

# Self-supervised Knowledge Triplet Learning for Zero-shot Question Answering

Pratyay Banerjee and Chitta Baral

Department of Computer Science, Arizona State University

pbanerj6, chitta@asu.edu

## Abstract

The aim of all Question Answering (QA) systems is to be able to generalize to unseen questions. Most of the current methods rely on learning every possible scenario which is reliant on expensive data annotation. Moreover, such annotations can introduce unintended bias which makes systems focus more on the bias than the actual task. In this work, we propose Knowledge Triplet Learning, a self-supervised task over knowledge graphs. We propose methods of how to use such a model to perform zero-shot QA and our experiments show considerable improvements over large pre-trained generative models.

## 1 Introduction

The ability to understand natural language and answer questions is one of the core focus in the field of natural language processing. To measure and study the different aspects of question answering, several datasets are developed, such as SQuAD (Rajpurkar et al., 2018), HotpotQA (Yang et al., 2018), and Natural Questions (Kwiatkowski et al., 2019) which require systems to perform extractive question answering. On the other hand datasets such as SocialQA (Sap et al., 2019b), CommonsenseQA (Talmor et al., 2018), Swag (Zellers et al., 2018) and Winogrande (Sakaguchi et al., 2019) require systems to choose the correct answer from a given set. These multiple-choice question answering datasets are very challenging, but recent large pre-trained language models such as BERT (Devlin et al., 2018), XLNET (Yang et al., 2019b) and RoBERTa (Liu et al., 2019b) have shown very strong performance on them. Moreover as shown in Winogrande (Sakaguchi et al., 2019), acquiring unbiased labels requires a “carefully designed crowdsourcing procedure”, which adds to the cost of data annotation. This is also quantified in other natural language tasks such as Natural Language Inference

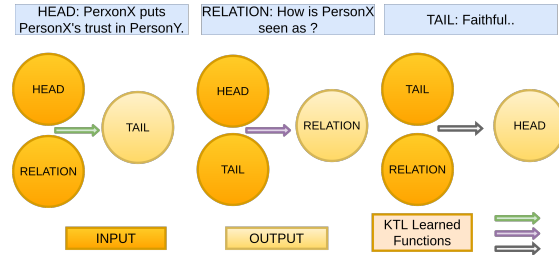


Figure 1: Knowledge Triplet Learning Framework, where given a triple  $(h, r, t)$  we learn to generate one of the inputs given the other two.

(Gururangan et al., 2018) and Argument Reasoning Comprehension (Niven and Kao, 2019), where such annotation artifacts lead to “Clever Hans Effect” in the models (Kaushik and Lipton, 2018; Poliak et al., 2018).

One way to resolve this is to design and create datasets in a clever way, such as in Winogrande (Sakaguchi et al., 2019), the other way is to ignore the data annotations and to build systems to perform unsupervised question answering (Teney and Hengel, 2016; Lewis et al., 2019). In this paper, we focus on building unsupervised zero-shot multiple-choice question answering systems.

The task of unsupervised question answering is very challenging. Recent work (Fabbri et al., 2020; Lewis et al., 2019) try to generate a synthetic dataset using a text corpus such as Wikipedia, to solve extractive QA. Other work (Bosselut and Choi, 2019; Schwartz et al., 2020) uses large pre-trained generative language models such as GPT-2 (Radford et al., 2019) to generate knowledge, questions, and answers and compare against the given choices.

In this work, we utilize the information present in Knowledge Graphs such as ATOMIC (Sap et al., 2019a) and ConceptNET (Liu and Singh, 2004) and define a new task of Knowledge Triplet Learning. Knowledge Triplet Learning is similar to Knowl-

edge Representation Learning but not limited to it. Knowledge Representation Learning (Lin et al., 2018) learns the low-dimensional projected and distributed representations of entities and relations defined in a knowledge graph. As shown in Figure 1, we define a triplet  $(h, r, t)$ , and given any two we try to recover the third. This forces the system to learn the all possible relations between the three inputs. We map the question answering task to Knowledge Triplet Learning, by mapping the *context*, *question* and *answer* to  $(h, r, t)$  respectively. We define two different ways to perform self-supervised Knowledge Triplet Learning. This task can be designed as a representation generation task or a language modeling task. We compare both the strategies in this work. We show how to use models trained on this task to perform zero-shot question answering without any additional knowledge or additional supervision. We also show how models pre-trained on this task perform considerably well compared to strong pre-trained language models on few-shot learning. We evaluate our approach on the SocialIQA dataset.

