Distribution Agreement

Zhengzhe Yang April 15, 2019

FriendsQA: Open-Domain Question Answering Dataset on TV Show Transcripts

by

Zhengzhe Yang

Jinho D. Choi
Adviser

Department of Computer Science

Jinho D. Choi

Adviser

Shun Yan Cheung

Committee Member

Jed Brody

Committee Member

2019

FriendsQA: Open-Domain Question Answering Dataset on TV Show Transcripts

By

Zhengzhe Yang

Jinho D. Choi

Adviser

An abstract of
a thesis submitted to the Faculty of Emory College of Arts and Sciences
of Emory University in partial fulfillment
of the requirements of the degree of
Bachelor of Sciences with Honors

Department of Computer Science

2019

Abstract

FriendsQA: Open-Domain Question Answering Dataset on TV Show Transcripts

By Zhengzhe Yang

This thesis presents FriendsQA, a challenging question answering dataset that contains 1,222 dialogues and 10,610 open-domain questions, to tackle machine comprehension on everyday conversations. Each dialogue, involving multiple speakers, is annotated with six types of questions {*what, when, why, where, who, how*} regarding the dialogue contexts, and the answers are annotated with contiguous spans in the dialogue. A series of crowdsourcing tasks are conducted to ensure good annotation quality, resulting a high inter-annotator agreement of 81.82%. A comprehensive annotation analytics is provided for a deeper understanding in this dataset. Three state-of-the-art QA systems are experimented, R-Net, QANet, and BERT, and evaluated on this dataset. BERT in particular depicts promising results, an accuracy of 74.2% for answer utterance selection and an F1-score of 64.2% for answer span selection, suggesting that the FriendsQA task is hard yet has a great potential of elevating QA research on multiparty dialogue to another level.

FriendsQA: Open-Domain Question Answering Dataset on TV Show Transcripts

By

Zhengzhe Yang

Jinho D. Choi

Adviser

A thesis submitted to the Faculty of Emory College of Arts and Sciences
of Emory University in partial fulfillment
of the requirements of the degree of
Bachelor of Sciences with Honors

Department of Computer Science

2019

Acknowledgements

Firstly, I would like to express my sincere gratitude to Dr. Jinho D. Choi. We first met at the start of junior year and he introduced me to the NLP world which I never thought I would love this much. He gave me the appreciation I really needed, especially when I was frustrated with internship hunts and academic challenges. His recognition and acceptance changed my life and helped me discover the inner self who I really am. His strictness in research and passion toward NLP did and will continue motivating me to work harder. I want to thank him for his insightful opinions, helpful input and painstaking efforts during my first research, first conference paper and first thesis, and this truly means a lot to me.

I would also like to thank my committee members, Dr. Shun Yan Cheung and Dr. Jed Brody, who have also helped me improve this thesis. They are wonderful professors both inside and outside classroom. Moreover, it is an honor to serve as Teaching Assistant for Dr. Cheung.

Lastly, I would like to thank all my friends in Emory NLP lab, Shen Gao, Gary Lai, Xinyi Jiang, Kate Li, Liyan Xu, Han He and James Finch, who have always been there when I needed help. The productive discussions and research meetings broadened my insights and deepened my understand toward NLP. I also would like to thank my friend Ying Wu, a student at UC San Diego who have supported me mentally, emotionally and academically.

Contents

List of Figures

List of Tables

Chapter 1

Introduction

Question answering (QA) is receiving a lot of attention over the recent years as neural network models have been consistently pushing the limit of machine comprehension to the level of human intelligence. QA is the task to challenge machines ability to understand a document and later apply the learned knowledge to answer to queries, either by completing a blank, selecting from a pool of available candidates, or pinpoint certain answer spans in the given text. Over the years, a lot of challenging and robust question answering datasets have appeared and gained a lot of interests. While numerous models have shown remarkable results with these datasets, the evidence passages, from which the answers are derived, mostly reside within wiki articles, newswire, (non-)fictional stories, or children's books, but not from multiparty dialogue. No system has demonstrated its ability to comprehend and respond to queries in human-to-human conversations, although it is the most natural means of communication. Moreover, the amount of data in this form has increased at a faster rate than any other type of textual data [17, 6].

The ultimate goal of QA systems is to revolutionize how humans and machine interact and respond to open-domain questions in an accurate and efficient manner. However, contextual understanding in dialogue, the most critical requirement to achieve such goal, is challenging because it needs to comprehend the contents composed by multiple speakers, and anticipate colloquial language filled with sarcasms, metaphors, humors, etc. This inspires us to create an open-domain Question Answering dataset, **FriendsQA**, that aims to enhance machine comprehension on this domain and facilitate the studies in informal question answering. Similar to other open-domain QA datasets, each dialogue will serve as context information, followed by a number of questions with their answers annotated in the context. An example of questions, answers and context pair can be found in Table 1.1. Dialogues in this dataset are excerpted from the TV show *Friends*, a beloved TV series world wide and also the go-to show for English learners since the expressions and vocabulary used are easy to understand. However, this is not the case for machines because the informal grammar, rhetoric questions and emotions conveyed in utterances can be more abundant and variant. Sarcasm, humors,

. . . . . .
u004 Interviewer And if I want to call for a reference on your last job ?
u005 Monica_Geller Oh , that 's there on the bottom , see the manager , Chandler Bing .
u006 Interviewer Alright , let s see if you 're as good in person as you are on paper . Make me **a salad** .
u007 Monica_Geller A salad ? Really I , **I could do something a little more complicated if you like** .
. . . . . .

1. What does the interviewer want Monica to make ?
a salad
2. Who is the person Monica uses as a reference ?
Chandler Bing
3. Why is Monica surprised by what the interviewer wants her to make ?
I could do something a little more complicated if you like

Table 1.1: An example of context-question-answer pair in FriendsQA

and dramatic exaggeration can make the processing of extracting evidence and features harder comparing to formal writings like those of SQuAD (Stanford Questiong Answering Dataset) [22]. Further more, in natural conversations, characters are referred to interchangeably with pronouns, short description (Paul the Wine Guy), and nicknames, which is very likely for the machine to get confused in the exchange of utterances.

To begin with, Chapter 2 will picture the current trend in the research of Question Answering and related works including other QA datasets and state-of-the-art neural models. Any foundation work prior and related to this dataset will be discussed. A comparison between FriendsQA with other datasets will also be presented to show its distinctness.

Chapter 3 will give examples and statistics of FriendsQA that proves the proposed FriendsQA can serve as a rich Question Answering research resource. The creation process of this dataset, quality assurance procedures and the follow-up analysis will be explained in details to show its validity, difficulty and diversity.

Chapter 4 will explain different approaches (R-Net, QANet and BERT) and directions to take the full advantage of FriendsQA in order to yield meaningful and insightful results. Configurations for each system will also be given for the results to be replicated.

Chapter 5 will report all the evaluation metrics, the input format and the results for all approaches attempted. The results, including running and comparing SOTA systems, evaluating top-k answer candidates, utterance prediction and co-reference replacements, will be presented in figures, charts and tables to visually demonstrate FriendsQA's potential as an open-domain QA dataset.

Chapter 6 wraps up all the contributions I have made to the research community in NLP and the field of Question Answering in particular. Future paths on this dataset will also be suggested for researchers who share mutual interests.

Chapter 2

Background

We begin with listing works that have been done in the field of Question Answering. First, existing QA datasets (Section 2.1) will be discussed and compared since these datasets are studied the most and a lot of powerful models have been constructed to test on them. Then, we will present the well-established approaches (Section 2.2) that are proposed to solve relevant tasks. The foundation of this work and previous annotations will be discussed (Section 2.3). Finally, the distinction needs to be drawn between FriendsQA with similar dataset to show its uniqueness and potential (Section 2.4).

