Exploring ways to incorporate additional knowledge to improve Natural Language Commonsense Question Answering

Arindam Mitra * and **Pratyay Banerjee*** and **Kuntal Kumar Pal*** and **Swaroop Mishra** * and **Chitta Baral**
Department of Computer Science, Arizona State University
amitra7,pbanerj6,kkpal,srmishr1@asu.edu,chitta@asu.edu

Abstract

DARPA and Allen AI have proposed a collection of datasets to encourage research in Question Answering domains where (commonsense) knowledge is expected to play an important role. Recent language models such as BERT and GPT that have been pre-trained on Wikipedia articles and books, have shown decent performance with little fine-tuning on several such Multiple Choice Question-Answering (MCQ) datasets. Our goal in this work is to develop methods to incorporate additional (commonsense) knowledge into language model based approaches for better question answering in such domains. In this work we first identify external knowledge sources, and show that the performance further improves when a set of facts retrieved through IR is prepended to each MCQ question during both training and test phase. We then explore if the performance can be further improved by providing task specific knowledge in different manners or by employing different strategies for using the available knowledge. We present three different modes of passing knowledge and five different models of using knowledge including the standard BERT MCQ model. We also propose a novel architecture to deal with situations where information to answer the MCQ question is scattered over multiple knowledge sentences. We take 200 predictions from each of our best models and analyze how often the given knowledge is useful, how many times the given knowledge is useful but system failed to use it and some other metrices to see the scope of further improvements.

Introduction

In recent months language models such as GPT (Radford et al. 2018), BERT (Devlin et al. 2019) and their variants (such as RoBERTa (Liu et al. 2019)) that have been pre-trained on Wikipedia articles and books are able to perform very well on many of the natural language question answering tasks. Most often they do better than models specifically designed for specific datasets and these days they form the defacto base line for most new datasets that are proposed. Some times, they even perform at superhuman level, on newly proposed natural language QA datasets (Rajpurkar et al. 2016; Zellers et al. 2018). These models do well even on some of the question answering tasks where question answering seemingly requires knowledge beyond what is given in the QA items. Perhaps it is because some of the needed knowledge that may be present in textual form is "encapsulated" by the language model based systems as they are trained on huge text corpora. But one may wonder whether more can be done; i.e., can the performance be improved by further infusion of the needed knowledge (or a knowledge base containing the needed knowledge), and what are ways of doing such knowledge infusion. Few months back DARPA and Allen AI upped the ante by developing several question answering challenges where commonsense knowledge and reasoning with them is expected to play an important rule. The expected additional challenge in these domains is that often commonsense knowledge is not readily available in textual form. To answer the above mentioned questions we consider three of those QA challenges: Abductive NLI, Physical Interaction QA and Social Interaction QA.

In this paper, we explore ways to infuse knowledge into any language model to reason and solve multiple choice question answering task. Considering a baseline performance of BERT whole-word-masked model, we improve the performance on each of the datasets with three strategies. First, in *revision strategy*, we fine-tune the BERT model on a knowledge-base (KB) which has knowledge statements relevant to that of each of the datasets and then use the model to answer questions. In the second, *Open-Book Strategy*, we choose a certain number of knowledge statements from the KB that are textually similar to each of the samples of the datasets. Then we fine-tune the pre-trained BERT model for the question answering task to choose the answer. In the final strategy, we take the advantage of both the above mentioned strategies. We first fine-tune the pre-trained BERT model on the KB and then use additional knowledge extracted for each sample for the question-answering.

To use the extracted knowledge from the KB, we propose five models, *concat*, *max*, *simple sum*, *weighted sum*, *mac*. Each of the models use knowledge in a different way to choose the correct answer among the options.

Apart from these we created a dataset, Parent and Family QA. The first dataset is intended to test BERT's memoriz-

[*] These authors contributed equally to this work.

Copyright © 2020, Association for the Advancement of Artificial Intelligence (www.aaai.org). All rights reserved.

ing ability for MCQ questions in a controlled environment, while the other is to test BERT's ability for answering MCQ questions with necessary information scattered over multiple knowledge sentences.

Our contribution in this paper are as follows:

- We develop a common library for solving multiple choice questions with external knowledge.
- We propose five novel models representing five ways knowledge can be used with the language models.
- We achieve the state of the art performance on all the three datasets, Abductive NLI, Physical IQA and Social IQA.
- We also synthetically create a dataset, Parent and Family QA, and make them publicly available.

MCQ Datasets

For the study of how to incorporate knowledge, we need datasets which are shown to need external knowledge for question-answering systems to be able to answer. We chose four datasets to evaluate our models, each with a different kind of common sense knowledge. Out of the four, three are made publicly available recently by Allen AI researchers and one is generated synthetically. To incorporate additional knowledge, we choose appropriate knowledge bases that are relevant to each of the datasets. The knowledge paragraphs are retrieved using Information Retrieval and Re-ranking methods.

