 scores for different baselines evaluated on both *within-world* and *across-world* settings.

tive is simply the sum of cross-entropy losses for predicting the start and end indices.

We use 50 1D filters for CNN character embedding, each with a width of 5. The word embedding size is 300 and the hidden dimension for LSTMs is 128. For optimization, we use Adam (Kingma and Ba, 2014) with an initial learning rate of 0.001, and use a minibatch size of 32 for 15 epochs. The training process takes roughly 20 hours for each domain on a Titan X GPU.

4.5 DrQA-M

DrQA (Chen et al., 2017) is an open-domain QA system that has demonstrated strong performance on multiple QA datasets. We modify the Document Reader component of DrQA and implement it in a similar framework as BiDAF-M for fair comparisons. First, we employ the same character-level and word-level encoding layers to both the input story and a given question. We then use the concatenation of the character and word embeddings as the final embeddings for words in the story and in the question. We compute the aligned question embedding (Chen et al., 2017) as a feature vector for each word in the story and concatenate it with the story word embedding and pass it into a bidirectional LSTM to obtain the contextual embeddings $\mathbf{E} \in \mathbb{R}^{d \times T}$ for words in the story. Another bidirectional LSTM is used to obtain the contextual embeddings for the question, and self-attention is used to compress them into one single vector $\mathbf{q} \in \mathbb{R}^d$. The final prediction layer uses a bilinear term to compute scores for predicting the start offset: $\mathbf{p}^{start} = \text{sigmoid}(\mathbf{q}^T \mathbf{W}_1 \mathbf{E})$ and another bilinear term for predicting the end offset: $\mathbf{p}^{end} = \text{sigmoid}(\mathbf{q}^T \mathbf{W}_2 \mathbf{E})$, where $\mathbf{W}_1$ and $\mathbf{W}_2$ are trainable weights. The training loss is the same as in BiDAF-M, and we use the same parameter setting. Training takes roughly 10 hours for each domain on a Titan X GPU.

5 Experiments

We use two evaluation settings for measuring performance at this task: *within-world* and *across-world*. In the *within-world* evaluation setting, we test on the same world that the model was trained on. We then compute the precision, recall and F_1 for each question and report the macro-average F1 score for questions in each world. In the *across-world* evaluation setting, the model is trained on four out of the five worlds, and tested on the remaining world. The *across-world* regime is obviously more challenging, as it requires the model to be able to learn to generalize to unseen relations and vocabulary. We consider the *across-world* evaluation setting to be the main evaluation criteria for any future models used on this dataset, as it mimics the practical requirement of any QA system used in personal assistants: it has to be able to answer questions on any new domain the user introduces to the system.

5.1 Results

We draw several important observations from our results. First, we observe that more compositional questions (i.e., those that integrate multiple relations) are more challenging for most models - as all models (except Seq2seq) decrease in performance with the number of relations composed in a question (Figure 5.1). This can be in part explained by the fact that more composition questions are typically longer, and also require the model to integrate more sources of information in the narrative in order to answer them. One surpris-

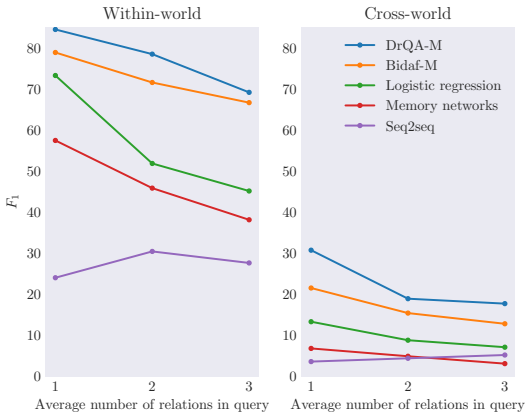

Figure 3: F_1 score breakdown based on the number of relations involved in the questions.

ing observation from our results is that the performance on questions that ask about a single relation and have only a single answer is lower than questions that ask about a single relation but that can have multiple answers (see detailed results in the Appendix). This is in part because questions that can have multiple answers typically have canonical entities as answers (e.g., person's name), and these entities generally repeat in the text, making it easier for the model to find the correct answer.