2.1 QA Datasets

The NLP community has been striving to propose Question Answering (QA) datasets that fall into three categories: reading comprehension QA, cloze-style QA and span-based QA, all of which are studied enthusiastically.

Reading comprehension QA requires the model to pick an answer from an available pool of answers after comprehending an evidence passage, similar to multiple choice questions. Important and relevant information must be learned attentively and accurately in order to predict the correct answer. Following is a list of datasets proposed with this format. MCTest is an open-domain dataset comprising short fictional stories [24]. RACE is a large dataset compiled from English assessments for 12-18 years old students [13]. TQA gives passages from middle school science lessons and textbooks [12]. SciQ gives passages from science exams collected via crowdsourcing [31]. DREAM gives multiparty dialogue passages from English-as-a-foreign-language exams [27].

The second is for cloze-style QA, for which the model fills in the blanks that obliterate certain contents in sentences describing the evidence passages. This task is challenging because it requires both context understanding surrounding the blank and a general but comprehensive understanding toward the evidence passage, since normally the prediction happens within a summary of the passage. Such style is important because it is also a popular way to test people's English skills and reading comprehension in a passage. CNN/Daily Mail targets on entities in bullet points summarizing articles from *CNN* and *Daily News* [7]. Children's Book Test focuses on named entities, nouns, verbs, and prepositions in passages from children's books [8]. BookTest is similar to Children's Book Test but 60 times larger [1]. Who-did-What gives description sentences and evidence passages extracted from news articles in English Gigaword Corpus [19].

Finally, span-based QA is a task in which the model finds the answer contents as spans in the evidence passages.This task is the hardest because giving an answer span resembles humans' way of answering questions. It also challenges the level of understanding of the evidence passages the greatest. bAbI aims to reinforce learning on event types and infer a sequence of event descriptions [32]. WikiQA [33] and SQuAD [22] use Wikipedia, whereas NewsQA [28] use CNN articles as evidence passages. MS MARCO gives questions involving zero to multiple answer contents from web documents [18]. TriviaQA is compiled by trivia enthusiasts to challenge machine comprehension [11]. CoQA focuses on conversational flows between a questioner and an answerer [23].

2.2 QA Systems

It has become a common understanding that RNN (Recurrent Neural Networks), CNN (Convolutional Neural Network) and miscellaneous attention mechanism are required to ensure competitive performance on QA datasets. A lot of neural models have been presented to demonstrate remarkable results on the datasets in Section 2.1. R-Net [30] uses gated attention-based recurrent networks and refines QA representation with self-matching attention. ReasoNet [26] takes multiple turns to reason over the relationships between query, documents, and answers. Attention Over Attention Reader [4] is designed to better capture the similarities between questions and answer contents. Reinforced Mnemonic Reader [9] combines the memorized attention with new attention. Self-attention [29] is solely applied to Question Answering tasks, which became known as the Transformer. Multi-layer Embedding with Memory Network (MEMEN) [20] captures better embeddings and document-query pair information. FusionNet [10] keeps the history of word representations and used multi-level attention. Novel and rich contextualized word representations[25] is also used with a standard neural architecture instead of focusing on the interaction between document and questions like other works do. Stochastic Answer Network (SAN) [14] utilizes a stochastic prediction dropout layer as the final layer. QANet [34] is constructed with both CNN and self-attention to combine local and global interactions. Embeddings from Language Models (ELMo) [21] extracts hidden states within bi-directional LSTMs and Bidirectional Encoder Representations (BERT) [5] uses deep-layered transformers to generate contextualized word embeddings.

2.3 Character Mining

The Character Mining dataset provides transcripts of the TV show *Friends* as well as annotation for several tasks [1]. Future research could combine this project with FriendsQA to generate more meaningful tasks and tools. The first two seasons are annotated [2] for character identification task, that is an entity linking task identifying personal mentions with character names. This annotation is extended [3] to the next two seasons and ambiguous mentions is further annotated. Building upon that, plural mentions to those four seasons [36] are also annotated for character identification tasks. Moreover, the first four seasons are annotated [35] for fine-grained emotion detection tasks. Finally, selected dialogues from all ten seasons are processed [15] for a cloze-style reading comprehension task.

[1] `github.com/emorynlp/character-mining`

2.4 Friends QA vs. Other Dialogue QA

Three datasets have been presented for QA on dialogue. However, they are different from FriendsQA and we will show that FriendsQA is much more difficult and worth extensive future study. Firstly, CoQA [23] aims to answer questions that are part of one-to-one conversations, whereas FriendsQA focuses on questions asked by third-parties listening to multiparty dialogues. Previous dataset based on transcripts of *Friends* is also presented [15]; however, their work aims to cloze-style QA restricted by `PERSON` entities, while we broadly focus on span-based QA with open-domain questions. Similarly, DREAM [27], although their passages are based on dialogue, tackles multiple-choice questions, which suit well for evaluating reading comprehension, but not necessarily for practical QA applications.

Chapter 3

The Corpus

In this chapter, the generation of FriendsQA (Section 3.1 will be discussed in details. The web interface used for crowdsourcing (Section 3.2, the different rounds of experiments (Section 3.6 and the two phases in each round (Section 3.3 and 3.5) will be elaborated and explained to demonstrate our dataset's integrity and diversity. To ensure the quality of the data, we additionally apply quality assurance procedures (Section 3.4), questions and answers pruning (Section 3.7), inter-annotators agreement (Section 3.8) and an extensive question-answer types analysis (Section 3.9) with a hope to convince that FriendsQA could serve as a valid and rich QA research resource in NLP community.

3.1 FriendsQA Dataset

To generate the FriendsQA dataset, we compile the first four seasons of the Character Mining dataset since they contain the additional annotations (Section 2.3) for the conveniences in future research. 1,222 scenes are excerpted and scenes with fewer than five utterances are discarded (83 of them). We concatenate all utterances in a dialogue to serve as an evidence passage. FriendQA can be viewed as answer span selection task similar to SQuAD [22], in which the model is expected to answer different types of questions proposed by human readers by finding contiguous spans in the dialogue containing the answer contents. Note that the questions generated are guaranteed to have a contiguous answer in the evidence passage.

The dialogue aspects of this dataset, however, make it more challenging than other datasets comprising passages in formal languages (Section 2.1). In dialogue, utterances are spoken by several people and context switching happens more frequently; thus, information needed to understand the dialogue context are often scattered across multiple utterances, which requires document-level inference. Also, the interchangeably use of co-references, the use of homophones to create humorous effects and the use of sarcasms to instead convey the opposite meanings are rather abundant. Three challenging

(a) Challenges with entity resolution. In this example (season 4, episode 12), $\{you_1$, $boys_2$, $us_3\}$ refer to the boys and $\{you_4$, $we_8\}$ refer to the girls. Many pronouns are used to refer different people, which makes it difficult to find the answer span for a question like "*who forced Rachel to raise the stakes*" by simply matching strings.

Rachel	Y'know what, **you**$_1$ are mean **boys**$_2$, who are just being mean!
Joey	Hey, don't get mad at **us**$_3$! No one forced **you**$_4$ to raise the stakes!
Rachel	That is not true. **She**$_5$ did! **She**$_6$ forced **me**$_7$!
Monica	Hey, **we**$_8$ would still be living here if **you**$_9$ hadnt gotten the question wrong!