Datasets

Abductive Natural Language Inference (aNLI) This benchmark dataset (Bhagavatula et al. 2019) is intended to judge potential of an AI system to do abductive reasoning and common sense in order to form possible explanations for a given set of observations. The dataset consists of a total of 169,654 training examples and 1532 validation examples. Given a pair of observations (O_1) and (O_2), the task is to find which of the hypothesis options (H_1) or (H_2) better explains the observations.

Physical Interaction QA This commonsense QA benchmark is created to evaluate the physics reasoning capability of an AI system. The dataset requires reasoning about the use of physical objects and how we use them in our daily life. Given a goal (G) and a pair of choices (C_1) and (C_2), the task is to predict the choice which is most relevant to the goal (G). There are 16,113 training and 1,838 validation samples.

Social Interaction QA The dataset is a collection of instances about reasoning on social interaction and the social implications of their statements. Given a context (C) of a social situation and a question (Q) about the situation, the task is to choose the most appropriate answer options (AO_i) out of three choices. There are several question types in this datasets, which are derived from ATOMIC inference dimensions (Sap et al. 2019b; 2019a). In total, there are 33,410 training and 1,954 validation samples.

Parent and Family QA We synthetically create this dataset to test both, the memorizing capability of neural language models and the ability to combine knowledge spread over multiple sentences. The knowledge retrieved for the three datasets mentioned in the above subsections, may be error prone and in some cases, absent. This is due to the errors from the Information Retrieval step. We create this synthetic dataset to have a better control over the knowledge and ensure we do have the appropriate knowledge to answer the questions.

The source of this dataset is DBPedia (Auer et al. 2007), from which we query for people and extract their parent information. Using this information, we generate 3 kinds of questions, which are, *Who is the parent of X?*, *Who is the grandparent of X?* and *Who is the sibling of X?*. The dataset has a question (Q) and 4 answer options (AO_i). The names of a parent and their family members have many things in common, which can be used to answer such a question. To make the task harder, we remove middle and last names from the answer options. To select wrong answer options, we select those names which are at an edit distance of one or two. This ensures, all the answer options are nearly same, and to actually answer the question, the system needs to have the appropriate knowledge. We also ensure all three kinds of questions for a particular person be present in that particular training or validation set. In total, there are 7,4035 training, 9,256 validation and 9,254 test questions.

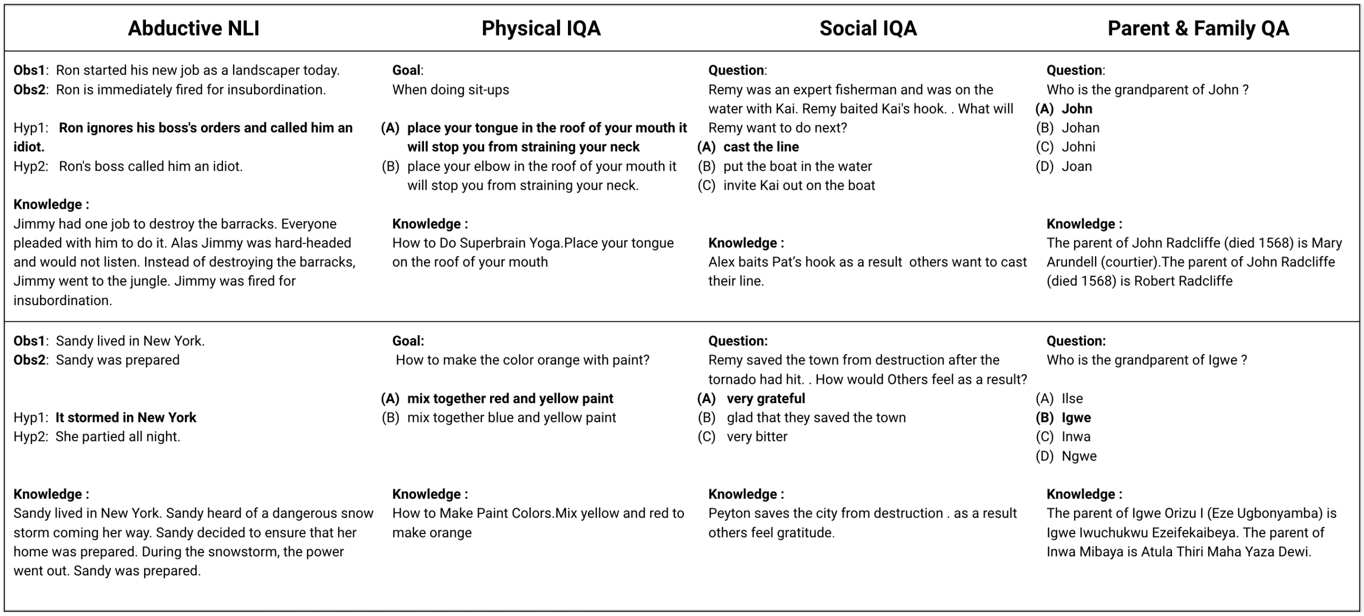
Knowledge Sources

Reasoning with data from each of the above mentioned datasets, needs some commonsense knowledge. We choose four different knowledge bases for each of them.