Table 3 reports the overall (macro-average) F1 scores for different baselines. We can see that BiDAF-M and DrQA-M perform surprisingly well in the *within-world* evaluation even though they do not use any entity span information. In particular, DrQA-M outperforms BiDAF-M which suggests that modeling question-context interactions using simple bilinear terms have advantages over using more complex bidirectional attention flows. The lower performance of MemN2N suggests that its effectiveness on the BABI dataset does not directly transfer to our dataset. Note that the original MemN2N architecture uses simple bag-of-words and position encoding for sentences. This may work well on dataset with a simple vocabulary, for example, MemN2N performs better in the SOFTWARE world compared to other worlds as the SOFTWARE world has a much smaller vocabulary. In general, we believe that better text representations for questions and narratives can lead to improved performance. Seq2Seq model also did not perform as well. This is due to the inherent difficulty of generation and encoding long sequences. We found that it performs better when training and testing on shorter stories (limited to 30 facts). Interestingly, the logistic regression baseline outperforms Seq2Seq and MemN2N, but there is still a large performance gap to BiDAF-M and DrQA-M, and the gap is greater for questions that compose multiple relations.

In the *across-world* setting, the performance of all methods dramatically decreases.[2] This suggests the limitations of these methods in generalizing to unseen relations and vocabulary. The span-based models BiDAF-M and DrQA-M have an advantage in this setting as they can learn to answer questions based on the alignment between the question and the narrative. However, the low performance still suggests their limitations in transferring question answering capabilities.

6 Conclusion

In this work, we have taken the first steps towards the task of multi-relational question answering expressed through personal narrative. Our hypothesis is that this task will become increasingly important as users begin to teach personal knowledge about their world to the personal assistants embedded in their devices. This task naturally synthesizes two main branches of question answering research: QA over KBs and QA over free text. One of our main contributions is a collection of diverse datasets that feature rich compositional questions over a dynamic knowledge graph expressed through simulated narrative. Another contribution of our work is a thorough set of experiments and analysis of different types of end-to-end architectures for QA at their ability to answer multi-relational questions of varying degrees of compositionality. Our long-term goal is that both the data and the simulation code we release will inspire and motivate the community to look towards the vision of letting end-users teach our personal assistants about the world around us.

References

David Ahn, Valentin Jijkoun, Gilad Mishne, Karin Müller, Maarten de Rijke, Stefan Schlobach, M Voorhees, and L Buckland. 2004. Using wikipedia at the trec qa track. In *TREC*. Citeseer.

[2] In order to allow generalization across different domains for the Seq2Seq model, we replace entities appearing in each story with an id that correlates to their appearance order. After the model outputs its prediction, the entity ids are converted back to the entity phrase.

Dzmitry Bahdanau, Kyunghyun Cho, and Yoshua Bengio. 2015. Neural machine translation by jointly learning to align and translate. In *ICLR*.

Michele Banko, Eric Brill, Susan Dumais, and Jimmy Lin. 2002. Askmsr: Question answering using the worldwide web. In *Proceedings of 2002 AAAI Spring Symposium on Mining Answers from Texts and Knowledge Bases*, pages 7–9.

Jonathan Berant, Andrew Chou, Roy Frostig, and Percy Liang. 2013. Semantic parsing on freebase from question-answer pairs. In *Proceedings of the 2013 Conference on Empirical Methods in Natural Language Processing*, pages 1533–1544.

Danqi Chen, Jason Bolton, and Christopher D Manning. 2016. A thorough examination of the cnn/daily mail reading comprehension task. In *ACL*.

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

Kyunghyun Cho, Bart Van Merriënboer, Caglar Gulcehre, Dzmitry Bahdanau, Fethi Bougares, Holger Schwenk, and Yoshua Bengio. 2014. Learning phrase representations using rnn encoder-decoder for statistical machine translation. In *EMNLP*.

Peter Clark. 2015. Elementary school science and math tests as a driver for ai: take the aristo challenge! In *AAAI*, pages 4019–4021.

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

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 preprint arXiv:1704.05179*.

