Transformers to Learn Hierarchical Contexts in Multiparty Dialogue for Span-based Question Answering

Changmao Li
Department of Computer Science
Emory University
Atlanta, GA, USA
`changmao.li@emory.edu`

Jinho D. Choi
Department of Computer Science
Emory University
Atlanta, GA, USA
`jinho.choi@emory.edu`

Abstract

We introduce a novel approach to transformers that learns hierarchical representations in multiparty dialogue. First, three language modeling tasks are used to pre-train the transformers, token- and utterance-level language modeling and utterance order prediction, that learn both token and utterance embeddings for better understanding in dialogue contexts. Then, multi-task learning between the utterance prediction and the token span prediction is applied to fine-tune for span-based question answering (QA). Our approach is evaluated on the FRIENDSQA dataset and shows improvements of 3.8% and 1.4% over the two state-of-the-art transformer models, BERT and RoBERTa, respectively.

1 Introduction

Transformer-based contextualized embedding approaches such as BERT (Devlin et al., 2019), XLM (CONNEAU and Lample, 2019), XLNet (Yang et al., 2019), RoBERTa (Liu et al., 2019), and AlBERT (Lan et al., 2019) have re-established the state-of-the-art for practically all question answering (QA) tasks on not only general domain datasets such as SQUAD (Rajpurkar et al., 2016, 2018), MS MARCO (Nguyen et al., 2016), TRIVIAQA (Joshi et al., 2017), NEWSQA (Trischler et al., 2017), or NARRATIVEQA (Koisk et al., 2018), but also multi-turn question datasets such as SQA (Iyyer et al., 2017), QUAC (Choi et al., 2018), COQA (Reddy et al., 2019), or CQA (Talmor and Berant, 2018). However, for span-based QA where the evidence documents are in the form of multiparty dialogue, the performance is still poor even with the latest transformer models (Sun et al., 2019; Yang and Choi, 2019) due to the challenges in representing utterances composed by heterogeneous speakers.

Several limitations can be expected for language models trained on general domains to process dialogue. First, most of these models are pre-trained on formal writing, which is notably different from colloquial writing in dialogue; thus, fine-tuning for the end tasks is often not sufficient enough to build robust dialogue models. Second, unlike sentences in a wiki or news article written by one author with a coherent topic, utterances in a dialogue are from multiple speakers who may talk about different topics in distinct manners such that they should not be represented by simply concatenating, but rather as sub-documents interconnected to one another.

This paper presents a novel approach to the latest transformers that learns hierarchical embeddings for tokens and utterances for a better understanding in dialogue contexts. While fine-tuning for span-based QA, every utterance as well as the question are separated encoded and multi-head attentions and additional transformers are built on the token and utterance embeddings respectively to provide a more comprehensive view of the dialogue to the QA model. As a result, our model achieves a new state-of-the-art result on a span-based QA task where the evidence documents are multiparty dialogue. The contributions of this paper are:[1]

- New pre-training tasks are introduced to improve the quality of both token-level and utterance-level embeddings generated by the transformers, that better suit to handle dialogue contexts (§2.1).

- A new multi-task learning approach is proposed to fine-tune the language model for span-based QA that takes full advantage of the hierarchical embeddings created from the pre-training (§2.2).

- Our approach significantly outperforms the previous state-of-the-art models using BERT and RoBERTa on a span-based QA task using dialogues as evidence documents (§3).

---

[1] All our resources including the source codes and the dataset with the experiment split are available at
`https://github.com/emorynlp/friendsqa`

(a) Token-level MLM (§2.1.1)