The contributions of this paper are summarized as follows:

- We define the Knowledge Triplet Learning over Knowledge Graph and show how to use it for zero-shot question answering.
- We compare two strategies for the above task.
- We achieve state-of-the-art results for zero-shot and propose a strong baseline for the few-shot question answering task.

## 2 Knowledge Triplet Learning

We define the task of Knowledge Triplet Learning (KTL) in this section. We define  $G = (V, E)$  as a Knowledge Graph, where  $V$  is the set of vertices,  $E$  is the set of edges.  $V$  consists of entities which can be phrases or named-entities depending on the given input Knowledge Graph. Let  $S$  be a set of fact triples,  $S \subseteq V \times E \times V$  with the format  $(h, r, t)$ , where  $h$  and  $t$  belong to set of vertices  $V$  and  $r$  belongs to set of edges. The  $h$  and  $t$  indicates the head and tail entities, whereas  $r$  indicates the relation between these entities.

For example, from the ATOMIC knowledge graph, *(PersonX puts PersonX's trust in PersonY, How is PersonX seen as?, faithful)* is one such triple. Here the head is *PersonX puts PersonX's trust in PersonY*, relation is **How is PersonX seen as?** and the tail is *faithful*. Do note  $V$  does not

contain homogenous entities, i.e, both *faithful* and *PersonX puts PersonX's trust in PersonY* belong to  $V$ .

We define the task of KTL as follows: Given input a triple  $(h, r, t)$ , we learn the following three functions.

$$\begin{aligned} f_t(h, r) &\Rightarrow t \\ f_h(r, t) &\Rightarrow h \\ f_r(h, t) &\Rightarrow r \end{aligned} \quad (1)$$

That is, each function learns to generate one component of the triple given the other two. The intuition behind learning these three functions is as follows. Let us take the above example: *(PersonX puts PersonX's trust in PersonY, How is PersonX seen as?, faithful)*. The first function  $f_t(h, r)$  learns to generate the answer  $t$  given the context and the question. The second function  $f_h(r, t)$  learns to generate one context where the question and the answer may be valid. The final function  $f_r(h, t)$  is a Jeopardy-style generating the question which connects the context and the answer.

In Multiple-choice QA, given the context, two answer choices may be true for two different questions. Similarly, given the question, two answer choices may be true for two different contexts. For example, given the context: *PersonX puts PersonX's trust in PersonY*, the answers *PersonX is considered trustworthy by others* and *PersonX is polite* are true for two different questions **How does this affect others?** and **How is PersonX seen as?**. Learning these three functions enables us to score these relations between the context, question, and answers.

### 2.1 Using KTL to perform QA

After learning this function in a self-supervised way, we can use them to perform question answering. Given a triple  $(h, r, t)$ , we define the following scoring function:

$$\begin{aligned} D_t &= D(t, f_t(h, r)) \\ D_h &= D(h, f_h(r, t)) \\ D_r &= D(r, f_r(h, t)) \\ \text{score}(h, r, t) &= D_t * D_h * D_r \end{aligned} \quad (2)$$

where  $h$  is the context,  $r$  is the question and  $t$  is one of the answer options.  $D$  is a distance function which measures the distance between the generated output and the ground-truth. The distance function varies depending on the instantiation of the

framework, which we will study in the following sections. The final answer is selected as:

$$ans = \arg \min_t (score(h, r, t)) \quad (3)$$

Since the scores are the distance from the ground-truth we select the answer which has the minimum score.

In the following sections, we define the different ways we can implement this framework.

## 2.2 Knowledge Representation Learning

In this implementation, we use Knowledge representation learning to learn equation (1). In contrast to Knowledge representation learning, where systems try to learn a score function  $f_r(h, t)$ , i.e, is the fact triple  $(h, r, t)$  true or false; in this work we learn to generate the inputs vector representations, i.e,  $f_r(h, t) \Rightarrow r$ . We can view equation 1 as generator functions, which given the two input learns to generate a vector representation of the third. As our triples  $(h, r, t)$  can have a many to many relations between each pair, we first project the two inputs from input encoding space to a different space similar to the work of TransD (Ji et al., 2015). We use a Transformer encoder  $Enc$  to encode our triples to the encoding space. We learn two projection functions,  $M_{i1}$  and  $M_{i2}$  to project the two inputs, and a third projection function  $M_o$  to project the entity to be generated. We combine the two projected inputs using a function  $C$ . These functions can be implemented using feedforward networks.