(b) Challenges with metaphors. In this example (season 1, episode 4), Joey mishears '*omnipotent*' as "*I'm impotent*" so that he metaphorically refers it to as "*Little Joey's dead*", which makes it difficult to answer a question like "*why would Joey want to kill himself for being omnipotent*".

Monica	Hey, Joey, what would you do if you were **omnipotent**?
Joey	Probably kill myself!
Monica	Excuse me?
Joey	Hey, if **Little Joey's dead**, then I got no reason to live!

(c) Challenges with sarcasm. In this example (season 3, episode 1), Chandler is being sarcastic about him making pancakes, which makes it difficult to answer a question like "*did Chandler make pancakes*".

Chandler	Morning.
Joey	Morning, hey, you made pancakes?
Chandler	**Yeah, like there's any way I could ever do that.**

Table 3.1: Challenges with entity resolution, metaphors, and sarcasm in understanding dialogue contexts for QA.

aspects that could be commonly found in dialogue QA are illustrated in Table 3.1.

3.2 Crowdsourcing

All annotation tasks are conducted on the Amazon Mechanical Turk. TALEN, a web-based tool for named entity annotation [16], is extended for our QA annotation such that it displays a dialogue segmented into a sequence of utterances with utterance IDs and speaker names on the left panel. It allows the annotator to select a span of words with left-click. At the completion of the click, a pop-up window will show and reveal the available labels that could be used to tag the current span. In our case, the labels would be the question ID (For example, this span of texts is the answer to *what* question). On the right panel it will ask crowd workers to generate questions regarding this dialogue and then select spans in the dialogue that contain the correct information.

Prior to the annotation, crowd workers are required to pass a simple quiz regarding the dialogue context, to verify if they have a good understanding in this context and the required knowledge to use this web interface. The actual annotation task remains hidden until they pass this quiz. Upon the submission, a series of validations will take place and make sure the question and answer format are acceptable (Section 3.4).

3.3 Phase 1: Question-Answer Generation

Our annotation guidelines give clear instructions to crowd workers, to ensure that annotated questions and answers can be used for robust QA modeling. An example of the questions generated regarding a dialogue could be found in Table 3.2.

For each dialogue, the crowd workers are required to generate at least 4 out of six types of questions, {*who*, *what*, *when*, *where*, *why*, *how*}, regarding the dialogue contexts. Every question must be answerable; in other words, there needs to be at least one contiguous answer span in the dialogue. The crowd workers are allowed to select more than one answer span per question if appropriate. If multiple mentions of the same entity are to be considered, annotators are instructed to select ones that fit the best for the question. For instance, to answer `Q2` in Table 3.2, although multiple mentions of *Casey* are found in this dialogue, only the first three are selected as the answer because the other mentions are not relevant to this particular question (e.g., *Casey* in `U08`). This type of selective answer spans adds another level of difficulty to the task of FriendsQA.

We understand that sometimes the speakers can be the answers. Therefore, we put the speaker's full name for each utterance in the front of each utterance

(a) A dialogue excerpted from *Friends* (season 4, episode 7).

U01	[Scene: Central Perk, Joey is getting a phone number from a woman (Casey) as Chandler watches from the doorway.]
U02	Casey: Here you go.
U03	Joey: Great! All right, so I'll call you later.
U04	Casey: Great!
U05	Chandler: Hey-Hey-Hey! Who was that?
U06	Joey: That would be Casey. We're going out tonight.
U07	Chandler: Goin' out, huh? Wow! Wow! So things didn't work out with Kathy, huh? Bummer.
U08	Joey: No, things are fine with Kathy. I'm having a late dinner with her tonight, right after my early dinner with Casey.
U09	Chandler: What?
U10	Joey: Yeah-yeah. And the craziest thing is that I just ate a whole pizza by myself!
U11	Chandler: Wait! You're going out with Kathy!
U12	Joey: Yeah. Why are you getting so upset?
U13	Chandler: Well, I'm upset for you. I mean, dating an endless line of beautiful women must be very unfulfilling for you.

(b) Six types of questions:{*who*, *what*, *when*, *where*, *why*, *how*}.

Q1	What is Joey going to do with Casey tonight?	Q4	Where are Joey and Chandler?
Q2	Who is Joey getting a phone number from?	Q5	**Why** is Chandler upset?
Q3	When will Joey have dinner with Kathy?	Q6	How are things between Joey and Kathy?

Table 3.2: A sample dialogue from the FriendsQA dataset comprising six types of questions, where the answer spans are annotated on the dialogue contents. Each utterance has the utterance ID, the speaker name, and the text. The answer spans for Q[1-6] are indicated by solid underlines, wavy underlines, double underlines, dashed underlines, **bold font**, and dotted underlines, respectively.

so that the annotators will have the option to select these speaker names if they are deemed to be the correct answers. This is useful for *who* questions asking about certain speakers yet no explicit mentions of them are found in the dialogue (e.g. *Chandler* has no explicit mention in Table 3.2).

Moreover, when an entire utterance is considered to be the answer, which is the most often in *why* and *how* questions, annotators are asked to select the corresponding utterance ID instead of the whole utterance to reduce span-related errors (e.g., `U13` for `Q5` in Table 3.2), which is later post-processed to replace the utterance ID with the corresponding utterance.

3.4 Quality Assurance

Each MTurk annotation job gives up to 6 questions and their answer spans, which are validated by the following tests before the submission:

1. Are there at least 4 types of questions annotated?
2. Does each question have at least one answer span associated with it?
3. Does any question have too much string overlaps with the original text in the dialogue?

The first test ensures that there are sufficiently large and diverse enough questions generated for developing practical QA models. The second test checks if there are any inappropriate associations between questions and answer spans. Finally, the third test prevents from creating mundane questions by copying and pasting the original text from the dialogue. No annotation job is accepted unless it passes all of these assurance tests. To accomplish this, a *Validate* button is created so that all annotators needs to pass the 3 validations for the *Submit* button to become clickable.

3.5 Phase 2: Verification and Paraphrasing

All dialogues with the questions from the first phase (Section 3.3) are again put to the second phase. During the second phase, annotators are asked to first answer the questions from phase 1. Then, they are asked to revise questions that are either unanswerable or too ambiguous. Recall that we require all the questions to be answerable. Finally, they are asked to paraphrase the questions, resulting two sets of questions for every dialogue where one is generated from Phase 1 and the other one is a paraphrase of the first. The same quality assurance tests (Section 3.4) with an additional test of checking string overlaps between the questions from phases 1 and 2 are run to preserve the challenging level of this dataset.

3.6 Four Rounds of Annotation

The same F1-score metric used for the evaluation of span-based QA systems [22] is used to measure the inter-annotation agreement (ITA) between the answer spans annotated in Phases 1 and 2 (Sections 3.3 and 3.5). Four rounds of crowdsourcing experiments are conducted to stabilize the quality of our annotation. Two randomly selected episodes from Seasons 1-4 are used for each round of the 4 rounds, respectively. After each round, ITA is measured and a sample set of annotations is manually checked. The annotation guidelines are updated based on this assessment if necessary. The column A from the rows R1 $\sim$ R4 in Table 3.3 illustrates the progressive ITA improvements over these four rounds. The followings show summaries of actions performed after each round (R[1-4]: round 1-4):

R1 We observe that the questions are often too ambiguous for humans to answer; thus, we update the guidelines and request annotators to make the questions as explicit as possible.

R2 We observe the 6.27% improvement on ITA from the first round; thus, we add more examples of questions and answer spans to the guidelines without updating other contents.

R3 We observe another 2.48% improvement on ITA from the second round; no update is made to the guidelines.