(b) Utterance-level MLM (§2.1.2)
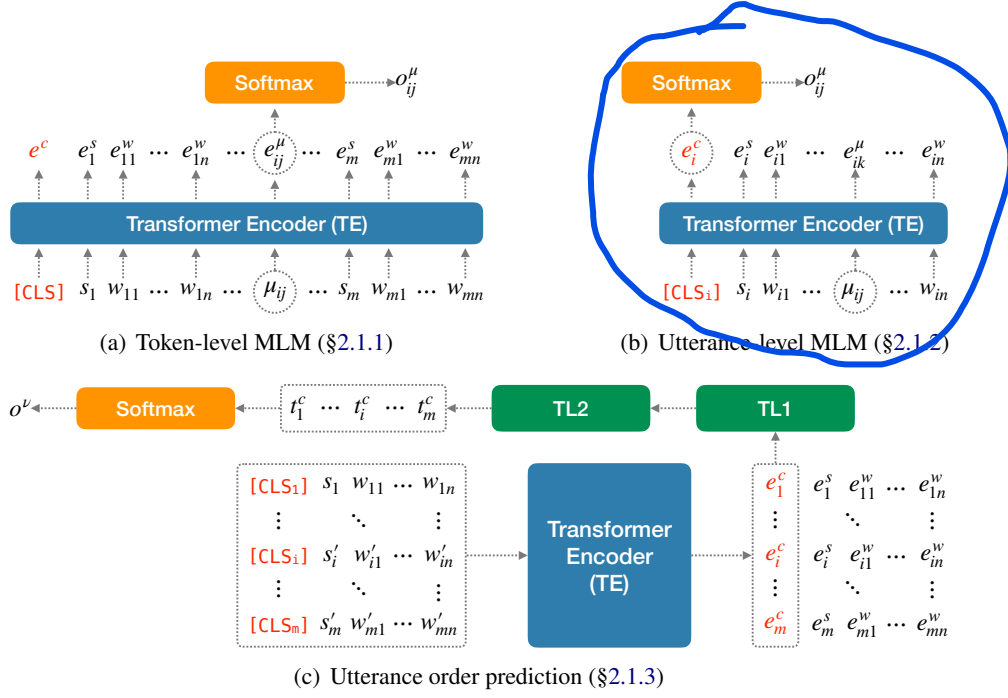
(c) Utterance order prediction (§2.1.3)

Figure 1: The overview of our models for the three pre-training tasks (Section 2.1).

2 Transformers for Learning Dialogue

This section introduces a novel approach for pre-training (Section 2.1) and fine-tuning (Section 2.2) transformers to effectively learn dialogue contexts. Our approach has been evaluated with two kinds of transformers, BERT (Devlin et al., 2019) and RoBERTa (Liu et al., 2019), and shown significant improvement to a question answering task (QA) on multiparty dialogue (Section 3).

2.1 Pre-training Language Models

Pre-training involves 3 tasks in sequence, the token-level masked language modeling (MLM; §2.1.1), the utterance-level MLM (§2.1.2), and the utterance order prediction (§2.1.3), where the trained weights from each task are transferred to the next task. Note that the weights of publicly available transformer encoders are adapted to train the token-level MLM, which allows our QA model to handle languages in both dialogues, used as evidence documents, and questions written in formal writing. Transformers from BERT and RoBERTa are trained with static and dynamic MLM respectively, as described by Devlin et al. (2019); Liu et al. (2019).

2.1.1 Token-level Masked LM

Figure 1(a) illustrates the token-level MLM model. Let $D = \{U_1, \ldots, U_m\}$ be a dialogue where $U_i = \{s_i, w_{i1}, \ldots, w_{in}\}$ is the i'th utterance in D, s_i is the speaker of U_i, and w_{ij} is the j'th token in U_i.

All speakers and tokens in D are appended in order with the special token CLS, representing the entire dialogue, which creates the input string sequence $I = \{\text{CLS}\} \oplus U_1 \oplus \ldots \oplus U_n$. For every $w_{ij} \in I$, let $I_{ij}^\mu = (I \setminus \{w_{ij}\}) \cup \{\mu_{ij}\}$, where μ_{ij} is the masked token substituted in place of w_{ij}. I_{ij}^μ is then fed into the transformer encoder (TE), which generates a sequence of embeddings $\{e^c\} \oplus E_1 \oplus \ldots \oplus E_m$ where $E_i = \{e_i^s, e_{i1}^w, .., e_{in}^w\}$ is the embedding list for U_i, and $(e^c, e_i^s, e_{ij}^w, e_{ij}^\mu)$ are the embeddings of $(\text{CLS}, s_i, w_{ij}, \mu_{ij})$ respectively. Finally, e_{ij}^μ is fed into a softmax layer that generates the output vector $o_{ij}^\mu \in \mathbb{R}^{|V|}$ to predict μ_{ij}, where V is the set of all vocabularies in the dataset.[2]