For aNLI, we retrieve knowledge from the *Story Cloze Test* and *ROCStories Corpora* (Mostafazadeh et al. 2016). Most of the examples in aNLI are based on everyday life stories which depict commonsense relations among daily life activities. Corpora consists of set of five sentence stories about daily life events. These are suitable for the situations present in the aNLI dataset. There are 101903 stories in the entire corpora consisting of ROCStories winter 2017 set, ROCStories spring 2016 set, Story Cloze Test Spring 2016 validation and test set.

Wikihow dataset (Koupaee and Wang 2018) is an ideal commonsense knowledge-base for solving questions of PhysicalIQA dataset. This is a large collection of paragraphs of detailed steps or actions needed to complete a task. The answers of these *How* type questions mostly deals with interactions of humans with physical objects in our surroundings in everyday life. We selected only the titles and headlines from the answers of around 214,544 questions from the dataset and cleaned them to create paragraphs. We ignored the details of each points to reduce the volume of the knowledge.

For Social IQA, we synthetically generate a knowledge-base from the events and inference dimensions provided by the *ATOMIC dataset* (Sap et al. 2019a). The ATOMIC dataset contains events and eight types of if-then inferences. The total number of events are 732,723. Some events are masked, which we fill by using a BERT Large model and



Abductive NLI	Physical IQA	Social IQA	Parent & Family QA
Obs1: Ron started his new job as a landscaper today. **Obs2**: Ron is immediately fired for insubordination. Hyp1: **Ron ignores his boss's orders and called him an idiot.** Hyp2: Ron's boss called him an idiot. **Knowledge :** Jimmy had one job to destroy the barracks. Everyone pleaded with him to do it. Alas Jimmy was hard-headed and would not listen. Instead of destroying the barracks, Jimmy went to the jungle. Jimmy was fired for insubordination.	**Goal**: When doing sit-ups **(A) place your tongue in the roof of your mouth it will stop you from straining your neck** (B) place your elbow in the roof of your mouth it will stop you from straining your neck. **Knowledge :** How to Do Superbrain Yoga.Place your tongue on the roof of your mouth	**Question**: Remy was an expert fisherman and was on the water with Kai. Remy baited Kai's hook. . What will Remy want to do next? **(A) cast the line** (B) put the boat in the water (C) invite Kai out on the boat **Knowledge :** Alex baits Pat's hook as a result others want to cast their line.	**Question**: Who is the grandparent of John ? **(A) John** (B) Johan (C) Johni (D) Joan **Knowledge :** The parent of John Radcliffe (died 1568) is Mary Arundell (courtier).The parent of John Radcliffe (died 1568) is Robert Radcliffe
Obs1: Sandy lived in New York. **Obs2**: Sandy was prepared Hyp1: **It stormed in New York** Hyp2: She partied all night. **Knowledge :** Sandy lived in New York. Sandy heard of a dangerous snow storm coming her way. Sandy decided to ensure that her home was prepared. During the snowstorm, the power went out. Sandy was prepared.	**Goal:** How to make the color orange with paint? **(A) mix together red and yellow paint** (B) mix together blue and yellow paint **Knowledge :** How to Make Paint Colors.Mix yellow and red to make orange	**Question:** Remy saved the town from destruction after the tornado had hit. . How would Others feel as a result? **(A) very grateful** (B) glad that they saved the town (C) very bitter **Knowledge :** Peyton saves the city from destruction . as a result others feel gratitude.	**Question:** Who is the grandparent of Igwe ? (A) Ilse **(B) Igwe** (C) Inwa (D) Ngwe **Knowledge :** The parent of Igwe Orizu I (Eze Ugbonyamba) is Igwe Iwuchukwu Ezeifekaibeya. The parent of Inwa Mibaya is Atula Thiri Maha Yaza Dewi.

Figure 1: Examples of Abductive NLI, Social IQA, Physical IQA and Parent & Family QA datasets with retrieved knowledge

the Masked Language Modelling task (Devlin et al. 2019). We extend the knowledge source, and replace *PersonX* and *PersonY*, as present in the original ATOMIC dataset, using gender neutral names.

For Parent and Family QA, we already possess the gold knowledge sentences. The knowledge for these questions are represented with a simple sentence, *The parent of X is Y.* We do not provide knowledge sentences for questions about *grandparents* and *siblings*. To answer such questions, the systems need to combine information spread over multiple sentences. Nearly all language models are trained over Wikipedia, so all language models would have seen this knowledge.

Relevant Knowledge Extraction

For knowledge retrieval, we use a similar approach as in (Banerjee et al. 2019). We first use an information retrieval model and then re-rank using Information Gain based Reranking. The query is generated using a simple heuristic of unique non-stopwords present in the question, answer option and context if present. For each dataset, we select the top ten knowledge sentences.