Marjan Ghazvininejad, Chris Brockett, Ming-Wei Chang, Bill Dolan, Jianfeng Gao, Wen-tau Yih, and Michel Galley. 2017. A knowledge-grounded neural conversation model. In *AAAI*.

Karl Moritz Hermann, Tomas Kocisky, 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 *ACL*.

Felix Hill, Antoine Bordes, Sumit Chopra, and Jason Weston. 2015. The goldilocks principle: Reading children's books with explicit memory representations. *arXiv preprint arXiv:1511.02301*.

Mandar Joshi, Eunsol Choi, Daniel S Weld, and Luke Zettlemoyer. 2017. Triviaqa: A large scale distantly supervised challenge dataset for reading comprehension. In *ACL*.

Rudolf Kadlec, Martin Schmid, Ondrej Bajgar, and Jan Kleindienst. 2016. Text understanding with the attention sum reader network. In *ACL*.

Daniel Khashabi, Snigdha Chaturvedi, Michael Roth, Shyam Upadhyay, and Dan Roth. 2018a. Looking beyond the surface: A challenge set for reading comprehension over multiple sentences. In *NAACL*.

Daniel Khashabi, Tushar Khot Ashish Sabharwal, and Dan Roth. 2018b. Question answering as global reasoning over semantic abstractions. In *AAAI*.

Tushar Khot, Ashish Sabharwal, and Peter Clark. 2017. Answering complex questions using open information extraction. In *ACL*.

Diederik P Kingma and Jimmy Ba. 2014. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*.

Tomáš Kočiskỳ, Jonathan Schwarz, Phil Blunsom, Chris Dyer, Karl Moritz Hermann, Gábor Melis, and Edward Grefenstette. 2017. The narrativeqa reading comprehension challenge. *arXiv preprint arXiv:1712.07040*.

Ankit Kumar, Ozan Irsoy, Peter Ondruska, Mohit Iyyer, James Bradbury, Ishaan Gulrajani, Victor Zhong, Romain Paulus, and Richard Socher. 2016. Ask me anything: Dynamic memory networks for natural language processing. In *International Conference on Machine Learning*, pages 1378–1387.

Tom Kwiatkowski, Eunsol Choi, Yoav Artzi, and Luke Zettlemoyer. 2013. Scaling semantic parsers with on-the-fly ontology matching. In *Proceedings of the 2013 Conference on Empirical Methods in Natural Language Processing*, pages 1545–1556.

Guokun Lai, Qizhe Xie, Hanxiao Liu, Yiming Yang, and Eduard Hovy. 2017. Race: Large-scale reading comprehension dataset from examinations. In *EMNLP*.

Moontae Lee, Xiaodong He, Wen-tau Yih, Jianfeng Gao, Li Deng, and Paul Smolensky. 2015. Reasoning in vector space: An exploratory study of question answering. *arXiv preprint arXiv:1511.06426*.

Alexander Miller, Adam Fisch, Jesse Dodge, Amir-Hossein Karimi, Antoine Bordes, and Jason Weston. 2016. Key-value memory networks for directly reading documents. *arXiv preprint arXiv:1606.03126*.

Nick Montfort. 2005. *Twisty Little Passages: an approach to interactive fiction*. Mit Press.

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. *arXiv preprint arXiv:1611.09268*.

Takeshi Onishi, Hai Wang, Mohit Bansal, Kevin Gimpel, and David McAllester. 2016. Who did what: A large-scale person-centered cloze dataset. *arXiv preprint arXiv:1608.05457*.

Baolin Peng, Zhengdong Lu, Hang Li, and Kam-Fai Wong. 2015. Towards neural network-based reasoning. *arXiv preprint arXiv:1508.05508*.

Pranav Rajpurkar, Jian Zhang, Konstantin Lopyrev, and Percy Liang. 2016. Squad: 100,000+ questions for machine comprehension of text. In *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*, pages 193–203.

Mrinmaya Sachan, Kumar Dubey, Eric Xing, and Matthew Richardson. 2015. Learning answer-entailing structures for machine comprehension. In *Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics and the 7th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, volume 1, pages 239–249.

