World Knowledge for Reading Comprehension: Rare Entity Prediction with Hierarchical LSTMs Using External Descriptions

Teng Long, Emmanuel Bengio, Ryan Lowe
Jackie Chi Kit Cheung, Doina Precup
{teng.long,emmanuel.bengio,ryan.lowe}@mail.mcgill.ca
{jcheung,dprecup}@cs.mcgill.ca
School of Computer Science
McGill University

Abstract

Humans interpret texts with respect to some background information, or *world knowledge*, and we would like to develop automatic reading comprehension systems that can do the same. In this paper, we introduce a task and several models to drive progress towards this goal. In particular, we propose the task of *rare entity prediction*: given a web document with several entities removed, models are tasked with predicting the correct missing entities conditioned on the document context and the lexical resources. This task is challenging due to the diversity of language styles and the extremely large number of rare entities. We propose two recurrent neural network architectures which make use of external knowledge in the form of entity descriptions. Our experiments show that our hierarchical LSTM model performs significantly better at the rare entity prediction task than those that do not make use of external resources.

1 Introduction

Reading comprehension is the ability to process some text and understand its contents, in order to form some beliefs about the world. The starting point of this paper is the fact that world knowledge plays a crucial role in human reading comprehension and language understanding. Work in the psychology of reading literature has demonstrated this point, for example by showing that readers are better able to recall the contents of a story when it describes a counter-intuitive but plausible sequence of events, rather than a bizarre or a highly predictable one (Barrett and Nyhof, 2001). This point is also central to work in the Schankian tradition of scripts (Schank and Abelson, 1977).

Despite the importance of world knowledge, previous data sets and tasks for reading comprehension have targeted other aspects of the reading comprehension problem, at times explicitly attempting to factor out its influence. In the Daily Mail/CNN dataset (Hermann et al., 2015), named entities such *Clarkson* and *Top Gear* are replaced by anonymized entity tokens like *ent212*. The Children's Book Test focuses on the role of context and memory (Hill et al., 2016a), and the fictional genre makes it difficult to connect the entities in the stories to real-world knowledge about those entities.

As a result, language models have proved to be a highly competitive solution to these tasks. Chen et al. (2016) showed that their attention-based LSTM model achieves state-of-the-art results on the Daily Mail/CNN data set. In fact, their analysis shows that more than half of the questions can be answered by exact word matching and sentence-level paraphrase detection, and that many of the remaining errors are difficult to solve exactly because the entity anonymization procedure removes necessary world knowledge.

In this paper, we propose a novel task called *rare entity prediction*, which places the use of external knowledge at its core, with the following key features. First, our task is similar in flavour to the Children's Book and other language modeling tasks, in that the goal of the models is to predict missing elements in text. However, our task involves predicting missing named entities, rather than missing words. Second, the number of unique named entities in the data set is very large, roughly on par with the number of documents. As such, there are very few instances per named entity for systems to train on. Instead, they must rely on external knowledge sources such as Freebase (Bollacker et al., 2008) in order to make inferences

Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing, pages 825–834
Copenhagen, Denmark, September 7–11, 2017.

Context
[...] _____, who lived from 1757 to 1827, was admired by a small group of intellectuals and artists in his day, but never gained general recognition as either a poet or painter. [...]
Candidate Entities
Peter Ackroyd: Peter Ackroyd is an English biographer, novelist and critic with a particular interest in the history and culture of London. [...]
William Blake: William Blake was an English poet, painter, and printmaker. [...]
Emanuel Swedenborg: Emanuel Swedenborg was a Swedish scientist, philosopher, theologian, revelator, and mystic. [...]

Table 1: An abbreviated example from the Wikilinks Rare Entity Prediction dataset. Shown is an excerpt from the text (context), with a missing entity that must be predicted from a list of candidate entities. Each candidate entity is also provided with its description from Freebase.

about the likely entities that fit the context.

For our task, we use a significantly enhanced version of the Wikilinks dataset (Singh et al., 2012), with entity descriptions extracted from Freebase serving as the lexical resources, which we call the *Wikilinks Rare Entity Prediction* dataset. An example from the Wikilinks Entity Prediction dataset is shown in Table 1.

We also introduce several recurrent neural network-based models for this task which take in entity descriptions of candidate entities. Our first model, DOUBENC, combines information derived from two encoders: one for the text passage being read, and one for the entity description. Our second model, HIERENC, is an extension which considers information from a document-level context, in addition to the local sentential context. We show that language modeling baselines that do not consider entity descriptions are unable to achieve good performance on the task. RNN-based models that are trained to leverage external knowledge perform much better; in particular, HIERENC achieves a 17% increase in accuracy over the language model baseline.

2 Related Work

Related to our work is the task of *entity prediction*, also called *link prediction* or *knowledge base completion*, in the context of multi-relational data. Multi-relational datasets like WordNet (Miller, 1995) and Freebase (Bollacker et al., 2008) consist of entity-relation *triples* of the form (head, relation, tail). In entity prediction, either the head or tail entity is removed, and the model has to predict the missing entity. Recent efforts have integrated different sources of knowledge, for example combining distributional and relational semantics for building word embeddings (Fried and Duh, 2015; Long et al., 2016). While this task requires understanding and predicting associations between entities, it does not require contextual reasoning with text passages, which is crucial in rare entity prediction.