$$\begin{aligned} I_{e1} &= Enc(I_1), I_{e2} = Enc(I_2), O_e = Enc(O) \\ I_{e1} &= M_{i1}(I_{e1}), I_{e2} = M_{i2}(I_{e2}), O_p = M_o(O_e) \\ \hat{O} &= C(I_{e1}, I_{e2}) \\ loss &= LossF(\hat{O}, O_p) \end{aligned}$$

where  $I_i$  is the input,  $\hat{O}$  is the generated output vector and  $O_p$  is the projected vector.  $M$  and  $C$  functions are learned using fully connected networks. In our implementation, we use RoBERTa as the  $Enc$  transformer, with the output representation of the  $[cls]$  token as the phrase representation.

We train this model using two types of loss functions, L2Loss where we try to minimize the L2 norm between the generated and the projected ground-truth, and Noise Contrastive Estimation (Gutmann and Hyvärinen, 2010) where along with the ground-truth we have  $k$  noise-samples. These noise samples are selected from other  $(h, r, t)$  triples such that the target output is not another true

fact triple, i.e,  $(h, r, t_{noise})$  is false. The NCELoss is defined as:

$$\begin{aligned} NCELoss(\hat{O}, O_p, [N_0 \dots N_k]) &= \\ -\log \frac{\exp(sim(\hat{O}, O_p))}{\exp(sim(\hat{O}, O_p)) + \sum_{k \in N} \exp(sim(\hat{O}, N_k))} \end{aligned}$$

where  $N_k$  are the projected noise samples,  $sim$  is the similarity function which can be the L2 norm or Cosine similarity,  $\hat{O}$  is the generated output vector and  $O_p$  is the projected vector.

The  $D$  distance function (2) for such a model is defined by the distance function used in the loss function. For L2Loss, it is the L2 norm, and in the case of NCELoss, we use  $1 - sim$  function.

## 2.3 Span Masked Language Modeling

In Span Masked Language Modeling, we model the equation 1 as a masked language modeling task. We tokenize and concatenate the triple  $(h, r, t)$  with a separator token between them, i.e,  $[cls][h][sep][r][sep][t][sep]$ . For the function  $f_r(h, t) \Rightarrow r$ , we mask all the tokens present in  $r$ , i.e,  $[cls][h][sep][mask][sep][t][sep]$ . We feed this tokens to a Transformer encoder  $Enc$ , and use a feed forward network to unmask the sequence of tokens. Similarly we mask  $h$  to learn  $f_h$  and  $t$  to learn  $f_t$ .

We train the same Transformer encoder to perform all the three functions. We use the cross-entropy loss to train the model:

$$\begin{aligned} CELoss(h, r, mask(t), t) &= \\ -\frac{1}{n} \sum_{i=1}^n \log_2 P_{MLM}(t_i | h, r, t_{1..t_i..tn}) \end{aligned}$$

where  $P_{MLM}$  is the masked language modeling probability of the token  $t_i$ , given the unmasked tokens  $h$  and  $r$  and other masked tokens in  $t$ . Do note we do not do progressive unmasking, i.e, all the masked tokens are jointly predicted.

The  $D$  distance function (2) for this model is same as the loss function defined above.

## 3 Datasets

### 3.1 SocialQA

To study our framework we evaluate it on SocialQA. This dataset is about reasoning over social interactions and the implications of social events. Each instance in this dataset contains a context  $C$  which is a social situation, a question  $Q$  about this situation, and three answer options. There are several question types that are derived from the

different ATOMIC inference dimensions, such as Intent, Effect, Attributes, etc. There are 33,410 training samples and 1954 validation samples, with a withheld test set.

Though this dataset is derived from the ATOMIC knowledge graph, the fact triples present in the graph are considerably different than the context, question and answers present in the dataset as they are crowdsourced. The average length of the event description in ATOMIC is 10, max length is 18. Whereas in SocialQA the average length is 36 and max is 124. This shows the varied type of questions present in SocialQA, and possess a much more challenge in unsupervised learning.