Mrinmaya Sachan and Eric Xing. 2016. Machine comprehension using rich semantic representations. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, volume 2, pages 486–492.

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

Sainbayar Sukhbaatar, Jason Weston, Rob Fergus, et al. 2015. End-to-end memory networks. In *Advances in neural information processing systems*, pages 2440–2448.

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

Oriol Vinyals and Quoc Le. 2015. A neural conversational model. *arXiv preprint arXiv:1506.05869*.

Jason Weston, Antoine Bordes, Sumit Chopra, Alexander M Rush, Bart van Merriënboer, Armand Joulin, and Tomas Mikolov. 2015. Towards ai-complete question answering: A set of prerequisite toy tasks. *arXiv preprint arXiv:1502.05698*.

Caiming Xiong, Victor Zhong, and Richard Socher. 2017. Dynamic coattention networks for question answering. In *ICLR*.

Scott Wen-tau Yih, Ming-Wei Chang, Xiaodong He, and Jianfeng Gao. 2015. Semantic parsing via staged query graph generation: Question answering with knowledge base. In *ACL*.

John M Zelle and Raymond J Mooney. 1996. Learning to parse database queries using inductive logic programming. In *Proceedings of the national conference on artificial intelligence*, pages 1050–1055.

Luke S Zettlemoyer and Michael Collins. 2012. Learning to map sentences to logical form: Structured classification with probabilistic categorial grammars. *arXiv preprint arXiv:1207.1420*.

Will Y Zou, Richard Socher, Daniel Cer, and Christopher D Manning. 2013. Bilingual word embeddings for phrase-based machine translation. In *Proceedings of the 2013 Conference on Empirical Methods in Natural Language Processing*, pages 1393–1398.

Dataset	Questions											
	Single Entity/Relation			Multiple Entities								
				Single Relation			Two Relations			Three Relations		
	P	R	F_1	P	R	F_1	P	R	F_1	P	R	F_1
Logistic Regression												
MEETING	42.0	78.1	51.0	50.6	74.6	56.6	33.3	66.3	41.1	31.8	57.6	38.0
HOMEWORK	39.7	57.8	44.2	98.6	99.1	98.8	57.4	78.7	62.2	25.4	42.0	28.0
SOFTWARE	55.0	73.3	59.0	54.3	98.2	66.5	58.2	76.0	62.3	46.3	84.6	56.4
DEPARTMENT	42.6	65.9	48.0	59.0	82.5	65.1	38.8	52.7	41.2	42.5	64.6	46.9
SHOPPING	53.1	70.2	56.2	79.6	83.4	79.0	53.1	60.5	52.3	53.4	67.9	56.0
Average	46.5	69.1	51.7	68.4	87.6	73.2	48.2	66.8	51.8	39.9	63.3	45.1
Sequence-to-Sequence												
MEETING	27.9	18.3	22.1	48.1	12.1	19.3	42.1	15.0	22.1	33.7	19.7	24.8
HOMEWORK	16.3	9.0	11.6	71.9	9.3	16.4	75.3	35.9	48.6	32.9	15.6	21.1
SOFTWARE	42.5	21.5	28.5	44.8	8.5	14.2	50.0	6.3	11.2	45.5	7.4	12.7
DEPARTMENT	49.9	35.6	41.5	54.1	20.3	29.6	57.2	38.0	45.7	43.9	39.7	41.7
SHOPPING	25.8	16.0	19.8	71.3	28.2	40.5	33.3	19.3	24.4	46.9	31.4	37.6
Average	32.5	20.1	24.7	58.0	15.7	24.0	51.6	22.9	30.4	40.6	22.7	27.6
MemN2N												
MEETING	56.9	56.0	54.7	66.8	58.4	58.6	57.0	57.5	54.8	38.7	40.7	38.8
HOMEWORK	42.6	41.2	41.3	97.9	63.7	73.9	60.4	47.9	49.4	36.5	29.0	30.1
SOFTWARE	68.5	71.6	68.5	72.9	73.2	70.9	69.7	67.3	66.1	75.0	74.8	72.6
DEPARTMENT	56.3	74.3	61.3	78.5	87.0	80.2	59.4	76.6	63.2	57.8	74.2	61.6
SHOPPING	51.3	45.4	45.5	74.9	54.1	59.0	45.6	40.6	40.2	44.3	37.6	37.9
Average	55.1	57.7	54.3	78.2	67.3	68.5	58.4	58.0	54.8	50.4	51.3	48.2
BIDAF-M												
MEETING	87.6	92.4	88.2	78.6	86.1	79.2	68.9	89.6	74.6	73.9	94.4	80.0
HOMEWORK	79.9	97.4	84.5	86.8	81.0	82.4	76.4	90.0	78.9	47.0	78.5	55.5
SOFTWARE	48.0	89.4	57.4	68.5	93.6	75.8	62.4	86.1	67.5	62.7	90.9	71.3
DEPARTMENT	57.0	64.6	58.1	73.6	85.9	76.6	67.0	83.2	70.8	63.1	71.4	64.0
SHOPPING	60.5	87.1	66.9	76.7	90.9	79.8	57.1	89.0	65.8	53.2	88.5	62.0
Average	66.6	86.2	71.0	76.8	87.5	78.8	66.4	87.6	71.5	60.0	84.7	66.6
DrQA-M												
MEETING	77.1	94.2	81.0	80.6	95.8	85.1	68.6	95.7	76.8	64.1	97.9	74.3
HOMEWORK	88.8	97.9	91.4	85.2	80.2	81.4	85.0	94.7	87.9	51.6	85.8	60.2
SOFTWARE	72.7	96.0	78.9	78.6	93.3	82.7	79.4	89.4	80.9	66.3	93.2	74.5
DEPARTMENT	67.1	97.9	76.1	80.3	95.0	84.1	67.1	94.4	74.8	55.8	95.2	66.9
SHOPPING	71.5	93.9	77.7	86.4	94.8	88.7	62.8	91.1	71.4	62.4	90.7	69.7
Average	75.4	96.0	81.0	82.2	91.8	84.4	72.6	93.1	78.4	60.0	92.6	69.1