R4 We observe a marginal ITA improvement of 0.67% from the third round, which implies that our annotation guidelines are stabilized. Thus, all of the rest episodes are pushed for annotation.

	S	**Q**	$\mathbf{Q_p}$	$\mathbf{Q_r}$	**A**	$\mathbf{A_p}$	**F1**	$\mathbf{F1_p}$	**EM**	$\mathbf{EM_p}$
R1	24	122	98	62	264	216	66.59	83.42	48.15	61.17
R2	26	242	185	57	484	368	72.86	83.99	50.00	57.69
R3	30	264	213	66	528	422	75.34	83.12	48.92	53.97
R4	37	370	296	75	740	593	76.01	88.17	52.25	60.78
S1	288	2,908	2,560	627	5,824	5,123	69.93	79.78	42.78	49.01
S2	259	2,682	2,314	587	5,372	4,633	69.21	80.86	44.01	51.73
S3	291	2,908	2,546	610	5,826	5,099	72.12	81.92	47.22	53.88
S4	267	2,768	2,398	594	5,553	4,808	72.26	83.27	49.52	57.41
Total	**1,222**	12,264	**10,610**	2,678	2,4591	**21,262**	71.17	**81.82**	46.35	**53.55**

Table 3.3: Statistics of the FriendsQA dataset. The R[1-4] rows show the statistics for the rounds 1-4, and the S[1-4] rows show the statistics for Seasons 1-4, respectively. S: # of dialogues, Q: # of questions, Q_p: Q after pruning, Q_r: # of revised questions during phase 2, A: # of answer spans, A_p: A after pruning, F1: F1-score to measure ITA, F1_p: F1 after pruning, EM: exact matching score to measure ITA, EM_p: EM after pruning.

3.7 Question/Answer Pruning

Once all annotation is collected, each question from phase 1 is represented by the bag-of-words model using TF-IDF scores and compared against its revised counterpart from phase 2 if available. About 21.8% of the questions from phase 1 are revised during phase 2, implying that the option to revise the question is beneficial to the overall quality of our dataset.

If the cosine similarity between the two questions is below 0.8, they are not considered similar so that the question and its answer spans from phase 1 are discarded because that question requires a major revision to be specific and answerable, which means that they are not as valuable as the questions from Phase 2.

Even when the questions are considered similar, if the F1 score between their answer spans is below 20, they are still discarded because annotators do not seem to agree on the answer. As a result, 13.5% of the questions and answer spans from phase 1 are pruned out from our final dataset. The pruning dramatically increases the ITA with a small fraction of questions and answers discarded. All pruning stats could again be found in Table 3.3.

3.8 Inter-annotator Agreement

Table 3.3 show the overall statistics of the FriendsQA dataset. There is a total of 1,222 dialogues, 10,610 questions, and 21,262 answer spans in this dataset after pruning (Section 3.7). There are at least 2 answers to each question since there are 2 phases during annotation, each of which will acquire an answer to the same question. Note that annotators were not asked to paraphrase questions during the second phase of the first round (R1 in Table 3.3), so the number of questions in R1 is about twice less than ones from the other rounds. The final inter-annotator agreement scores are 81.82% and 53.55% for the F1 and exact matching scores respectively, indicating high-quality annotation in our dataset.

3.9 Question Types vs. Answer Categories

Now that we have reached the conclusion that FriensQA contains quality questions and answers on which the annotators agree, extensive analysis is further applied to investigate its diversity in terms of different answer categorizations. 250 questions are randomly sampled out to perform such analysis. Table 3.4 shows the statistics between the question types and answer categories, where answers to each question type are categorized into 2 types. Questions show balanced distributions across different types, indicating good diversity of the dataset. Description to each answer type can be found below.

Type	**Count**	**Answer Categories (%)**			
What	2,058	Factual:	100.00	Abstract:	0.00
Where	1,896	Factual:	77.78	Abstract:	22.22
Who	1,847	Speaker:	30.56	Content:	69.44
Why	1,688	Explicit:	73.53	Implicit:	26.47
How	1,628	Explicit:	77.42	Implicit:	22.58
When	1,493	Absolute:	62.07	Relative:	37.93

Table 3.4: Statistics of the question types as well as the answer categories.

What No distinct categorization is found for answers to *what* questions, which are entirely factual. This is because annotators are mostly driven by factoid contents for the generation of *what* questions.

Where Answers to *where* questions can be categorized into factual and abstract, meaning that they are either concrete facts (e.g., named entities) or abstract concepts (e.g., *the wild*, *out there*), where the majority is driven by factoid contents (77.78%).

Who Answers to *who* questions can be annotated on either speaker names or utterance contents. Recall that the annotators might select the speaker names as answers if they are not explicitly mentioned in the dialogue. The majority of *who* questions (69.44%) finds their answers in the utterance contents.

Why and How Answers to *why* and *how* questions are categorized into explicit and implicit such that they are either directly answering the questions (e.g., why doesn't Joey want to throw the chair out? → *Joey: I built this thing with my own hand*), or indirectly implying the answers (e.g., How are Joey and Chandler going to get to Monica's place? → *Chandler: we're not gonna have to walk there, right?*). Explicit answers are more common for both *why* (73.53%) and *how* (77.42%) questions.

When Answers to *when* questions can be categorized into absolute and relative such that they can be either exact timing (e.g., clock time, specific date, holiday) or timing of action relative to another event (e.g., I called her *while I was watching TV*). About two third of the answers are considered explicit for *when* questions.

Through such analysis, it is safe to conclude that FriendsQA is valid as a Question Answering research resource and diverse in nature given its miscellaneous types of questions and answers. The proposed approaches to officially make the best out of FriendsQA will be discussed in the next chapter.

Chapter 4

Approach

To prove the potential of FriendsQA, three of the state-of-the-art QA systems, R-Net, which is based on recurrent neural networks (RNN) (Section 4.1), QANet, which is based on convolutional neural networks (CNN) with self-attention (Section 4.2), and BERT, which is based on deep feed-forward neural networks with transformers (Section 4.3), are used to validate our dataset as a practical resource for building advanced deep learning models. All models will output two positions which will be combined to form answer spans. These systems are chosen because they give a good survey among different types of neural networks in combination with attention mechanisms that are dominant in the research of contemporary question answering. The results shown by these systems should represent other standard neural models' performances on FriendsQA.

4.1 R-Net

R-Net held the 1st place on the SQuAD leaderboard at the time of its publication [30]. It builds representations for questions and evidence passages using RNN and presents a self-matching mechanism to aggregate key information from the evidence passages in order to compensate the limitedly memorized information from RNN. The same configuration described in the original paper is used to train models for our experiments.

4.2 QANet

QANet is another state-of-the-art open-domain QA system utilizing CNN and self-attention [34]. Dramatic is the speed-up gained by QANet, which enables it to train with more data with data augmentation. Their original configuration cannot fit in a 12GB GPU machine using our dataset; thus, the configuration is compromised for our experiments as follows:

- The number of filters: 96 instead of 128,
- The number of attention heads: 1 instead of 8.

Given this configuration, its performance may not be optimal but at least can be directly compared to other models trained on the FriendsQA dataset.

4.3 BERT

The Bidirectional Encoder Representations from Transformers (BERT) pushed all current state-of-the-art scores to another level [5]. Trained with the masked language model on next sentence prediction tasks, BERT shows extremely promising results on several tasks in NLP. The pre-trained decapitalized BERT model with 12-layers is fine-tuned on our dataset. The larger BERT model with 24-layers again cannot be fit in a 12GB GPU machine; thus, it is not used for our experiments.