## 4 Experiments

### 4.1 Baselines

We compare our models to three strong baselines. The first one is a pre-trained language model, GPT-2 (all sizes) which are scored using language modeling cross-entropy loss. We concatenate the context and question and find the cross-entropy loss for each of the answer choices and choose the answer which has the minimum loss. The second baseline is another pre-trained language model, RoBERTa-large. We follow the same Span Masked Language Model (SMLM) scoring using the pre-trained RoBERTa model. For the third baseline, we finetune the RoBERTa-large model using the original Masked Language Modeling task over our concatenated fact triples  $(h, r, t)$ .

### 4.2 KTL Training

We train the Knowledge Representation Learning (KRL) model using both L2Loss and NCELoss. For NCELoss we also train it with both L2 norm and Cosine similarity. Both the KRL model and SMLM model uses RoBERTa-large as the Transformer encoder. We train the model with the following hyper-parameters: batch sizes 16,32; learning rate in range:  $[1e-5, 5e-5]$ ; warm-up steps in range  $[0, 0.1]$ . We use the transformers package (Wolf et al., 2019). From the ATOMIC knowledge graph, we generate 595595 unique triplets. All these triplets are positive facts. We learn using these triplets. For NCE, we choose  $k$  equal to 10, i.e, 10 negative samples. We perform 3 hyper-parameter trials for each model, and train models with 3 different seeds  $[0, 21, 42]$ .

Type	Model	Train-Val	Val	Deviation
Baselines	Majority	33.5	33.6	
	GPT2-L	41.3	41.1	
	RoBERTa	33.6	33.4	
	RoBERTa-MLM	36.4	34.3	+/-0.8
Shwartz et al.	Self-Talk	46.5	46.2	
Ours	KRL-L2	43.2	43.8	+/-0.3
	KRL-NCE-L2	46.4	46.2	+/-0.2
	KRL-NCE-Cos	46.6	46.4	+/-0.2
	SMLM	<b>48.7</b>	<b>48.5</b>	+/-0.4
Supervised	Sup RoBERTa		76.6	+/-0.6
	Human		86.9	

Table 1: Accuracy comparison with our baseline models on the SocialQA dataset. We compare the models on the Zero-shot task. We compare them on both Train-Val split (35k) and the Validation split (2k), to enable measuring better statistical significance.

## 5 Discussion

Table 1 shows our evaluation and baseline comparisons for the zero-shot task. We can observe our KTL trained models perform significantly well compared to the baselines. When comparing the different KRL models, the NCELoss with Cosine similarity performs the best. This might be due to the additional supervision provided by the negative samples as the L2Loss model only tries to minimize the two projections. When comparing different KTL instantiations we can see the SMLM model performs the best overall but has a slightly higher deviation. We are analyzing our model to understand this phenomenon better. Our KRL model does perform equal to the current state-of-the-art model, Self-Talk (Shwartz et al., 2020) which uses two GPT type models. Our models have half the parameters compared to Self-Talk. We also compare our model on the entire Train-Val set of 35,364 questions in the zero-shot setting to better gauge the statistical significance of the different model accuracies.

Model	Val	Deviation
RoBERTa	44.4	+/-1.2
RoBERTa+MLM	46.8	+/-0.6
KRL-NCE-Cos	58.6	+/-0.8
SMLM	<b>69.1</b>	+/-0.4

Table 2: Accuracy comparison on Few-shot Question Answering. We re-use our Transformer encoder for the few-shot task with only 8 % train samples.



Table 2 compares the different pre-trained Transformer encoders in the few-shot question answering task. We randomly sample three sets of 2,400 samples as training data from the training set. We train our models on these three sets and measure the validation set accuracy. We can see the Transformer encoder trained on KTL perform significantly better than the baseline models in this setting. This shows that encoders trained on KTL are able to learn with a few samples. We plan to continue analyzing our models and evaluate the KTL framework on different datasets.

## 6 Related Work

### 6.1 Unsupervised Question Answering

Recent work on unsupervised question answering approach the problem in two ways, a domain adaptation or transfer learning problem (Chung et al., 2018), or a data augmentation problem (Yang et al., 2017; Dhingra et al., 2018; Wang et al., 2018; Alberti et al., 2019). The work of (Lewis et al., 2019; Fabbri et al., 2020; Puri et al., 2020) use style transfer or template-based question, context and answer triple generation, and learn using these to perform unsupervised extractive question answering. There is also another approach of learning generative models, generating the answer given a question or clarifying explanations and/or questions, such as GPT-2 (Radford et al., 2019) to perform unsupervised question answering (Shwartz et al., 2020; Bosselut and Choi, 2019; Bosselut et al., 2019). In contrast, our work focuses on learning from knowledge graphs and generate vector representations or sequences of tokens not restricted to the answer but including the context and the question using the masked language modeling objective.