Examples of each dataset and their retrieved knowledge from respective KBs are shown in Figure 1.

Standard BERT MCQ Model

After extracting relevant knowledge from the respective KBs, we move onto the task of Question Answering. In all our experiments we use BERT's uncased whole-word-masked model ($BERT_{UWWM}$) (Devlin et al. 2019).

Question Answering Model

As a baseline model, we used pre-trained $BERT_{UWWM}$ for the question answering task with an extra feed-forward layer for classification as a fine-tuning step.

Modes of Knowledge Infusion

We experiment with five different models of using knowledge with the standard BERT architecture for the open-book strategy. Each of these modules take as input a problem instance which contains a question Q, n answer choices $a_1, ..., a_n$ and a list called *premises* of length n. Each element in *premises* contains m number of knowledge passages which might be useful while answering the question Q. Let k_{ij} denotes the j-th knowledge passage for the i-th answer option. Each model computes a score $score(i)$ for each of the n answer choices. The final answer is the answer choice that receives the maximum score. Here, we describe how the different models compute the scores differently.

Concat

In this model, all the m knowledge passages for the $i-th$ choice is joined together to make a single knowledge passage k_i. The sequence of tokens $\{$[CLS] K_i [SEP] Qa_i [SEP]$\}$ is then passed to BERT to pool the [CLS] embedding from the last layer. This way we get n [CLS] embeddings for n answer choices, each of which is projected to a real number ($score(i)$) using a linear layer.

Parallel-Max

For each answer choice a_i, it uses each of the knowledge passage k_{ij} to create the sequence $\{$[CLS] K_{ij} [SEP] Qa_i [SEP]$\}$ which is then passed to the BERT model to obtain the [CLS] embedding from the last layer which is then projected to a real number using a linear layer. $score(i)$ is then taken as the maximum of the m scores obtained using each of the m knowledge passage.

Simple Sum

Unlike the previous model, *simple sum* and the next two models assume that the information is scattered over multiple knowledge passages and try to aggregate those scattered information. To do this, the *simple sum* model, for each answer choice a_i and each of the knowledge passage k_{ij} creates the sequence $\{$[CLS] K_{ij} [SEP] Qa_i [SEP]$\}$ which it then passes to the BERT model to obtain the [CLS] embedding from the last layer. All of these m vectors are then summed to find the summary vector, which then is projected to a scalar using a linear layer to obtain the $score(i)$.

Weighted Sum

The *weighted sum* model unlike the *simple sum* computes a weighted sum of the [CLS] embeddings as some of the knowledge passage might be more useful than others. It computes the [CLS] embeddings in a similar way to that of the *simple sum* model. It computes a scalar weight w_{ij} for each of the m [CLS] embedding using a linear projection layer which we will call as the *weight layer*. The weights are then normalized through a softmax layer and used to compute the weighted sum of the [CLS] embeddings. It then uses (1) a new linear layer or (2) reuses the weight layer (*tied version*) to compute the final score $score(i)$ for the option a_i. We experiment with both of these options.

MAC

The Multi-Sentence Alignment Classification (MAC) model, similar to the *weighted sum* model, computes a weight-sum of the m [CLS] embeddings however with an additional weight-adjustment step. It first obtains a score w_{ij} for a knowledge passage k_{ij} following the *weighted sum* model and normalize them with a softmax. It then reduces the normalized scores further using the following formula:

$$w'_{ij} = w_{ij} - (1 - w_{ij}) * max_{j \neq l \wedge l \in \{1...m\}} \{link\_strength_{ijl}\} \quad (1)$$

Here, $link\_strength_{ijl} \in [0, 1]$ captures how well the two knowledge passage k_{ij} and k_{il} can be "joined" in the sense of joining rows of two tables. Intuitively we want a high *link strength* score between the two knowledge passages "Facebook was launched in Cambridge" and "Cambridge is in MA" but the score should be less for "Facebook was launched in Cambridge" and "Boston is in MA". If two knowledge passage has good *link strength* score then probably they can be joined to infer new information such as "Facebook was launched in MA". The intuition of the weight reduction in equation 1 is that if k_{il} is not strong enough to support the answer choice a_i and it cannot be "joined" with another knowledge passage then probably there is no need to consider it during the final prediction stage. See that if w_{ij} is too close to 1 i.e. if a k_{ij} is very informative, the penalty because of "joinable" or not is negligible. It only becomes prominent when w_{ij} neither too low or too high.