Chapter 5

Experiments

For our experiments, all dialogues from Table 3.3 are randomly shuffled and redistributed as the training (80%), development (10%), and test (10%) as shown in Table 5.1.

Set	**Dialogues**	**Questions**	**Answers**
Training	977	8,535	17,074
Development	122	1,010	2,057
Test	123	1,065	2,131

Table 5.1: Data split for our experiments.

5.1 Model Development

Each instance consists of an evidence dialogue, a question and an answer span. To create one evidence passage from the dialogue, we simply concatenate all the utterances, each containing the utterance ID and the speaker name in the front. Recall that an utterance ID could be annotated to represent the whole utterance (Section 3.3). Therefore, they are pre-processed and replaced by the corresponding utterance from the dialogue. Since each question can have multiple answers, the following strategies are experimented to acquire one gold answer span for each training instance:

Shortest The shortest answer span is chosen and all the other spans are discarded from training.

Longest The longest answer span is chosen and all the other spans are discarded from training.

Multiple The question is paired with every answer to create multiple instances. For example, a question q with two answer spans, a_1 and a_2, generate two instances, (q, a_1) and (q, a_2), and trained along with other instances.

5.2 Evaluation Metrics

Given the uniqueness of our dataset, three evaluation metrics are adopted for our experiments to demonstrate the systems' performance on FriendsQA.

First, following SQuAD[22], Span Match (SM) is adapted to evaluate answer span selection, where each a_i^p is treated as a bag-of-tokens (ϕ) and compared to the bag-of-tokens of a_i^g; the macro-average F1 score across all questions is measured for the final evaluation (P: precision, R: recall):

$$\mathbf{SM} = \frac{1}{n}\sum_{i=1}^{n}\frac{2 \cdot P(\phi(a_i^p), \phi(a_i^g))R(\phi(a_i^p), \phi(a_i^g))}{P(\phi(a_i^p), \phi(a_i^g)) + R(\phi(a_i^p), \phi(a_i^g))}$$

Additionally, Exact Match (EM) is also adopted to evaluate answer span selection that checks the exact span match between the gold and predicted answers, which results in a score either 1 or 0.

Given the nature of FriendsQA in which each utterance is treated as a single unit in conversations, Utterance Match (UM) could serve as an effective measure to evaluate the accuracy since the model is considered to be powerful if it is always looking for answers in the correct utterance. High Utterance Match could indicate high precision of the model's global understanding toward the dialogue. Given a prediction a_i^p, UM mainly checks if it resides within the same utterance u_i^g as the gold answer span a_i^g, and is measured as follows: (n: # of questions):

$$\mathbf{UM} = \frac{1}{n}\sum_{i=1}^{n}(1 \text{ if } a_i^p \in u_i^g; \text{otherwise}, 0)$$

5.3 Results

Table 5.2 shows results from 9 models trained by the three state-of-the-art systems (Chapter 4) using the three answer selection strategies (Section 5.1).

Model	**Shortest-Answer Strategy**			**Longest-Answer Strategy**			**Multiple-Answer Strategy**		
	UM	**SM**	**EM**	**UM**	**SM**	**EM**	**UM**	**SM**	**EM**
R-Net	45.41 (±1.16)	35.69 (±1.28)	**25.55** (±1.60)	**49.50** (±0.54)	**37.26** (±0.72)	23.77 (±0.42)	43.77 (±0.56)	33.97 (±0.75)	23.02 (±1.30)
QANet	42.12 (±3.21)	34.04 (±0.03)	22.89 (±0.42)	46.21 (±4.51)	34.55 (±1.87)	21.15 (±1.21)	**47.10** (±1.30)	**35.38** (±1.33)	**23.16** (±1.15)
BERT	72.61 (±0.20)	63.64 (±0.42)	48.33 (±1.41)	72.16 (±1.93)	60.36 (±1.53)	43.23 (±1.83)	**74.18** (±0.21)	**64.15** (±0.29)	**48.96** (±0.42)
BERT$_R$	66.38 (±0.86)	49.28 (±0.28)	28.41 (±1.25)	65.60 (±5.63)	**58.00** (±0.99)	**40.52** (±0.63)	**68.65** (±1.63)	54.87 (±0.43)	38.87 (±1.46)

Table 5.2: Results from the three state-of-the-art QA systems. All models are experimented three times and their average scores with standard deviations are reported. UM: Utterance Match, SM: Span Match, EM: Exact Match. BERT$_R$: BERT with PERSON entities replaced

All experiments are run three times and their average scores with standard deviations are reported. BERT and QANet perform better with the multiple-answer strategy which gives more training instances per question and potentially takes all answer spans into account, whereas R-Net performs better with the other strategies when only shortest or longest answers are considered. The relatively worse performance of the multiple-answer strategy for R-Net could be due to its self-matching mechanism that gets confused when multiple answers are provided for training the same question. BERT models significantly outperform ones from the other two systems in all evaluations. However, since our hyper-parameters are tuned around grids provided by the original papers, it is possible that these results are still suboptimal, which points out another important property of BERT that it is not as sensitive to different QA datasets.

Type	**Dist.**	**UM**	**SM**	**EM**
What	19.70%	77.43	69.39	55.04
Where	18.28%	84.35	78.86	65.93
Who	17.17%	74.12	64.34	55.29
Why	15.76%	60.47	50.03	27.14
How	14.65%	65.52	52.04	32.64
When	14.44%	80.65	65.81	51.98

Table 5.3: Results with respect to question types using BERT and the multiple-answer strategy.

Results Based on Question Type Table 5.3 shows results from BERT's multiple answer models by question types. Answers to *where* and *when* questions are mostly factoid, which show the highest performance. On the other hand, answers to *why* and *how* usually span out to longer sequences and requires cross-utterance reasoning, leading to worse performance. Answers to *who* and *what* questions give a good mixture of proper and common nouns and show moderate performance.

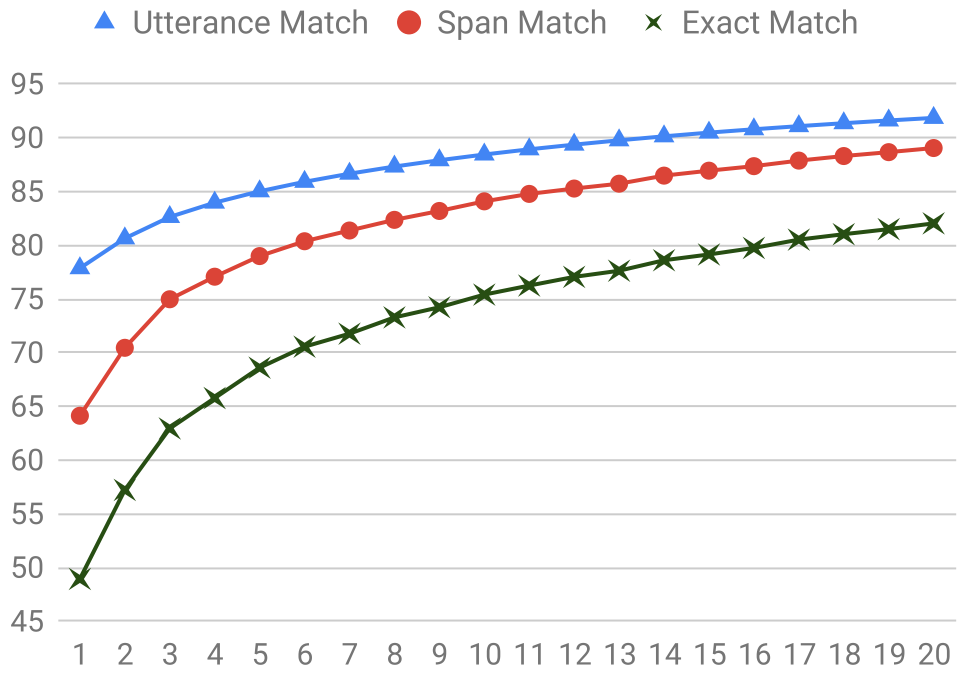

Figure 5.1: Increasing score with top-k answer candidates. From top to bottom: Utterance Match, Span Match and Exact Match.

