A Model and Implementation Details

A detailed visualization of our model, described in Section 2.2 of the main paper is shown in Fig. 1.

1.1 Knowledge Encoding

We describe the fact encoding and provide comprehensive visualization of the Bi-directional GRU execution on Figure 2. For each instance in the dataset, we retrieve a number of relevant facts. Each retrieved fact is represented as a triple $f = (w_{1..L_{subj}}, w^{rel}_{0}, w_{1..L_{obj}})$, where $w_{1..L_{subj}}$ and $w_{1..L_{obj}}$ are multi-word expressions representing the subject and object with sequence lengths $L_{subj}$ and $L_{obj}$, and $w^{rel}_{0}$ is a word token corresponding to a relation.$^1$ As a result of fact encoding, we obtain a separate knowledge memory for each instance in the data.

To encode the knowledge we use a BiGRU to encode the triple argument tokens into the following context-encoded representations:

$$f_{last}^{subj} = \text{BiGRU}(\text{Emb}(w^{subj}_{1..L_{subj}}), 0)$$
$$f_{last}^{rel} = \text{BiGRU}(\text{Emb}(w^{rel}_{0}), f_{last}^{subj})$$
$$f_{last}^{obj} = \text{BiGRU}(\text{Emb}(w^{obj}_{1..L_{obj}}), f_{last}^{rel})$$

where $f_{last}^{subj}$, $f_{last}^{rel}$, $f_{last}^{obj}$ are the final hidden states of the context encoder BiGRU, that are also used as initial representations for the encoding of the next triple attribute in left-to-right order. The motivation behind this encoding is: (i) We encode the knowledge fact attributes in the same vector space as the plain tokens; (ii) we preserve the triple directionality; (iii) we use the relation type as a way of filtering the subject information to initialize the object.

1.2 Model Implementation Parameters

We implement our model in TensorFlow 0.12 (Abadi et al., 2015). Below we report preprocessing steps and hyper-parameters required for reproducing the model.

Dataset. We perform experiments on the Common Nouns and Named Entities parts of the Children’s Book Test (CBT) (Hill et al., 2015).$^2$

Pre-processing. For each instance of the dataset (21 sentences, 20 for the story and 1 for question), we remove the line number, which is originally presented in the text as a first token of the sentence and split the tokens using `str.split()` in Python 2.7. We then concatenate the tokens for the sentences in the story into a single list of story tokens $d_{1..m}$.

Knowledge Source. We use knowledge from the Open Mind Common Sense (OMCS, Singh et al. (2002)) part of ConceptNet (Speer et al., 2017), a crowd-sourced resource of commonsense knowledge with a total of $\sim 630k$ facts.$^3$ The exact knowledge splits required for our experiments will be available in json format.$^4$

Vocabulary. To build the vocabulary we select the words that occur at least 5 times in the training set. We extend the vocabulary with all words retrieved from the knowledge source. All words are lowercased. Following Kadlec et al. (2016) we use multiple unknown tokens (UNK$_1$, UNK$_2$, . . . , UNK$_{100}$). In each example, for each unknown word, we pick randomly an unknown token from

$^1$The 0 in $w^{rel}_{0}$ indicates that we encode the relation as a single relation type word. Ex. /r/IsUsedFor.

$^2$The dataset can be downloaded from: http://www.thespermwhale.com/jaseweston/babi/CBTest.tgz

$^3$ConceptNet 5 github page: https://github.com/commonsense/conceptnet5

Figure 1: The Knowledgeable Reader combines plain context & enhanced (context + knowledge) repres. of $D$ and $Q$ and retrieved knowledge from the explicit memory with the Key-Value approach.

Figure 2: Encoding the knowledge triple using BiGRU.

Word Embeddings. We use Glove 100D\textsuperscript{5} word embeddings pre-trained on 6B tokens from Wikipedia and Gigaword5. We initialize the out-of-vocabulary words by sampling from a uniform distribution in range $[-0.1, 0.1]$. We optimize all word embeddings in the first 8000 training steps.