The *link strength* score $link\_strength_{ijl}$ can be computed in different ways. Here we show a memory-efficient way. Since, loading BERT itself takes lot of memory if we create sequences like $\{$[CLS] K_{ij} [SEP] k_{il} [SEP]$\}$ to compute the $link\_strength_{ijl}$ score, it will add a lot of memory overhead and if m is big, it might throw memory exceptions. Here we show how we compute the link strength scores from the BERT outputs of the $\{$[CLS] k_{ij} [SEP] Qa_i [SEP]$\}$ sequences without producing any additional $\{$[CLS] K_{ij} [SEP] k_{il} [SEP]$\}$ sequences. We take the last layer output from the BERT model and use the segment id information (see that segment id for the tokens starting from [CLS] to the first [SEP] token is 0 and is 1 for the remaining tokens) to extract only the token embeddings that belongs to the knowledge passage k_{ij}. Let $h_{ij}^1, ..., h_{ij}^p$ be those token embeddings. We compute a link vector $link_{ij}$ from these token embeddings for the the knowledge passage k_{ij}. The score $link\_strength_{ijl}$ is then computed as follows:

$$link\_strength_{ijl} = \frac{exp(link_{ij}^T\ link_{il})}{\sum_{x \neq j}^{x=1...m} exp(link_{ij}^T\ link_{ix})}$$

To compute the link vector $link_{ij}$ we first pass each token embedding h_{ij}^t through a linear layer which assigns a scalar score s_{ij}^t denoting whether h_{ij}^t should be part of link description $link_{ij}$ or not. The link vector is then calculated as follows:

$$link_{ij} = \sum_{t=1}^{p} s_{ij}^t * h_{ij}^t$$

Related Works

Datasets like SQuAD (Rajpurkar et al. 2016), TriviaQA (Joshi et al. 2017), WikiQA (Yang, Yih, and Meek 2015), CoQA (Reddy, Chen, and Manning 2019) have gained enormous attention over the past few years. Various models have been proposed to solve them. The questions from these datasets are easy to solve since the answers are present in either the passages, contexts or in the options itself.

A more challenging task is, when the multiple choice questions do not have sufficient knowledge to answer correctly given a passage, context or options like ARC (Clark et al. 2018), RACE (Lai et al. 2017), OpenBook QA (Mihaylov et al. 2018). But the language models trained on huge amount of data have been able to solve them quite comfortably.

Our focus in this paper is on datasets which not only requires external facts but also commonsense knowledge to predict the correct options like Abductive NLI (Bhagavatula et al. 2019), Physical IQA (AI 2018) and Social IQA (Sap et al. 2019b).

Experiments

Let D be an MCQ dataset and T be a pre-trained language model, K_D be a knowledge base (a set of paragraphs or sentences) which is useful for D and let K be a general knowledge base where T was pre-trained and K might or might not contain K_D. We took three approaches to infuse knowledge.

Dataset	Strategy	Concat	Max	Sim-Sum	Wtd-Sum	Mac
Abductive NLI	Only Openbook	73.89	73.69	73.50	73.26	73.69
	Only Revision	72.65	NA	NA	NA	NA
	Revision & Openbook	74.35	74.28	74.02	75.13	74.15
Physical IQA	Only Openbook	67.84	72.41	72.58	72.52	75.52
	Only Revision	74.53	NA	NA	NA	NA
	Revision & Openbook	67.74	73.83	76.76	76.82	75.46
Social IQA	Only Openbook	70.22	67.75	70.21	69.96	70.26
	Only Revision	69.45	NA	NA	NA	NA
	Revision & Openbook	68.80	66.56	68.86	69.29	70.01
Parent & Family QA	Only Openbook	91.21	89.8	93.16	91.96	91.15
	Only Revision	78.30	NA	NA	NA	NA
	Revision & Openbook	87.21	91.92	93.32	90.63	91.20

Table 1: Performance of each of the five models (Concat, Max, simple sum, Weighted sum, mac) across four datasets with external knowledge.

Dataset	Model	Dev	Test
Abductive NLI	Baseline	67.36	66.75
	Baseline (Ours)	70.36	NA
	Best Model	75.13	**74.96**
Physical IQA	Baseline	70.89	69.23
	Baseline (Ours)	71.44	NA
	Best Model	75.63	**72.28**
Social IQA	Baseline	66.00	64.50
	Baseline (Ours)	68.86	NA
	Best Model	70.36	**67.53**
Parent & Family QA	Baseline	NA	NA
	Baseline (Ours)	77.85	76.96
	Best Model	93.32	91.24

Table 2: Performance of the best knowledge infused model on the Test set. State-of-the-art models are in bold.

Revision Strategy

In this strategy, T is fine-tuned on K_D with respect to Masked LM and next sentence prediction task and then fine-tuned on the dataset D with respect to the Question Answering task.

Open Book strategy

Here a subset of K_D is assigned to each of the training samples on the dataset D and the model T is fine-tuned on the modified dataset D.

Revision along with an Open Book Strategy

In this strategy, T is fine-tuned on K_D with respect to Masked LM and next sentence prediction task and also a subset of K_D is assigned to each of the training samples on D. The model is then fine-tuned with respect to the modified dataset as a Question Answering task.