2.1.2 Utterance-level Masked LM

The token-level MLM (t-MLM) learns attentions among all tokens in D regardless of the utterance boundaries, allowing the model to compare every token to a broad context; however, it fails to catch unique aspects about individual utterances that can be important in dialogue. To learn an embedding for each utterance, the utterance-level MLM model is trained (Figure 1(b)). Utterance embeddings can be used independently and/or in sequence to match contexts in the question and the dialogue beyond the token-level, showing an advantage in finding utterances with the correct answer spans (§2.2.1).

---

[2] n: the maximum number of words in every utterance,
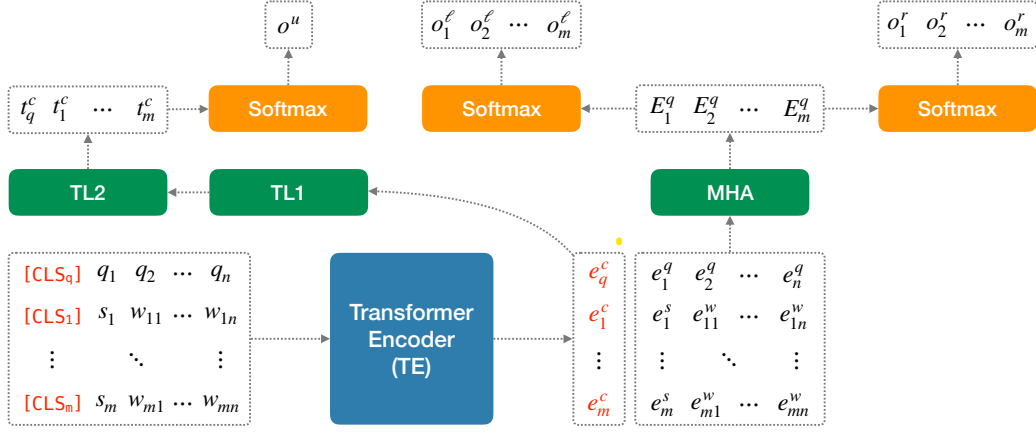m: the maximum number of utterances in every dialogue.

5710

Figure 2: The overview of our fine-tuning model exploiting multi-task learning (Section 2.2).

For every utterance U_i, the masked input sequence $I_{ij}^\mu = \{\text{CLS}_i\} \oplus \{(U_i \setminus \{w_{ij}\}) \cup \mu_{ij}\}$ is generated. Note that CLS_i now represents U_i instead of D and I_{ij}^μ is much shorter than the one used for t-MLM. I_{ij}^μ is fed into TE, already trained by t-MLM, and the embedding sequence $E_i = \{e_i^c, e_i^s, e_{i1}^w, .., e_{in}^w\}$ is generated. Finally, e_i^c, instead of e_{ij}^μ, is fed into a softmax layer that generates o_{ij}^μ to predict μ_{ij}. The intuition behind the utterance-level MLM is that once e_i^c learns enough contents to accurately predict any token in U_i, it consists of most essential features about the utterance; thus, e_i^c can be used as the embedding of U_i.

2.1.3 Utterance Order Prediction

The embedding e_i^c from the utterance-level MLM (u-MLM) learns contents within U_i, but not across other utterances. In dialogue, it is often the case that a context is completed by multiple utterances; thus, learning attentions among the utterances is necessary. To create embeddings that contain cross-utterance features, the utterance order prediction model is trained (Figure 1(c)). Let $D = D_1 \oplus D_2$ where D_1 and D_2 comprise the first and the second halves of the utterances in D, respectively. Also, let $D' = D_1 \oplus D_2'$ where D_2' contains the same set of utterances as D_2 although the ordering may be different. The task is whether or not D' preserves the same order of utterances as D.