### 6.2 Use of External Knowledge for Question Answering

There are several approaches to add external knowledge into models to improve question answering. Broadly they can be classified into two, learning from unstructured knowledge and structured knowledge. In learning from unstructured knowledge, recent large pre-trained language models (Peters et al., 2018; Radford et al., 2019; Devlin et al., 2018; Liu et al., 2019b; Clark et al., 2020; Lan et al., 2019; Joshi et al., 2020; Bosselut et al., 2019) learn general-purpose text encoders from a huge text corpus. On the other hand, learning from structured knowledge includes learning from structured

knowledge bases (Yang and Mitchell, 2017; Bauer et al., 2018; Mihaylov and Frank, 2018; Wang and Jiang, 2019; Sun et al., 2019) by learning knowledge enriched word embeddings. Using structured knowledge to refine pre-trained contextualized representations learned from unstructured knowledge is another approach (Peters et al., 2019; Yang et al., 2019a; Zhang et al., 2019; Liu et al., 2019a).

Another approach of using external knowledge includes retrieval of knowledge sentences from a text corpora (Mittra et al., 2019; Banerjee et al., 2019; Baral et al., 2020; Das et al., 2019; Chen et al., 2017; Lee et al., 2019; Banerjee and Baral, 2020; Banerjee, 2019) or knowledge triples from knowledge bases (Min et al., 2019; Wang et al., 2020) that are useful to answer a specific question. In our work, we use knowledge graphs to learn a self-supervised generative task to be able to perform zero-shot multiple-choice QA.

### 6.3 Knowledge Representation Learning

Over the years there are several methods discovered to perform the task of knowledge representation learning, i.e., embedding entities and relations in knowledge graphs to low-dimensional continuous vector space. We mention few of them here, such as TransE (Bordes et al., 2013) which views relations as a translation vector between head and tail entities, TransH (Wang et al., 2014) which overcomes TransE’s inability to model complex relations, and TransD (Ji et al., 2015) which aims to reduce the parameters by proposing two different mapping matrices for head and tail entities. For much detailed reading, we refer to this survey by Lin et al.. KRL has been used in various ways to generate natural answers (Yin et al., 2016; He et al., 2017) and generate factoid questions (Serban et al., 2016). In our work, we modify TransD and adapt it to our KTL framework to perform zero-shot QA.

## 7 Conclusion

In this work, we propose a new framework of Knowledge Triplet Learning over Knowledge Graphs. We show learning all three possible functions,  $f_r, f_h$ , and  $f_t$  helps the model to perform zero-shot multiple-choice question answering. We learn from the ATOMIC knowledge graph and evaluate our framework on the SocialQA dataset. Our framework achieves state-of-the-art in the zero-shot question answering task and sets a strong baseline in the few-shot question answering task.