Table 4: Test performance at the task of question answering by question type using the *within-world* evaluation.

Dataset	Questions			
	Single Entity/Relation	Across Entities		
		Single Relation	Two Relations	Three Relations
Logistic Regression				
MEETING	8.8	10.9	7.2	5.6
HOMEWORK	7.5	20.2	8.5	6.7
SOFTWARE	8.2	12.0	12.9	10.6
DEPARTMENT	7.4	14.4	9.7	6.1
SHOPPING	8.2	9.0	5.9	6.6
Average	8.0	13.3	8.8	7.1
Sequence-to-Sequence				
MEETING	7.4	8.1	10.0	14.0
HOMEWORK	4.2	2.9	3.1	2.3
SOFTWARE	5.0	0.6	0.9	1.1
DEPARTMENT	5.5	4.0	5.6	5.6
SHOPPING	2.5	2.6	2.3	2.8
Average	4.9	3.6	4.4	5.2
MemN2N				
MEETING	9.0	34.2	33.0	27.4
HOMEWORK	3.3	12.4	1.0	2.5
SOFTWARE	13.4	0.8	3.2	2.9
DEPARTMENT	12.9	20.8	13.0	9.4
SHOPPING	0.1	0.07	0.05	0.03
Average	7.8	13.7	10.1	8.4
BIDAF-M				
MEETING	31.1	40.2	30.4	30.0
HOMEWORK	10.4	20.3	2.3	7.8
SOFTWARE	19.2	13.4	22.7	9.1
DEPARTMENT	23.3	30.5	19.0	13.5
SHOPPING	5.6	3.2	2.6	3.4
Average	17.9	21.5	15.4	12.8
DrQA-M				
MEETING	44.5	58.8	33.3	37.1
HOMEWORK	19.8	30.1	5.9	9.4
SOFTWARE	26.4	23.4	24.0	19.4
DEPARTMENT	31.0	38.8	24.4	15.7
SHOPPING	19.3	2.3	6.7	7.1
Average	28.2	30.7	18.9	17.7

Table 5: Test performance (F_1 score) at the task of question answering by question type using the *across-world* evaluation.