For each $U_i \in D'$, the input $I_i = \{\text{CLS}_i\} \oplus U_i$ is created and fed into TE, already trained by u-MLM, to create the embeddings $E_i = \{e_i^c, e_i^s, e_{i1}^w, .., e_{in}^w\}$. The sequence $E^c = \{e_1^c, \ldots, e_n^c\}$ is fed into two transformer layers, TL1 and TL2, that generate the new utterance embedding list $T^c = \{t_1^c, \ldots, t_n^c\}$. Finally, T^c is fed into a softmax layer that generates $o^\nu \in \mathbb{R}^2$ to predict whether or not D' is in order.

2.2 Fine-tuning for QA on Dialogue

Fine-tuning exploits multi-task learning between the utterance ID prediction (§2.2.1) and the token span prediction (§2.2.2), which allows the model to train both the utterance- and token-level attentions. The transformer encoder (TE) trained by the utterance order prediction (UOP) is used for both tasks. Given the question $Q = \{q_1, \ldots, q_n\}$ (q_i is the i'th token in Q) and the dialogue $D = \{U_1, \ldots, U_m\}$, Q and all U_* are fed into TE that generates $E_q = \{e_q^c, e_1^q, .., e_n^q\}$ and $E_i = \{e_i^c, e_i^s, e_{i1}^w, .., e_{in}^w\}$ for Q and every U_i, respectively.

2.2.1 Utterance ID Prediction

The utterance embedding list $E^c = \{e_q^c, e_1^c, .., e_n^c\}$ is fed into TL1 and TL2 from UOP that generate $T^c = \{t_q^c, t_1^c, .., t_n^c\}$. T^c is then fed into a softmax layer that generates $o^u \in \mathbb{R}^{m+1}$ to predict the ID of the utterance containing the answer span if exists; otherwise, the 0'th label is predicted, implying that the answer span for Q does not exist in D.

2.2.2 Token Span Prediction

For every E_i, the pair (E_q', E_i') is fed into the multi-head attention layer, MHA, where $E_q' = E_q \setminus \{e_q^c\}$ and $E_i' = E_i \setminus \{e_i^c\}$. MHA (Vaswani et al., 2017) then generates the attended embedding sequences, $T_1^a, \ldots, T_m^a$, where $T_i^a = \{t_i^s, t_{i1}^w, .., t_{in}^w\}$. Finally, each T_i^a is fed into two softmax layers, SL and SR, that generate $o_i^\ell \in \mathbb{R}^{n+1}$ and $o_i^r \in \mathbb{R}^{n+1}$ to predict the leftmost and the rightmost tokens in U_i respectively, that yield the answer span for Q. It is possible that the answer spans are predicted in multiple utterances, in which case, the span from the utterance that has the highest score for the utterance ID prediction is selected, which is more efficient than the typical dynamic programming approach.

5711

3 Experiments

3.1 Corpus

Despite of all great work in QA, only two datasets are publicly available for machine comprehension that take dialogues as evidence documents. One is DREAM comprising dialogues for language exams with multiple-choice questions (Sun et al., 2019). The other is FRIENDSQA containing transcripts from the TV show *Friends* with annotation for span-based question answering (Yang and Choi, 2019). Since DREAM is for a reading comprehension task that does not need to find the answer contents from the evidence documents, it is not suitable for our approach; thus, FRIENDSQA is chosen.

Each scene is treated as an independent dialogue in FRIENDSQA. Yang and Choi (2019) randomly split the corpus to generate training, development, and evaluation sets such that scenes from the same episode can be distributed across those three sets, causing inflated accuracy scores. Thus, we re-split them by episodes to prevent such inflation. For fine-tuning (§2.2), episodes from the first four seasons are used as described in Table 1. For pre-training (§2.1), all transcripts from Seasons 5-10 are used as an additional training set.

Set	D	Q	A	E
Training	973	9,791	16,352	1 - 20
Development	113	1,189	2,065	21 - 22
Evaluation	136	1,172	1,920	23 - *

Table 1: New data split for FriendsQA. D/Q/A: # of dialogues/questions/answers, E: episode IDs.