NBest Results Figure 5.1 shows improvement of BERT's multiple-answer models by accepting the top-k answer predictions; the scores are measured by picking the best matching answer within thes top-k predictions. UM surpasses 90% and SM approaches to 90% when $k = 14$ and 20, respectively. More importantly, the gap between UM and SM gets smaller as k increases, which implies that FriendsQA is not only learnable by deep learning but also can be enhanced by re-ranking the answer predictions.

. . . . . .

u004 ent0 And if ~~I~~ ⇒ ent0 want to call for a reference on ~~your~~ ⇒ ent1 last job ?
u005 ent1 Oh , that 's there on the bottom , see the manager , ~~Chandler Bing~~ ⇒ ent2 .
u006 ent0 Alright , let s see if ~~you~~ ⇒ ent1 're as good in person as ~~you~~ ⇒ ent1 are on paper . Make ~~me~~ ⇒ ent0 **a salad** .
u007 ent1 A salad ? Really ~~I~~ ⇒ ent1, ~~I~~ ⇒ **ent1 could do something a little more complicated if ~~you~~ ⇒ ent0 like** .

. . . . . .

1. What does ~~the interviewer~~ ⇒ ent0 want ~~Monica~~ ⇒ ent1 to make ?
a salad
2. Who is the person ~~Monica~~ ⇒ ent1 uses as a reference ?
ent2
3. Why is ~~Monica~~ ⇒ ent1 surprised by what ~~the interviewer~~ ⇒ ent0 wants ~~her~~ ⇒ ent1 to make ?
ent1 could do something a little more complicated if ent0 like

Entity Map:
Interviewer → ent0
Monica Geller → ent1
Chandler Bing → ent2

Table 5.4: An example of context-question-answer pair with PERSON entities replaced in FriendsQA

Co-references Replacement Recall that the first 4 seasons of Friends transcripts available in the Character Mining project (Section 2.3) contain co-reference resolution annotations. Therefore, in a hope to reduce the confusion caused by the entities and to experiment a potential direction for future research, for each scene, a map that encodes character's full names to entity IDs is kept so that every PERSON entity could be replaced by that entity

ID, similar to the dataset in [15]. Note that mentions in both dialogue and questions are encoded to keep the dataset consistent and plural mentions are handled in a naive way that is direct substitution (e.g. ~~we are back~~ ⇒ ent0 ent1 ent2 are back). The same example used in Chapter 1 could be found in Table 5.4. As for the results, the span-based and exact-match performance worsen by 6% and 8%, respectively as shown in Table 5.2. This indicates that direct substitution is not the effective way to handle PERSON entities and requires deeper research.

SoU	Acc
1	57.23
2	57.62
3	55.25
Avg	56.70

Table 5.5: Results of Start of Utterance

Start of Utterance Prediction Recall that Utterance Match is used to evaluate the performance of the selected models. UM checks if the prediction resides within the same utterance as the gold answer. However, this is not truly predicting utterances. To challenge the models' ability to predict the correct utterance, another experiment is designed: the answer to each question is replaced with the start position of the utterance instead of the start and end positions of answer spans. Therefore, the models will no longer need two output layers: only one is needed to indicate the start of the utterance. The model will be trained to select the utterance that contains the answers by pointing at the start of the utterance, and the evaluation will simply give the accuracy of the selection. The results are reported in Table 5.5. This experiment does not show competitive results, but it yields another interesting finding: neural network could distinguish a bunch of utterances concatenated together without intentional delimiters, since each prediction is indeed the start of an utterance. Based on the training set, the model could figure out what type of answers it should be predicting, indicating the power of neural networks.

5.4 Error Analysis

An extensive error analysis is manually performed on 100 randomly sampled, completely mismatched predictions (F1 = 0) to provide insights for future research. Figure 5.2 shows six types of errors that become evident through this analysis and will be explained as following.

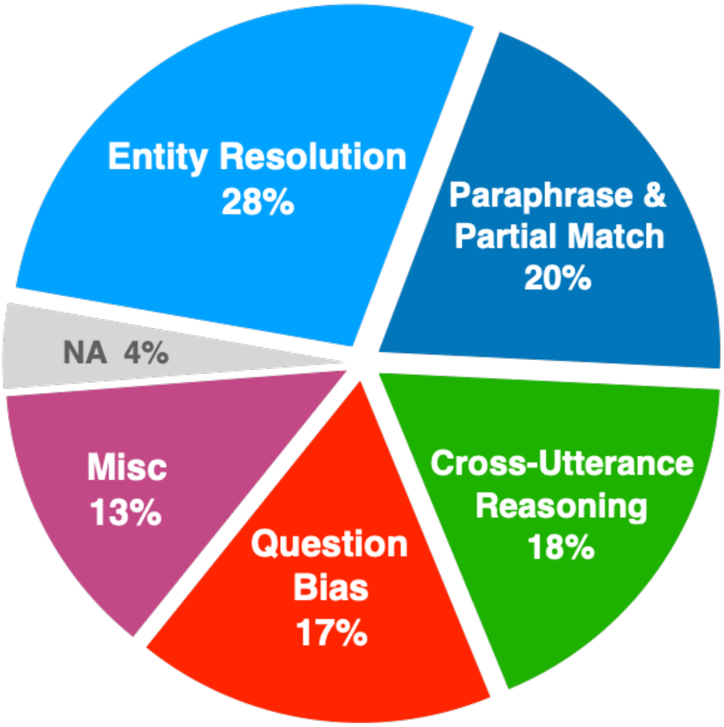

Figure 5.2: The distribution of six error types analyzed in 100 sampled predictions. NA: Noise in annotation.

Entity Resolution This type is the most frequent and often occurs when many of the same entities are mentioned in multiple utterances. The recurring use of coreference and anaphora can be confusing. This error also occurs when the QA system is asked about a specific person, but predicts wrong people. For example, the question asks for Chandler's opinion about marriage, but the model matches comments from Joey instead due to the lack of referent resolution made in those comments. Such errors take approximately up to 28%.

Paraphrase and Partial Match This type of error, which is the second most frequent, might not be considered as errors to human readers. Such type of error happens if a fact, a story or an item is referred to in numerous ways (paraphrasing, abstraction, nicknames, etc.) somewhere else in the conversation. Moreover, answers might also be partially correct, especially for *why* and *how* questions, which could be acceptable in practice and motivates us to evaluate using **Utterance Match**. Such errors take approximately up to 20%.

Cross-Utterance Reasoning This type reveals an universal challenge in understanding human-to-human conversation. To correctly predict an answer span in the dialogue, the system should be equipped with the ability to reason across multiple utterances back and forth, especially if a story or an event unfolds gradually, scatters in different places, and is told by different speakers. Such errors take approximately up to 18%.

Question Bias This type occurs when the answer predictions overly rely on the question types. For *why* questions, the model tends to blindly selects spans following certain keywords such as *because* even though they are placed in wrong utterances since the model is learned to be biased to the term *because*, neglecting other important factors that might otherwise lead to the correct answers. This applies to other types of questions as well, when there are multiple time phrases for *when* questions, locations for *where* questions and names for *who* questions. Such errors take approximately up to 17%.