## References

- Chris Alberti, Daniel Andor, Emily Pitler, Jacob Devlin, and Michael Collins. 2019. [Synthetic QA corpora generation with roundtrip consistency](#). In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 6168–6173, Florence, Italy. Association for Computational Linguistics.
- Pratyay Banerjee. 2019. [ASU at TextGraphs 2019 shared task: Explanation ReGeneration using language models and iterative re-ranking](#). In *Proceedings of the Thirteenth Workshop on Graph-Based Methods for Natural Language Processing (TextGraphs-13)*, pages 78–84, Hong Kong. Association for Computational Linguistics.
- Pratyay Banerjee and Chitta Baral. 2020. Knowledge fusion and semantic knowledge ranking for open domain question answering. *arXiv preprint arXiv:2004.03101*.
- Pratyay Banerjee, Kuntal Kumar Pal, Arindam Mitra, and Chitta Baral. 2019. [Careful selection of knowledge to solve open book question answering](#). In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 6120–6129, Florence, Italy. Association for Computational Linguistics.
- Chitta Baral, Pratyay Banerjee, Kuntal Kumar Pal, and Arindam Mitra. 2020. Natural language qa approaches using reasoning with external knowledge. *arXiv preprint arXiv:2003.03446*.
- Lisa Bauer, Yicheng Wang, and Mohit Bansal. 2018. [Commonsense for generative multi-hop question answering tasks](#). In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 4220–4230, Brussels, Belgium. Association for Computational Linguistics.
- Antoine Bordes, Nicolas Usunier, Alberto Garcia-Duran, Jason Weston, and Oksana Yakhnenko. 2013. Translating embeddings for modeling multi-relational data. In *Advances in neural information processing systems*, pages 2787–2795.
- Antoine Bosselut and Yejin Choi. 2019. Dynamic knowledge graph construction for zero-shot commonsense question answering. *arXiv preprint arXiv:1911.03876*.
- Antoine Bosselut, Hannah Rashkin, Maarten Sap, Chaitanya Malaviya, Asli Celikyilmaz, and Yejin Choi. 2019. Comet: Commonsense transformers for automatic knowledge graph construction. *arXiv preprint arXiv:1906.05317*.
- Danqi Chen, Adam Fisch, Jason Weston, and Antoine Bordes. 2017. [Reading Wikipedia to answer open-domain questions](#). In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1870–1879, Vancouver, Canada. Association for Computational Linguistics.
- Yu-An Chung, Hung-Yi Lee, and James Glass. 2018. [Supervised and unsupervised transfer learning for question answering](#). 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 1585–1594, New Orleans, Louisiana. Association for Computational Linguistics.
- Kevin Clark, Minh-Thang Luong, Quoc V Le, and Christopher D Manning. 2020. Electra: Pre-training text encoders as discriminators rather than generators. *arXiv preprint arXiv:2003.10555*.
- Rajarshi Das, Shehzaad Dhuliawala, Manzil Zaheer, and Andrew McCallum. 2019. Multi-step retriever-reader interaction for scalable open-domain question answering. *arXiv preprint arXiv:1905.05733*.
- Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2018. Bert: Pre-training of deep bidirectional transformers for language understanding. *arXiv preprint arXiv:1810.04805*.
- Bhuwan Dhingra, Danish Danish, and Dheeraj Rajagopal. 2018. [Simple and effective semi-supervised question answering](#). In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 2 (Short Papers)*, pages 582–587, New Orleans, Louisiana. Association for Computational Linguistics.
- Alexander R Fabbri, Patrick Ng, Zhiguo Wang, Ramesh Nallapati, and Bing Xiang. 2020. Template-based question generation from retrieved sentences for improved unsupervised question answering. *arXiv preprint arXiv:2004.11892*.
- Suchin Gururangan, Swabha Swayamdipta, Omer Levy, Roy Schwartz, Samuel Bowman, and Noah A Smith. 2018. Annotation artifacts in natural language inference data. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 2 (Short Papers)*, pages 107–112.
- Michael Gutmann and Aapo Hyvärinen. 2010. Noise-contrastive estimation: A new estimation principle for unnormalized statistical models. In *Proceedings of the Thirteenth International Conference on Artificial Intelligence and Statistics*, pages 297–304.
- Shizhu He, Cao Liu, Kang Liu, and Jun Zhao. 2017. Generating natural answers by incorporating copying and retrieving mechanisms in sequence-to-sequence learning. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 199–208.