3.2 Models

The weights from the $BERT_{base}$ and $RoBERTa_{base}$ models (Devlin et al., 2019; Liu et al., 2019) are transferred to all models in our experiments. Four baseline models, BERT, $BERT_{pre}$, RoBERTa, and $RoBERTa_{pre}$, are built, where all models are fine-tuned on the datasets in Table 1 and the $*_{pre}$ models are pre-trained on the same datasets with the additional training set from Seasons 5-10 (§3.1). The baseline models are compared to $BERT_{our}$ and $RoBERTa_{our}$ that are trained by our approach.[3]

3.3 Results

Table 2 shows results achieved by all the models. Following Yang and Choi (2019), exact matching (EM), span matching (SM), and utterance matching (UM) are used as the evaluation metrics. Each

---

[3]Detailed experimental setup are provided in Appendices.

model is developed three times and their average score as well as the standard deviation are reported. The performance of RoBERTa$*$ is generally higher than BERT$*$ although $RoBERTa_{base}$ is pre-trained with larger datasets including CC-NEWS (Nagel, 2016), OPENWEBTEXT (Gokaslan and Cohen, 2019), and STORIES (Trinh and Le, 2018) than $BERT_{base}$ such that results from those two types of transformers cannot be directly compared.

Model	EM	SM	UM
BERT	43.3(±0.8)	59.3(±0.6)	70.2(±0.4)
$BERT_{pre}$	45.6(±0.9)	61.2(±0.7)	71.3(±0.6)
$BERT_{our}$	**46.8**(±1.3)	**63.1**(±1.1)	**73.3**(±0.7)
RoBERTa	52.6(±0.7)	68.2(±0.3)	80.9(±0.8)
$RoBERTa_{pre}$	52.6(±0.7)	68.6(±0.6)	81.7(±0.7)
$RoBERTa_{our}$	**53.5**(±0.7)	**69.6**(±0.8)	**82.7**(±0.5)

Table 2: Accuracies (± standard deviations) achieved by the BERT and RoBERTa models.

The $*_{pre}$ models show marginal improvement over their base models, implying that pre-training the language models on FRIENDSQA with the original transformers does not make much impact on this QA task. The models using our approach perform noticeably better than the baseline models, showing 3.8% and 1.4% improvements on SM from BERT and RoBERTa, respectively.

Type	Dist.	EM	SM	UM
Where	18.16	66.1(±0.5)	79.9(±0.7)	89.8(±0.7)
When	13.57	63.3(±1.3)	76.4(±0.6)	88.9(±1.2)
What	18.48	56.4(±1.7)	74.0(±0.5)	87.7(±2.1)
Who	18.82	55.9(±0.8)	66.0(±1.7)	79.9(±1.1)
How	15.32	43.2(±2.3)	63.2(±2.5)	79.4(±0.7)
Why	15.65	33.3(±2.0)	57.3(±0.8)	69.8(±1.8)

Table 3: Results from the $RoBERTa_{our}$ model by different question types.

Table 3 shows the results achieved by $RoBERTa_{our}$ w.r.t. question types. UM drops significantly for Why that often spans out to longer sequences and also requires deeper inferences to answer correctly than the others. Compared to the baseline models, our models show more well-around performance regardless the question types.[4]

3.4 Ablation Studies

Table 4 shows the results from ablation studies to analyze the impacts of the individual approaches. $BERT_{pre}$ and $RoBERTa_{pre}$ are the same as in Table 2, that are the transformer models pre-trained by

---

[4]Question type results for all models are in Appendices.

the token-level masked LM (§2.1.1) and fine-tuned by the token span prediction (§2.2.2). BERT$_{uid}$ and RoBERTa$_{uid}$ are the models that are pre-trained by the token-level masked LM and jointly fine-tuned by the token span prediction as well as the utterance ID prediction (UID: §2.2.1). Given these two types of transformer models, the utterance-level masked LM (ULM: §2.1.2) and the utterance order prediction (UOP: §2.1.3) are separately evaluated.