Results

Table 1 and Table 2 show summary of our experiments on the four datasets. We can see knowledge helps in improving the performance of neural language models. Both the Open Book and the Revision strategy works, together the performance improves even further. We achieve state of the art performances on aNLI, Social IQA and Physical IQA datasets.

The performance of the Revision strategy is poor for the Social IQA dataset. The reason behind this drop in performance can be attributed to the synthetic nature of the sentences and the unavailability of next sentence prediction task data. This leads to a decrease in the performance of the language model. All the sentences in the KB for Social IQA are single sentence statements, and not paragraphs. The results for Physical IQA and Abductive NLI datasets are better due to the presence of natural and contiguous knowledge sentences.

Discussion and Error Analysis

To understand how knowledge is used and whether the knowledge is useful or not, we do the following analysis: For each of the datasets we have randomly selected 100 samples where our best performing model predicts correctly and 100 samples where it has failed. We identified the following broad categories of analysis.

For the correct predictions, we check, (1) Exact appropriate knowledge is present, (2) A related but relevant knowledge is present, (3) Knowledge is present only in the correct option, and (4) No knowledge is present. Figure 2 shows the counts for the above categories. All the cases do not occur in all the datasets.

For the errors (Figure 3), we analyze, (1) Is the knowledge insufficient, (2) Is the knowledge present in the wrong answer, (3) Knowledge is appropriate but model fails, and (4) Gold label is questionable.

We also analyze given appropriate knowledge, how the model performs. From Figure 2, it can be seen that BERT can answer quite a number of question without knowledge. Also from Figure 3, it is clear that inspite of having good knowledge, BERT fails to answer correctly.

In the following subsections, we analyze the different dataset specific errors.

Social IQA

We measure the performance across the 8 different ATOMIC inference dimensions for the best knowledge infused model. In figure 4 we can see both with and without knowledge

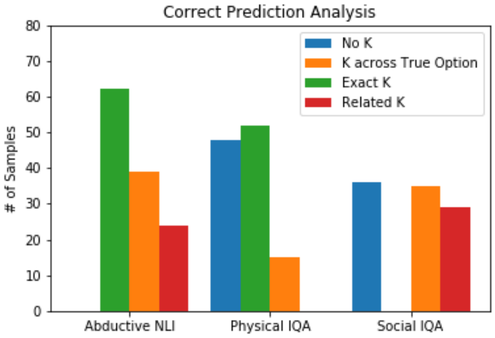


Figure 2: Measure of performance across different knowledge presence in correct predictions

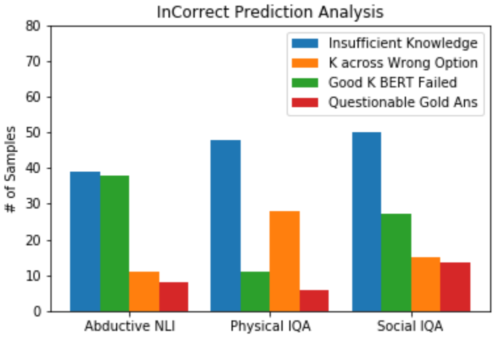


Figure 3: Measure of performance across different knowledge presence in incorrect predictions.

the model performs nearly equally across all dimensions. There is no considerable improvement across any particular dimension.

In some cases the model fails to predict the correct answer inspite of the appropriate knowledge being present.

Question: *Kendall took their dog to the new dog park in the neighborhood. . What will Kendall want to do next?*
(A) *walk the dog* (B) **meet other dog owners**
Knowledge: Jody takes Jody's dog to the dog park, as a result Jody wants to socialize with other dog owners.

In the above example, the above knowledge was retrieved but still the model predicted the wrong option. 341 questions were predicted wrongly after addition of knowledge. We also identified out of the set of 100 analyzed correct predictions, 29% of the questions had partial information relevant to the question.

Parent and Family QA

In Figure 5, we see with addition of knowledge, there is a considerable improvement in performance. Other than questions asking about parents, which just need a look up to answer, the sibling and grandparent questions need models to combine information present across multiple sentences. We can see the model improves even in this questions, showing knowledge infusion helps. Out of the three types of the questions, the performance is lowest on the sibling questions, indicating that it is harder for the models to perform this task. The model accuracy is reasonably good on this dataset, which shows BERT has a strong capability to memorize factual knowledge. Its performance improves with infusion of knowledge,

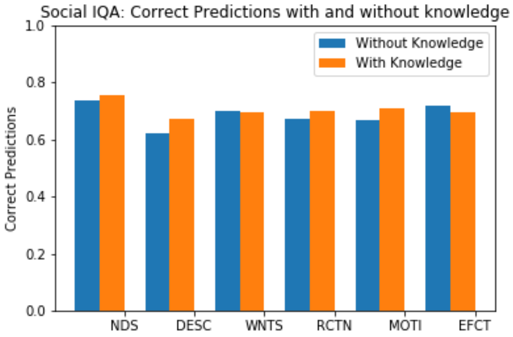


Figure 4: Performance of the model with (MAC model) and without knowledge (Baseline) across different types of ATOMIC inference dimensions.