- Guoliang Ji, Shizhu He, Liheng Xu, Kang Liu, and Jun Zhao. 2015. Knowledge graph embedding via dynamic mapping matrix. 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), pages 687–696.
- Mandar Joshi, Kenton Lee, Yi Luan, and Kristina Toutanova. 2020. [Contextualized representations using textual encyclopedic knowledge](#).
- Divyansh Kaushik and Zachary C Lipton. 2018. How much reading does reading comprehension require? a critical investigation of popular benchmarks. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 5010–5015.
- Tom Kwiatkowski, Jennimaria Palomaki, Olivia Redfield, Michael Collins, Ankur Parikh, Chris Alberti, Danielle Epstein, Illia Polosukhin, Jacob Devlin, Kenton Lee, et al. 2019. Natural questions: a benchmark for question answering research. *Transactions of the Association for Computational Linguistics*, 7:453–466.
- 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. *arXiv preprint arXiv:1909.11942*.
- Kenton Lee, Ming-Wei Chang, and Kristina Toutanova. 2019. [Latent retrieval for weakly supervised open domain question answering](#). In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 6086–6096, Florence, Italy. Association for Computational Linguistics.
- Patrick Lewis, Ludovic Denoyer, and Sebastian Riedel. 2019. [Unsupervised question answering by cloze translation](#). In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 4896–4910, Florence, Italy. Association for Computational Linguistics.
- Yankai Lin, Xu Han, Ruobing Xie, Zhiyuan Liu, and Maosong Sun. 2018. Knowledge representation learning: A quantitative review. *arXiv preprint arXiv:1812.10901*.
- Hugo Liu and Push Singh. 2004. Conceptnet a practical commonsense reasoning tool-kit. *BT Technology Journal*, 22:211–226.
- Weijie Liu, Peng Zhou, Zhe Zhao, Zhiruo Wang, Qi Ju, Haotang Deng, and Ping Wang. 2019a. K-bert: Enabling language representation with knowledge graph. *arXiv preprint arXiv:1909.07606*.
- Yinhan Liu, Myle Ott, Naman Goyal, Jingfei Du, Mandar Joshi, Danqi Chen, Omer Levy, Mike Lewis, Luke Zettlemoyer, and Veselin Stoyanov. 2019b. Roberta: A robustly optimized bert pretraining approach. *arXiv preprint arXiv:1907.11692*.
- Todor Mihaylov and Anette Frank. 2018. [Knowledgeable reader: Enhancing cloze-style reading comprehension with external commonsense knowledge](#). In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 821–832, Melbourne, Australia. Association for Computational Linguistics.
- Sewon Min, Danqi Chen, Luke Zettlemoyer, and Hananeh Hajishirzi. 2019. Knowledge guided text retrieval and reading for open domain question answering. *arXiv preprint arXiv:1911.03868*.
- Arindam Mitra, Pratyay Banerjee, Kuntal Kumar Pal, Swaroop Mishra, and Chitta Baral. 2019. Exploring ways to incorporate additional knowledge to improve natural language commonsense question answering. *arXiv preprint arXiv:1909.08855*.
- Timothy Niven and Hung-Yu Kao. 2019. Probing neural network comprehension of natural language arguments. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 4658–4664.
- Matthew Peters, Mark Neumann, Mohit Iyyer, Matt Gardner, Christopher Clark, Kenton Lee, and Luke Zettlemoyer. 2018. Deep contextualized word representations. 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 2227–2237.
- Matthew E. Peters, Mark Neumann, Robert Logan, Roy Schwartz, Vidur Joshi, Sameer Singh, and Noah A. Smith. 2019. [Knowledge enhanced contextual word representations](#). In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 43–54, Hong Kong, China. Association for Computational Linguistics.
- Adam Poliak, Jason Naradowsky, Aparajita Haldar, Rachel Rudinger, and Benjamin Van Durme. 2018. Hypothesis only baselines in natural language inference. In *Proceedings of the Seventh Joint Conference on Lexical and Computational Semantics*, pages 180–191.
- Raul Puri, Ryan Spring, Mostofa Patwary, Mohammad Shoeybi, and Bryan Catanzaro. 2020. Training question answering models from synthetic data. *arXiv preprint arXiv:2002.09599*.
- Alec Radford, Jeffrey Wu, Rewon Child, David Luan, Dario Amodei, and Ilya Sutskever. 2019. Language models are unsupervised multitask learners. *OpenAI Blog*, 1(8):9.
- Pranav Rajpurkar, Robin Jia, and Percy Liang. 2018. Know what you don’t know: Unanswerable questions for squad. *arXiv preprint arXiv:1806.03822*.