Model	EM	SM	UM
BERT$_{pre}$	45.6($\pm$0.9)	61.2($\pm$0.7)	71.3($\pm$0.6)
$\oplus$ULM	45.7($\pm$0.9)	61.8($\pm$0.9)	71.8($\pm$0.5)
$\oplus$ULM$\oplus$UOP	45.6($\pm$0.9)	61.7($\pm$0.7)	71.7($\pm$0.6)
BERT$_{uid}$	45.7($\pm$0.8)	61.1($\pm$0.8)	71.5($\pm$0.5)
$\oplus$ULM	46.2($\pm$1.1)	62.4($\pm$1.2)	72.5($\pm$0.8)
$\oplus$ULM$\oplus$UOP	**46.8**($\pm$1.3)	**63.1**($\pm$1.1)	**73.3**($\pm$0.7)
RoBERTa$_{pre}$	52.6($\pm$0.7)	68.6($\pm$0.6)	81.7($\pm$0.7)
$\oplus$ULM	52.9($\pm$0.8)	68.7($\pm$1.1)	81.7($\pm$0.6)
$\oplus$ULM$\oplus$UOP	52.5($\pm$0.8)	68.8($\pm$0.5)	81.9($\pm$0.7)
RoBERTa$_{uid}$	52.8($\pm$0.9)	68.7($\pm$0.8)	81.9($\pm$0.5)
$\oplus$ULM	53.2($\pm$0.6)	69.2($\pm$0.7)	82.4($\pm$0.5)
$\oplus$ULM$\oplus$UOP	**53.5**($\pm$0.7)	**69.6**($\pm$0.8)	**82.7**($\pm$0.5)

Table 4: Results for the ablation studies. Note that the $\star_{uid}\oplus$ULM$\oplus$UOP models are equivalent to the $\star_{our}$ models in Table 2, respectively.

These two dialogue-specific LM approaches, ULM and UOP, give very marginal improvement over the baseline models, that is rather surprising. However, they show good improvement when combined with UID, implying that pre-training language models may not be enough to enhance the performance by itself but can be effective when it is coupled with an appropriate fine-tuning approach. Since both ULM and UOP are designed to improve the quality of utterance embeddings, it is expected to improve the accuracy for UID as well. The improvement on UM is indeed encouraging, giving 2% and 1% boosts to BERT$_{pre}$ and RoBERTa$_{pre}$, respectively and consequently improving the other two metrics.

3.5 Error Analysis

As shown in Table 3, the major errors are from the three types of questions, `who`, `how`, and `why`; thus, we select 100 dialogues associated with those question types that our best model, RoBERTa$_{our}$, incorrectly predicts the answer spans for. Specific examples are provided in Tables 12, 13 and 14 (§A.3). Following Yang et al. (2019), errors are grouped into 6 categories, entity resolution, paraphrase and partial match, cross-utterance reasoning, question bias, noise in annotation, and miscellaneous.

Table 5 shows the errors types and their ratios with respect to the question types. Two main error types are entity resolution and cross-utterance reasoning. The entity resolution error happens when many of the same entities are mentioned in multiple utterances. This error also occurs when the QA system is asked about a specific person, but predicts wrong people where there are so many people appearing in multiple utterances. The cross-utterance reasoning error often happens with the `why` and `how` questions where the model relies on pattern matching mostly and predicts the next utterance span of the matched pattern.

Error Types	Who	How	Why
Entity Resolution	**34%**	23%	20%
Paraphrase and Partial Match	14%	14%	13%
Cross-Utterance Reasoning	25%	**28%**	**27%**
Question Bias	11%	13%	17%
Noise in Annotation	4%	7%	9%
Miscellaneous	12%	15%	14%

Table 5: Error types and their ratio with respect to the three most challenging question types.

4 Conclusion

This paper introduces a novel transformer approach that effectively interprets hierarchical contexts in multiparty dialogue by learning utterance embeddings. Two language modeling approaches are proposed, utterance-level masked LM and utterance order prediction. Coupled with the joint inference between token span prediction and utterance ID prediction, these two language models significantly outperform two of the state-of-the-art transformer approaches, BERT and RoBERTa, on a span-based QA task called *FriendsQA*. We will evaluate our approach on other machine comprehension tasks using dialogues as evidence documents to further verify the generalizability of this work.

Acknowledgments

References

Eunsol Choi, He He, Mohit Iyyer, Mark Yatskar, Wen-tau Yih, Yejin Choi, Percy Liang, and Luke Zettlemoyer. 2018. Quac: Question answering in context.

Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing.