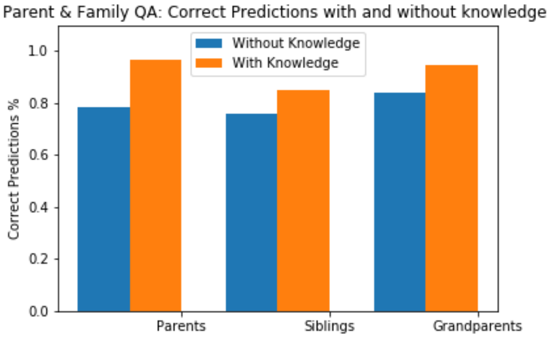


Figure 5: Performance of the model across the three different type of questions.

Here also, 1,790 questions which were previously predicted correctly, are predicted wrong with addition of knowledge.

Physical IQA

Out of the 100 failures that we have analysed, we found that for 8 samples the *goal* matches the knowledge statements but the answers present in the knowledge is different. As for example,

Goal: *How can I soothe my tongue if I burn it?*
(A) Put some salt on it. **(B) Put some sugar on it.**

Knowledge: How to Soothe a Burnt Tongue.Chew a menthol chewing gum.

Also, there are 33 samples in the whole train and dev dataset for which the words in one options are a subset of second option. In those cases, the knowledge retrieved is same for both the options and this confuses the BERT model.

Goal: *What can I drink wine out of if I don't have a wine glass?*
(A) Just pour the wine into a regular mug or glass and drink. **(B) Just pour the wine into a regular mug or wine glass and drink.**
Knowledge: How to Serve Foie Gras. Pour a glass of wine.

On addition of knowledge, 359 samples have become correctly predicted with our best model for Physical IQA dataset which were initially incorrect. But in the process, 166 samples which were correct in our baseline model have now been incorrectly predicted.

Abductive NLI

In this dataset, we also have some examples where negative knowledge is being fed to the model, and it still produces the correct output. There are 8 such examples among the 100 samples we analyzed. For example:

Obs1: *Pablo likes to eat worms.*
Obs2: *Pablo does not enjoy eating worms.*
(Hyp1) Pablo thought that worms were a delicious source of protein. **(Hyp2) Pablo then learned what worms really are.**

Knowledge: Pablo likes to eat worms. He read a book in school on how to do this. He fries them in olive oil. He likes to do this at least once a month. Pablo enjoys worms and views them as a delicacy.

Similarily, we have examples where knowledge favors incorrect hypothesis, however our system still produces correct output. We found 12 such examples among the 100 samples we analyzed. For example:

Obs1: *Dotty was being very grumpy.*
Obs2: *She felt much better afterwards.*
(Hyp1) Dotty ate something bad. **(Hyp2) Dotty call some close friends to chat.**
Knowledge: Allie felt not so good last night. She ate too much. So she had to sleep it off. Then she woke up. She felt so much better

We have 12 cases among 100 analyzed samples, where both hypothesis are very similar. So,our system is unable to produce correct output. For example:

Obs1: *Bob's parents grounded him.*
Obs2: *He came back home but his parents didn't even know he left.*
(Hyp1) Bob got caught sneaking out. **(Hyp2) Bob got away with sneaking out.**

We also have 34 examples where incorrect hypothesis has more word similarity with the observation and knowledge, whereas correct hypothesis has been paraphrased or has less word similarity. The system predicts the wrong answer in such a situation. One such example is:

Obs1: *Mary's mom came home with more bananas than they could possibly eat.*
Obs2: *That was the best way ever to eat a banana!*
(Hyp1) Mary and her mom decided to make chocolate covered frozen bananas to avoid waste.
(Hyp2) Mary made pineapple splits for everyone.
Knowledge: Mary s mom came home with more bananas than they could possibly eat. She wondered why she had bought them all. Then after dinner that night she got a surprise. Mom made banana splits for the whole family. That was the best way ever to eat a banana

Another area where the system fails, is where the problem seems to be open-ended, and many hypotheses can explain the pair of observations. It is tough to find exact knowledge in such a scenario. For example,

Obs1: *Lisa went for her routine bike ride.*
Obs2: *Some days turn out to be great adventures.*
(Hyp1) Lisa spotted a cat and followed it off trail
(Hyp2) Lisa saw a lot of great food.
Knowledge: Lisa went for her routine bike ride.Only this time she noticed an abandoned house.She stopped to look in the house.It was full of amazing old antiques.Some days turn out to be great adventures.

Conclusion

In this work, we have evaluated different ways to incorporate knowledge into language models. We have pushed the current state of the art of the three commonsense knowledge tasks. We have provided five new models for multiple choice natural language QA using knowledge and analyzed their performance on these commonsense datasets. We also make a synthetic dataset available which measures the memorizing and reasoning ability of language models.