Encoder Hidden Size. We use a hidden size of 256 for the GRU encoder states (512 output for our bi-directional encoding). This setting has been shown to perform well for the Attention Sum Reader (Kadlec et al., 2016).

Batching, Learning rate, Sampling. We sort the data examples in the training set by document length and create batches with 64 examples. For each training step we pick batches randomly. After every 1000 training steps we evaluate the models on the validation Dev set. We train for 60 epochs and pick the model with the highest validation accuracy to make the predictions for Test.

Optimization. We use cross entropy loss on the predicted scores for each answer candidate. We use Adam (Kingma and Ba, 2015) optimizer with

\[ f_{\text{sub}} = \{ \text{GRU}^{fw}(\text{strong, horse}), \text{GRU}^{bw}(\text{horse, strong}) \} \]
\[ f_{\text{rel}} = \{ \text{GRU}^{fw}(\text{strong, horse, UsedFor}), \text{GRU}^{bw}(\text{horse, strong, UsedFor}) \} \]
\[ f_{\text{obj}} = \{ \text{GRU}^{fw}(\text{strong, horse, UsedFor, ride, fast}), \text{GRU}^{bw}(\text{horse, strong, UsedFor, ride, fast}) \} \]
initial learning rate of 0.001 and clip the gradients in the range $[-10, 10]$.

B Quantitative Analysis

1.3 Additional Ablation Experiments

Due to space limitation in the main paper, we present additional results here. In addition to ablation of model components for 50 facts, we perform experiments for 100 as well. The results are shown in Table 1. The results show a similar tendency, but in this setting, omitting the model without knowledge enrichment yields best results for the CN data.

1.4 Results for Ensemble Models

For each dataset we combine our best 11 runs and use majority voting to predict the answer for our Ensemble model.

In Table 2 we show the comparison with multi-hop models. We report Accuracy on the Dev and Test sets, rounded to the first decimal point as done in previous work. The AoA Reader (Cui et al., 2017) uses re-ranking as a post-processing step and the other neural models are not directly comparable.

C Manual Analysis and Visualization

Case 1 We provide an extended illustration of the example discussed in the main paper in Figure 3. We manually inspect examples from the evaluation sets where KnReader improves prediction or makes the prediction worse. Figure 3 shows the question with placeholder, followed by answer candidates and their associated attention weights as assigned by the model w/o knowledge. The matrix shows selected facts and their learned weights representation (w/o knowledge) $Q_{ctx}$ to $D_{ctx + kn}$ interaction with candidate head.

Case 2 Figure 4 shows another interesting example. The document is part of the The kings new clothes by Hans Christian Andersen. While, given the story, many of the choices are plausible (cloth, clothes, nothing, air, cloak) the model without knowledge selects cloth as the most probable answer. Adding the knowledge facts reverts the answer. We can speculate that the reason is the fact clothes /r/Antonym undressed retrieved by the answer candidate token clothes which has multiple occurrences in the text, and since the updated representation combines well with the phrase put on which is antonym to undressed clothes /r/Antonym undressed and clothes /r/Antonym naked. A rea-
Story: ... ' what has a bird, in spite of all his singing, in the winter-time? he must starve and freeze, and that must be very pleasant for him, i must say! '

Figure 3: **Case 1**: Interpreting the components of KnReader (Full model). Adding retrieved knowledge to Q and D helps the model to increase the score for the correct answer. Results for top 5 candidates are shown. (Subj/Obj as key-value memory, 50 facts, CN5Sel) (Item #357)

son for this could also be the high frequency of clothes in the story. However, the example cannot be answered using the story context alone, as it talks about the imaginary, not existing (air, nothing) new clothes of the king.

The example also shows what kind of knowledge is missing in our currently used resources: ideally, the question can be answered using information from the question alone, by analyzing the meaning of the phrases *take off your clothes* and *then we will put on the new XXXX*. If they were available, the model could exploit the knowledge that *taking off (clothes)* and *putting on (clothes)* are actions often performed in temporal sequence.