Noise in Annotation (NA) Our dataset, although gives high inter-annotator agreement (Section 3.8), still includes noise caused by wrong spans, ambiguous and unanswerable questions, or typos. Noisy annotations take a small fraction of 4%.

Miscellaneous Errors in this category have no apparent cause to understand why the model predicts these answers, which often seem irrelevant to the questions so that they need more investigation. Such predictions take about 13%.

Given this analysis, we hope many challenges become clear and easier to be overcome in future studies. For instance, coreferent mentions, especially plural mentions, should be more intelligently processed [36]. Moreover, the speaker information, which are currently treated as the first tokens in utterances, can be better encoded to give more insights.

Chapter 6

Conclusion

This thesis presents an open-domain question answering dataset called **FriendsQA**, compiled from the transcripts of the beloved TV show *Friends* to promote the understanding of human-to-human conversations and the answering to open-domain queries under colloquial context. An extensive and comprehensive analysis on the types of questions and answers is performed to show FriendsQA's validity, difficulty and diversity. Multiple strategies to select gold answer spans are experimented and reported, providing more than one way of generating the training set. Three state-of-the-art models are run and compared, and show the full potential of FriendsQA as a rich QA research resource by presenting meaningful results from three different evaluation metrics. Tentative co-reference resolution is naively incorporated using direct substitution suggesting that a more sophisticated handling of character mentions [36] are needed. Finally, erroneous answer predictions are sampled out for a further analysis to offer insightful retrospective and make suggestions to future deeper study.

For future work, the question-type (Table 5.3) and error analysis (Section 5.4) can serve as guidelines to further enhance the QA model performance. *Why* and *how* questions should be studied more attentively to improve the overall performance. Questions that require global understanding and inference should be treated with special care, probably with a task-specific model. To deal with the fact that the models are likely to get confused when predicting a specific character, the speaker information, which are currently treated with no difference from other words, can be somehow encoded into the utterance to better distinguish between characters. Top-k answer analysis also brings up another challenging but tangible task to re-rank the answer predictions. More tasks such as answer existence prediction and an utterance-based model to select among utterance candidates can easily be generated.

Appendix 7

Complete Results

Model	Iteration	Shortest-Answer Strategy			Longest-Answer Strategy			Multiple-Answer Strategy		
		UM	SM	EM	UM	SM	EM	UM	SM	EM
R-Net	1	46.71	36.97	26.95	49.46	37.97	24.25	43.19	33.13	21.66
	2	44.46	34.41	23.80	48.98	36.54	23.53	44.31	34.58	24.26
	3	45.06	35.69	25.90	50.06	37.26	23.53	43.81	34.21	23.14
QANet	1	40.97	34.02	22.46	45.64	33.14	20.22	48.14	36.90	23.95
	2	39.64	34.03	23.30	42.01	33.84	20.70	45.64	34.83	23.70
	3	45.75	34.07	22.91	50.97	36.67	22.52	47.52	34.43	21.84
BERT	1	72.77	63.78	48.45	71.44	59.06	41.39	73.94	63.90	49.29
	2	72.68	63.96	49.67	74.34	62.05	45.04	74.34	64.07	48.48
	3	72.39	63.16	46.85	70.69	59.98	43.26	74.24	64.46	49.09
BERT_R	1	66.48	49.61	29.45	63.83	57.51	38.94	66.67	54.16	38.87
	2	65.47	49.15	28.75	67.02	58.21	40.80	75.00	56.01	39.50
	3	67.19	49.09	27.03	65.96	58.29	41.82	64.29	54.45	38.25

Table 7.1: Complete results from the three state-of-the-art QA systems with 3 iterations. UM: Utterance Match, SM: Span Match, EM: Exact Match. BERT_R: BERT with `PERSON` entities replaced

Bibliography

[1] Ondrej Bajgar, Rudolf Kadlec, and Jan Kleindienst. Embracing data abundance: BookTest Dataset for Reading Comprehension. *arXiv*, 1610.00956, 2016. URL `http://arxiv.org/abs/1610.00956`.

[2] Henry Yu-Hsin Chen and Jinho D. Choi. Character Identification on Multiparty Conversation: Identifying Mentions of Characters in TV Shows. In *Proceedings of the 17th Annual Meeting of the Special Interest Group on Discourse and Dialogue*, SIGDIAL'16, pages 90–100, 2016. URL `http://www.aclweb.org/anthology/W16-3612`.

[3] Henry Yu-Hsin Chen, Ethan Zhou, and Jinho D. Choi. Robust Coreference Resolution and Entity Linking on Dialogues: Character Identification on TV Show Transcripts. In *Proceedings of the 21st Conference on Computational Natural Language Learning*, CoNLL'17, 2017. URL `http://www.aclweb.org/anthology/K17-1023`.

[4] Yiming Cui, Zhipeng Chen, Si Wei, Shijin Wang, Ting Liu, and Guoping Hu. Attention-over-Attention Neural Networks for Reading Comprehension. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics*, ACL'17, pages 593–602, 2017. URL `http://aclweb.org/anthology/P17-1055`.

[5] Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding. *arXiv*, 1810.04805, 2018. URL `http://arxiv.org/abs/1810.04805`.

[6] Pedro Gonçalves. 10 graphs that show why your business should be available through messaging apps, 2 2017. URL `https://medium.com/hijiffy/10-graphs-that-show-the-immense-power-of-messaging-apps-4a41385b24d6`.

[7] Karl Moritz Hermann, Tomás Kociský, Edward Grefenstette, Lasse Espeholt, Will Kay, Mustafa Suleyman, and Phil Blunsom. Teaching Machines to Read and Comprehend. In *Annual Conference on Neural Information Processing Systems*, NIPS'15,

pages 1693–1701, 2015. URL https://papers.nips.cc/paper/5945-teaching-machines-to-read-and-comprehend.

[8] Felix Hill, Antoine Bordes, Sumit Chopra, and Jason Weston. The Goldilocks Principle: Reading Children's Books with Explicit Memory Representations. In *Proceedings of the 6th International Conference on Learning Representations*, ICLR'16, 2016. URL https://arxiv.org/abs/1511.02301.

[9] Minghao Hu, Yuxing Peng, Zhen Huang, Xipeng Qiu, Furu Wei, and Ming Zhou. Reinforced Mnemonic Reader for Machine Reading Comprehension. In *Proceedings of the 27th International Joint Conference on Artificial Intelligence*, IJCAI'18, pages 4099–4106, 2017. URL https://www.ijcai.org/proceedings/2018/0570.pdf.

[10] Hsin-Yuan Huang, Chenguang Zhu, Yelong Shen, and Weizhu Chen. FusionNet: Fusing via Fully-aware Attention with Application to Machine Comprehension. In *Proceedings of the International Conference on Learning Representations*, page ICLR'18, 2018. URL https://openreview.net/forum?id=BJIgi_eCZ.

[11] Mandar Joshi, Eunsol Choi, Daniel Weld, and Luke Zettlemoyer. Trivi-

aQA: A Large Scale Distantly Supervised Challenge Dataset for Reading Comprehension. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics*, ACL'17, pages 1601–1611, 2017. URL `http://aclweb.org/anthology/P17-1147`.

[12] Aniruddha Kembhavi, Minjoon Seo, Dustin Schwenk, Jonghyun Choi, Ali Farhadi, and Hannaneh Hajishirzi. Are You Smarter Than a Sixth Grader? Textbook Question Answering for Multimodal Machine Comprehension. In *The IEEE Conference on Computer Vision and Pattern Recognition*, CVPR'17, 2017. URL `https://ieeexplore.ieee.org/document/8100054`.