We observe that, existing knowledge bases even though do not contain all the knowledge that is needed to answer the questions, they do provide a significant amount of knowledge. BERT, even though utilizes some of the knowledge, there are areas where model can be further improved, particularly the ones where the knowledge is present but the model could not answer, and where it predicted wrong answers with irrelevant knowledge. Our future work is to analyze the source of this errors and try to explore possible solutions.

References

AI, A. 2018. Physical iqa. *URL https://allenai.org/*.

Auer, S.; Bizer, C.; Kobilarov, G.; Lehmann, J.; Cyganiak, R.; and Ives, Z. 2007. Dbpedia: A nucleus for a web of open data. In *Proceedings of the 6th International The Semantic Web and 2Nd Asian Conference on Asian Semantic Web Conference*, ISWC'07/ASWC'07, 722–735. Berlin, Heidelberg: Springer-Verlag.

Banerjee, P.; Pal, K. K.; Mitra, A.; and Baral, C. 2019. Careful selection of knowledge to solve open book question answering. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, 6120–6129. Florence, Italy: Association for Computational Linguistics.

Bhagavatula, C.; Bras, R. L.; Malaviya, C.; Sakaguchi, K.; Holtzman, A.; Rashkin, H.; Downey, D.; Yih, S. W.-t.; and Choi, Y. 2019. Abductive commonsense reasoning. *arXiv preprint arXiv:1908.05739*.

Clark, P.; Cowhey, I.; Etzioni, O.; Khot, T.; Sabharwal, A.; Schoenick, C.; and Tafjord, O. 2018. Think you have solved question answering? try arc, the ai2 reasoning challenge. *arXiv preprint arXiv:1803.05457*.

Devlin, J.; Chang, M.-W.; Lee, K.; and Toutanova, K. 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, Volume 1 (Long and Short Papers)*, 4171–4186. Minneapolis, Minnesota: Association for Computational Linguistics.

Joshi, M.; Choi, E.; Weld, D. S.; and Zettlemoyer, L. 2017. Triviaqa: A large scale distantly supervised challenge dataset for reading comprehension. *arXiv preprint arXiv:1705.03551*.

Koupaee, M., and Wang, W. Y. 2018. Wikihow: A large scale text summarization dataset. *arXiv preprint arXiv:1810.09305*.

Lai, G.; Xie, Q.; Liu, H.; Yang, Y.; and Hovy, E. 2017. Race: Large-scale reading comprehension dataset from examinations. In *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing*, 785–794.

Liu, Y.; Ott, M.; Goyal, N.; Du, J.; Joshi, M.; Chen, D.; Levy, O.; Lewis, M.; Zettlemoyer, L.; and Stoyanov, V. 2019. Roberta: A robustly optimized BERT pretraining approach. *CoRR* abs/1907.11692.

Mihaylov, T.; Clark, P.; Khot, T.; and Sabharwal, A. 2018. Can a suit of armor conduct electricity? a new dataset for open book question answering. In *EMNLP*.

Mostafazadeh, N.; Chambers, N.; He, X.; Parikh, D.; Batra, D.; Vanderwende, L.; Kohli, P.; and Allen, J. 2016. A corpus and evaluation framework for deeper understanding of commonsense stories. *arXiv preprint arXiv:1604.01696*.

Radford, A.; Narasimhan, K.; Salimans, T.; and Sutskever, I. 2018. Improving language understanding by generative pre-training. *URL https://s3-us-west-2. amazonaws. com/openai-assets/researchcovers/languageunsupervised/language understanding paper. pdf*.

Rajpurkar, P.; Zhang, J.; Lopyrev, K.; and Liang, P. 2016. Squad: 100,000+ questions for machine comprehension of text. In *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing*, 2383–2392.

Reddy, S.; Chen, D.; and Manning, C. D. 2019. Coqa: A conversational question answering challenge. *Transactions of the Association for Computational Linguistics* 7:249–266.

Sap, M.; Le Bras, R.; Allaway, E.; Bhagavatula, C.; Lourie, N.; Rashkin, H.; Roof, B.; Smith, N. A.; and Choi, Y. 2019a. Atomic: an atlas of machine commonsense for if-then reasoning. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 33, 3027–3035.

Sap, M.; Rashkin, H.; Chen, D.; LeBras, R.; and Choi, Y. 2019b. Socialiqa: Commonsense reasoning about social interactions. *arXiv preprint arXiv:1904.09728*.

Yang, Y.; Yih, W.-t.; and Meek, C. 2015. Wikiqa: A challenge dataset for open-domain question answering. In *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing*, 2013–2018.

Zellers, R.; Bisk, Y.; Schwartz, R.; and Choi, Y. 2018. Swag: A large-scale adversarial dataset for grounded commonsense inference. *arXiv preprint arXiv:1808.05326*.