**Case 3** In Figure 5 we have an example where the model overcomes the frequency bias of the story (*magician* occurs 4 times) to select a more plausible example (*father*) using the fact *father/r/Antonym son*.

**Case 4** Figure 6 shows an example where a correct initial prediction obtained without knowl-
**Story:** ... they pretended they were taking the cloth from the loom, cut with huge scissors in the air, sewed with needles without thread, and then said at last, 'now the clothes are finished!' the emperor came himself with his most distinguished knights, and each impostor held up his arm just as if he were holding something, and said, 'see! here are the breeches! here is the coat! here the cloak!' and so on.

‘spun clothes are so comfortable that one would imagine one had nothing on at all; but that is the beauty of it!’ ‘yes,’ said all the knights, but they could see nothing, for there was nothing there.

Figure 4: **Case 2:** Interpreting the components of KnReader (Full model). Adding retrieved knowledge to Q and D helps the model to increase the score for the correct answer. Results for top 5 candidates are shown. (Subj/Obj as key-value memory, 50 facts, CN5Sel) (Item #52)

edge is reversed and a clearly wrong answer is selected instead. Although a relevant fact is selected (people /r/UsedFor help you), apparently, the model misses the information that brothers are people and can’t combine the acquired concept help you with the question context and with their help dragged ..., and thus, the correct answer is not sufficiently promoted.

**Case 5** The example in Figure 7 illustrates the lack of knowledge about locations. The context of
Story: .. a celebrated magician, who had given the seed to my father, promised him that they would grow into the three finest trees the world had ever seen.

... after this I had the beautiful fruit of these trees carefully guarded by my most faithful servants; but every year, on this very night, the fruit was plucked and stolen by an invisible hand, and next morning not a single apple remained on the trees.

For some time past I have given up even having the trees watched.

Figure 5: Case 3: Interpreting the components of KnReader (Full model). Adding retrieved knowledge to Q and D helps the model to increase the score for the correct answer. Results for top 5 candidates are shown. (Subj/Obj as key-value memory, 50 facts, CN5Sel) (Item #240)

Q talks about climbing up and while the text-only module selects the right answer cliff, the available knowledge modifies the representation and reverses the answer to sea which is usually on lower level. Here the association is made with a cliff and sea by the fact inlet /r/PartOf sea and beach /r/PartOf shore. That is, the context-only neural representation guesses that the plausible answer is similar to cliff (inlet and shores are usually associated with cliff). Again, we are missing knowledge of actions, e.g., that climbing is done to move up steep locations such as hills, or cliffs. In future work we plan to experiment with sources that offer more information about events.
Story: ... in the same village there lived three brothers, who were all determined to kill the mischievous hawk. ... ... his eyelids closed, and his head sank on his shoulders, but the thorns ran into him and were so painful that he awoke at once. The hawk fell heavily under a big stone, severely wounded in its right wing.

The youth ran to look at it, and saw that a huge abyss had opened below the stone.

Figure 6: Case 4: Interpreting the components of KnReader (Full model). Adding retrieved knowledge to Q and D confuses the model and decreases the score for the correct answer. Results for top 5 candidates are shown. (Subj/Obj as key-value memory, 50 facts, CN5Sel) (Item #172)

References


Yiming Cui, Zhiheng Chen, Si Wei, Shijin Wang, Ting Liu, and Guoping Hu. 2017. Attention-over-
Story: ... i also lay this belt beside you , to put on when you awaken ; it will keep you from growing faint with hunger.

the woman now disappeared , and unk. 98 woke , and saw that all her dream had been true.

the rope hung down from the cliff , and the clew and belt lay beside her.

dog

Figure 7: Case 5: Interpreting the components of KnReader (Full model). Adding retrieved knowledge to Q and D confuses the model and decreases the score for the correct answer. Results for top 5 candidates are shown. (Subj/Obj as key-value memory, 50 facts, CN5Sel) (Item #187)