[13] Guokun Lai, Qizhe Xie, Hanxiao Liu, Yiming Yang, and Eduard Hovy. RACE: Large-scale ReAding Comprehension Dataset From Examinations. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing*, EMNLP'17, pages 785–794, 2017. URL `http://aclweb.org/anthology/D17-1082`.

[14] Xiaodong Liu, Yelong Shen, Kevin Duh, and Jianfeng Gao. Stochastic Answer Networks for Machine Reading Comprehension. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics*,

ACL'18, pages 1694–1704, 2018. URL http://aclweb.org/anthology/P18-1157.

[15] Kaixin Ma, Tomasz Jurczyk, and Jinho D. Choi. Challenging Reading Comprehension on Daily Conversation: Passage Completion on Multiparty Dialog. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long Papers)*, pages 2039–2048, New Orleans, Louisiana, June 2018. Association for Computational Linguistics. URL http://www.aclweb.org/anthology/N18-1185.

[16] Stephen Mayhew and Dan Roth. TALEN: Tool for Annotation of Low-resource ENtities. In *Proceedings of the ACL System Demonstrations*, ACL:DEMO'18, pages 80–86, 2018. URL http://aclweb.org/anthology/P18-4014.

[17] Frank Newport. The New Era of Communication Among Americans, 11 2014. URL https://news.gallup.com/poll/179288/new-era-communication-americans.aspx.

[18] Tri Nguyen, Mir Rosenberg, Xia Song, Jianfeng Gao, Saurabh Tiwary, Rangan Majumder, and Li Deng. MS MARCO: A Human Generated

MAchine Reading COmprehension Dataset. In *Proceedings of the NIPS Workshop on Cognitive Computation: Integrating Neural and Symbolic Approaches*, 2016. URL `http://ceur-ws.org/Vol-1773/CoCoNIPS_2016_paper9.pdf`.

[19] Takeshi Onishi, Hai Wang, Mohit Bansal, Kevin Gimpel, and David McAllester. Who did What: A Large-Scale Person-Centered Cloze Dataset. In *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing*, EMNLP'16, pages 2230–2235, 2016. URL `https://aclweb.org/anthology/D16-1241`.

[20] Boyuan Pan, Hao Li, Zhou Zhao, Bin Cao, Deng Cai, and Xiaofei He. Memen: Multi-layer embedding with memory networks for machine comprehension. *CoRR*, abs/1707.09098, 2017.

[21] Matthew Peters, Mark Neumann, Mohit Iyyer, Matt Gardner, Christopher Clark, Kenton Lee, and Luke Zettlemoyer. Deep Contextualized Word Representations. In *Proceedings of the Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, NAACL'18, pages 2227–2237, 2018. URL `http://www.aclweb.org/anthology/N18-1202`.

[22] Pranav Rajpurkar, Jian Zhang, Konstantin Lopyrev, and Percy Liang. SQuAD: 100,000+ Questions for Machine Comprehension of Text. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing*, EMNLP'16, pages 2383–2392, 2016. URL `https://aclweb.org/anthology/D16-1264`.

[23] Siva Reddy, Danqi Chen, and Christopher D. Manning. CoQA: A Conversational Question Answering Challenge. *arXiv*, 1808.07042, 2018. URL `http://arxiv.org/abs/1808.07042`.

[24] Matthew Richardson, Christopher J.C. Burges, and Erin Renshaw. MCTest: A Challenge Dataset for the Open-Domain Machine Comprehension of Text. In *Proceedings of the 2013 Conference on Empirical Methods in Natural Language Processing*, EMNLP'13, pages 193–203, 2013. URL `http://aclweb.org/anthology/D13-1020`.

[25] Shimi Salant and Jonathan Berant. Contextualized Word Representations for Reading Comprehension. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, ACL'18, pages 554–559, 2018. URL `http://aclweb.org/anthology/N18-2088`.

[26] Yelong Shen, Po-Sen Huang, Jianfeng Gao, and Weizhu Chen. ReasoNet: Learning to Stop Reading in Machine Comprehension. In *Proceedings of the 23rd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, KDD'17, pages 1047–1055, 2017. URL `https://dl.acm.org/citation.cfm?id=3098177`.

[27] Kai Sun, Dian Yu, Jianshu Chen, Dong Yu, Yejin Choi, and Claire Cardie. DREAM: A Challenge Dataset and Models for Dialogue-Based Reading Comprehension. *Transactions of the Association for Computational Linguistics*, 7, 2019. URL `https://arxiv.org/abs/1902.00164`.

[28] Adam Trischler, Tong Wang, Xingdi Yuan, Justin Harris, Alessandro Sordoni, Philip Bachman, and Kaheer Suleman. NewsQA: A Machine Comprehension Dataset. In *Proceedings of the 2nd Workshop on Representation Learning for NLP*, pages 191–200, 2017. URL `http://www.aclweb.org/anthology/W17-2623`.

[29] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Ł ukasz Kaiser, and Illia Polosukhin. Attention Is All You Need. In *Proceedings of the 31st Conference on Neural Informa-*

tion Processing Systems, NIPS'17, pages 5998–6008, 2017. URL `https://papers.nips.cc/paper/7181-attention-is-all-you-need.pdf`.

[30] Wenhui Wang, Nan Yang, Furu Wei, Baobao Chang, and Ming Zhou. Gated Self-Matching Networks for Reading Comprehension and Question Answering. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics*, ACL'17, pages 189–198, 2017. URL `http://aclweb.org/anthology/P17-1018`.

[31] Johannes Welbl, Nelson F. Liu, and Matt Gardner. Crowdsourcing Multiple Choice Science Questions. In *Proceedings of the 3rd Workshop on Noisy User-generated Text*, pages 94–106, 2017. URL `http://aclweb.org/anthology/W17-4413`.

[32] Jason Weston, Antoine Bordes, Sumit Chopra, Alexander M. Rush, Bart van Merriënboer, Armand Joulin, and Tomas Mikolov. Towards AI-Complete Question Answering: A Set of Prerequisite Toy Tasks. In *Proceedings of the 5th International Conference on Learning Representations*, ICLR'16, 2016. URL `https://arxiv.org/pdf/1502.05698`.

[33] Yi Yang, Wen-tau Yih, and Christopher Meek. WIKIQA: A Challenge Dataset for Open-Domain Question Answering. In *Proceedings*

of the Conference on Empirical Methods in Natural Language Processing, EMNLP'15, pages 2013–2018, 2015. URL `https://aclweb.org/anthology/D15-1237`.

[34] Adams Wei Yu, David Dohan, Minh-Thang Luong, Rui Zhao, Kai Chen, Mohammad Norouzi, and Quoc V. Le. QANet: Combining Local Convolution with Global Self-Attention for Reading Comprehension. In *Proceedings of the 6th International Conference on Learning Representations*, ICLR'18, 2018. URL `https://openreview.net/pdf?id=B14TlG-RW`.

[35] Sayyed Zahiri and Jinho D. Choi. Emotion Detection on TV Show Transcripts with Sequence-based Convolutional Neural Networks. In *Proceedings of the AAAI Workshop on Affective Content Analysis*, AFFCON'18, New Orleans, LA, 2018. URL `https://sites.google.com/view/affcon18`.

[36] Ethan Zhou and Jinho D. Choi. They Exist! Introducing Plural Mentions to Coreference Resolution and Entity Linking. In *Proceedings of the 27th International Conference on Computational Linguistics*, pages 24–34, 2018. URL `http://www.aclweb.org/anthology/C18-1003`.