- Keisuke Sakaguchi, Ronan Le Bras, Chandra Bhagavatula, and Yejin Choi. 2019. Winogrande: An adversarial winograd schema challenge at scale. *arXiv preprint arXiv:1907.10641*.
- Maarten Sap, Ronan Le Bras, Emily Allaway, Chandra Bhagavatula, Nicholas Lourie, Hannah Rashkin, Brendan Roof, Noah A. Smith, and Yejin Choi. 2019a. Atomic: An atlas of machine commonsense for if-then reasoning. *ArXiv*, abs/1811.00146.
- Maarten Sap, Hannah Rashkin, Derek Chen, Ronan LeBras, and Yejin Choi. 2019b. Socialiqa: Commonsense reasoning about social interactions. *arXiv preprint arXiv:1904.09728*.
- Iulian Vlad Serban, Alberto Garcia-Duran, Caglar Gulcehre, Sungjin Ahn, Sarath Chandar, Aaron Courville, and Yoshua Bengio. 2016. Generating factoid questions with recurrent neural networks: The 30m factoid question-answer corpus. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 588–598.
- Vered Shwartz, Peter West, Ronan Le Bras, Chandra Bhagavatula, and Yejin Choi. 2020. Unsupervised commonsense question answering with self-talk. *arXiv preprint arXiv:2004.05483*.
- Haitian Sun, Tania Bedrax-Weiss, and William Cohen. 2019. [PullNet: Open domain question answering with iterative retrieval on knowledge bases and text](#). In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 2380–2390, Hong Kong, China. Association for Computational Linguistics.
- Alon Talmor, Jonathan Herzig, Nicholas Lourie, and Jonathan Berant. 2018. Commonsenseqa: A question answering challenge targeting commonsense knowledge. *arXiv preprint arXiv:1811.00937*.
- Damien Teney and Anton van den Hengel. 2016. Zero-shot visual question answering. *arXiv preprint arXiv:1611.05546*.
- Chao Wang and Hui Jiang. 2019. [Explicit utilization of general knowledge in machine reading comprehension](#). In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 2263–2272, Florence, Italy. Association for Computational Linguistics.
- Liang Wang, Sujian Li, Wei Zhao, Kewei Shen, Meng Sun, Ruoyu Jia, and Jingming Liu. 2018. [Multi-perspective context aggregation for semi-supervised cloze-style reading comprehension](#). In *Proceedings of the 27th International Conference on Computational Linguistics*, pages 857–867, Santa Fe, New Mexico, USA. Association for Computational Linguistics.
- Ruize Wang, Duyu Tang, Nan Duan, Zhongyu Wei, Xuanjing Huang, Cuihong Cao, Daxin Jiang, Ming Zhou, et al. 2020. K-adapter: Infusing knowledge into pre-trained models with adapters. *arXiv preprint arXiv:2002.01808*.
- Zhen Wang, Jianwen Zhang, Jianlin Feng, and Zheng Chen. 2014. Knowledge graph embedding by translating on hyperplanes. In *Twenty-Eighth AAAI conference on artificial intelligence*.
- Thomas Wolf, Lysandre Debut, Victor Sanh, Julien Chaumond, Clement Delangue, Anthony Moi, Pierric Cistac, Tim Rault, R’emi Louf, Morgan Funtowicz, and Jamie Brew. 2019. Huggingface’s transformers: State-of-the-art natural language processing. *ArXiv*, abs/1910.03771.
- An Yang, Quan Wang, Jing Liu, Kai Liu, Yajuan Lyu, Hua Wu, Qiaoqiao She, and Sujian Li. 2019a. [Enhancing pre-trained language representations with rich knowledge for machine reading comprehension](#). In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 2346–2357, Florence, Italy. Association for Computational Linguistics.
- Bishan Yang and Tom Mitchell. 2017. [Leveraging knowledge bases in LSTMs for improving machine reading](#). In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1436–1446, Vancouver, Canada. Association for Computational Linguistics.
- Zhilin Yang, Zihang Dai, Yiming Yang, Jaime Carbonell, Russ R Salakhutdinov, and Quoc V Le. 2019b. Xlnet: Generalized autoregressive pretraining for language understanding. In *Advances in neural information processing systems*, pages 5754–5764.
- Zhilin Yang, Junjie Hu, Ruslan Salakhutdinov, and William Cohen. 2017. [Semi-supervised QA with generative domain-adaptive nets](#). In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1040–1050, Vancouver, Canada. Association for Computational Linguistics.
- Zhilin Yang, Peng Qi, Saizheng Zhang, Yoshua Bengio, William W Cohen, Ruslan Salakhutdinov, and Christopher D Manning. 2018. Hotpotqa: A dataset for diverse, explainable multi-hop question answering. *arXiv preprint arXiv:1809.09600*.
- Jun Yin, Xin Jiang, Zhengdong Lu, Lifeng Shang, Hang Li, and Xiaoming Li. 2016. Neural generative question answering. In *Proceedings of the Twenty-Fifth International Joint Conference on Artificial Intelligence*, pages 2972–2978.
- Rowan Zellers, Yonatan Bisk, Roy Schwartz, and Yejin Choi. 2018. Swag: A large-scale adversarial dataset for grounded commonsense inference. *arXiv preprint arXiv:1808.05326*.



Zhengyan Zhang, Xu Han, Zhiyuan Liu, Xin Jiang, Maosong Sun, and Qun Liu. 2019. [ERNIE: Enhanced language representation with informative entities](#). In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 1441–1451, Florence, Italy. Association for Computational Linguistics.

**A Appendices**

**B Supplemental Material**