Alexis CONNEAU and Guillaume Lample. 2019. Cross-lingual language model pretraining. In H. Wallach, H. Larochelle, A. Beygelzimer, F. d'Alché-Buc, E. Fox, and R. Garnett, editors, *Advances in Neural Information Processing Systems 32*, pages 7057–7067. Curran Associates, Inc.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, NAACL'19, pages 4171–4186.

Aaron Gokaslan and Vanya Cohen. 2019. *OpenWebText Corpus*.

Mohit Iyyer, Wen-tau Yih, and Ming-Wei Chang. 2017. Search-based neural structured learning for sequential question answering. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1821–1831, Vancouver, Canada. Association for Computational Linguistics.

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

Tom Koisk, Jonathan Schwarz, Phil Blunsom, Chris Dyer, Karl Moritz Hermann, Gbor Melis, and Edward Grefenstette. 2018. The narrativeqa reading comprehension challenge. *Transactions of the Association for Computational Linguistics*, 6:317328.

Zhenzhong Lan, Mingda Chen, Sebastian Goodman, Kevin Gimpel, Piyush Sharma, and Radu Soricut. 2019. Albert: A lite bert for self-supervised learning of language representations.

Yinhan Liu, Myle Ott, Naman Goyal, Jingfei Du, Mandar Joshi, Danqi Chen, Omer Levy, Mike Lewis, Luke Zettlemoyer, and Veselin Stoyanov. 2019. RoBERTa: A Robustly Optimized BERT Pretraining Approach. *arXiv*, 1907.11692.

Sebastian Nagel. 2016. *News Dataset Available*.

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. In *Proceedings of the Workshop on Cognitive Computation: Integrating neural and symbolic approaches 2016 co-located with the 30th Annual Conference on Neural Information Processing Systems (NIPS 2016), Barcelona, Spain, December 9, 2016.*

Pranav Rajpurkar, Robin Jia, and Percy Liang. 2018. Know what you dont know: Unanswerable questions for squad. *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers).*

Pranav Rajpurkar, Jian Zhang, Konstantin Lopyrev, and Percy Liang. 2016. Squad: 100,000+ questions for machine comprehension of text. *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing.*

Siva Reddy, Danqi Chen, and Christopher D. Manning. 2019. Coqa: A conversational question answering challenge. *Transactions of the Association for Computational Linguistics*, 7:249266.

Kai Sun, Dian Yu, Jianshu Chen, Dong Yu, Yejin Choi, and Claire Cardie. 2019. DREAM: A Challenge Data Set and Models for Dialogue-Based Reading Comprehension. *Transactions of the Association for Computational Linguistics*, 7:217–231.

Alon Talmor and Jonathan Berant. 2018. The web as a knowledge-base for answering complex questions. *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long Papers).*

Trieu H. Trinh and Quoc V. Le. 2018. A Simple Method for Commonsense Reasoning. *arXiv*, 1806.02847.

Adam Trischler, Tong Wang, Xingdi Yuan, Justin Harris, Alessandro Sordoni, Philip Bachman, and Kaheer Suleman. 2017. Newsqa: A machine comprehension dataset. *Proceedings of the 2nd Workshop on Representation Learning for NLP.*

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, Lukasz Kaiser, and Illia Polosukhin. 2017. Attention is all you need. In *Proceedings of the 31st International Conference on Neural Information Processing Systems*, NIPS'17, pages 6000–6010, USA. Curran Associates Inc.

Zhengzhe Yang and Jinho D. Choi. 2019. FriendsQA: Open-domain question answering on TV show transcripts. In *Proceedings of the 20th Annual SIGdial Meeting on Discourse and Dialogue*, pages 188–197, Stockholm, Sweden. Association for Computational Linguistics.

Zhilin Yang, Zihang Dai, Yiming Yang, Jaime Carbonell, Russ R Salakhutdinov, and Quoc V Le. 2019. Xlnet: Generalized autoregressive pretraining for language understanding. In H. Wallach, H. Larochelle, A. Beygelzimer, F. d'Alché-Buc, E. Fox, and R. Garnett, editors, *Advances in Neural Information Processing Systems 32*, pages 5754–5764. Curran Associates, Inc.