Rare entity prediction is also clearly distinct from tasks such as entity tagging and recognition (Ritter et al., 2011), as models are provided with the actual name of the entity in question, and only have to match the entity with related concepts and tags. It is more closely related to the machine reading literature from e.g. Etzioni et al. (2006); however, the authors define machine reading as primarily unsupervised, whereas our task is supervised.

A similar supervised reading comprehension task was proposed by Hermann et al. (2015) using news articles from CNN and the Daily Mail. Given an article, models are tasked with filling in blanks of one-sentence summaries of the article. The original dataset was found to have a low ceiling for machine improvement (Chen et al., 2016); thus, alternative datasets have been proposed that consist of more difficult questions (Trischler et al., 2016; Rajpurkar et al., 2016). A dataset with a similar task was also proposed by Hill et al. (2016a), where models must answer questions about short children's stories. While these tasks require the understanding of unstructured natural language, they do not require integration with external knowledge sources.

Hill et al. (2016b) proposed a method of combining distributional semantics with an external knowledge source in the form of dictionary definitions. The purpose of their model is to obtain more accurate word and phrase embeddings by combining lexical and phrasal semantics, and they achieve fairly good performance on reverse dictionaries and crossword puzzle solving tasks.

Perhaps the most related approach to our work is the one developed by Ahn et al. (2016). The authors propose a WikiFacts dataset where Wikipedia descriptions are aligned with Freebase

facts. While they also aim to integrate external knowledge with unstructured natural language, their task differs from ours in that it is primarily a language modeling problem.

More recently, Bahdanau et al. (2017) investigated a similar approach to generate embeddings for out-of-vocabulary words from their definitions and applied it to a number of different tasks. However, their method mainly focuses on modeling generic concepts and is evaluated on tasks that do not require the understanding of world knowledge specifically. Our work, on the other hand, shows the effectiveness of incorporating external descriptions for modeling real-world named entities and is evaluated on a task that explicitly requires the understanding of such external knowledge.

3 Rare Entity Prediction

3.1 The *Wikilinks* Dataset

The *Wikilinks* dataset (Singh et al., 2012) is a large dataset originally designed for cross-document coreference resolution, the task of grouping entity mentions from a set of documents into clusters that represent a single entity. The dataset consists of a list of non-Wikipedia web pages (discovered using the Google search index) that contain hyperlinks to Wikipedia, such as random blog posts or news articles. Every token with a hyperlink to Wikipedia is then marked and considered an entity mention in the dataset. Each entity mention is also linked back to a knowledge base through their corresponding Freebase IDs

In order to ensure the hyperlinks refer to the correct Wikipedia pages, additional filtering is conducted to ensure that either (1) at least one token in the hyperlink (or *anchor*) matches a token in the title of the Wikipedia page, or (2) the anchor text matches exactly an anchor from the Wikipedia page text, which can be considered an alias of the page. As many near-duplicate copies of Wikipedia pages can be found online, any web pages where more than 70% of the sentences match those from their linked Wikipedia pages are discarded.

3.2 The *Wikilinks Rare Entity Prediction* Dataset

We use a significantly pre-processed and augmented version of the *Wikilinks* dataset for the purpose of entity prediction, which we call the *Wikilinks Rare Entity Prediction* dataset. In particular, we parse the HTML texts of the web pages and extract their page contents to form our corpus. Entity mentions with hyperlinks to Wikipedia are marked and replaced by a special token (**blank**), serving as the placeholder for missing entities that we would like the models to predict. The correct missing entity $\tilde{e}$ is preserved as a target. Additionally, we extract the lexical definitions of all entities that are marked in the corpus from Freebase using their Freebase IDs, which are available for all entities in the *Wikilinks* dataset. These lexical definitions will serve as the external knowledge to our models.

Number of documents	269,469
Average # blanks *per* doc	3.69
Average # candidates *per* doc	3.35
Number of unique entities	245,116
# entities with $n <= 5$	207,435 (**84.6%**)
# entities with $n <= 10$	227,481 (**92.8%**)
# entities with $n <= 20$	238,025 (**97.1%**)

Table 2: Statistics for the augmented version of the *Wikilinks* dataset, where n represents the entity frequency in the corpus. Web documents with more than 10 blanks to fill are filtered out for computational reasons.

Table 2 shows some basic statistics of a subset of the corpus used in our experiments. As we can see, unlike the Children's Book dataset, which has 50k candidate entities for almost 700k context and query pairs (Hill et al., 2016a), the number of unique entities found in our dataset has the same order of magnitude as the number of documents available.

Moreover, the majority of entities appears a relatively small number of times, with 92.8% observed less than or equal to 10 times across the entire corpus. This suggests that models that only rely on the surrounding context information may not be able to correctly predict the missing entities. This further motivates us to incorporate additional information into the decision process to improve the performance. In the experiments section, we show that the external entity descriptions are indeed necessary to achieve better results.

3.3 Task Definition[1]

Here we formalize the task definition of the entity prediction problem. Given a document $\mathcal{D}$ in

[1]On notation: we use $\mathbf{A}$ to denote sequences, $\mathcal{A}$ to denote sets, a to denote words / entities, $\mathbf{a}$ to denote vectors, A to denote matrices.

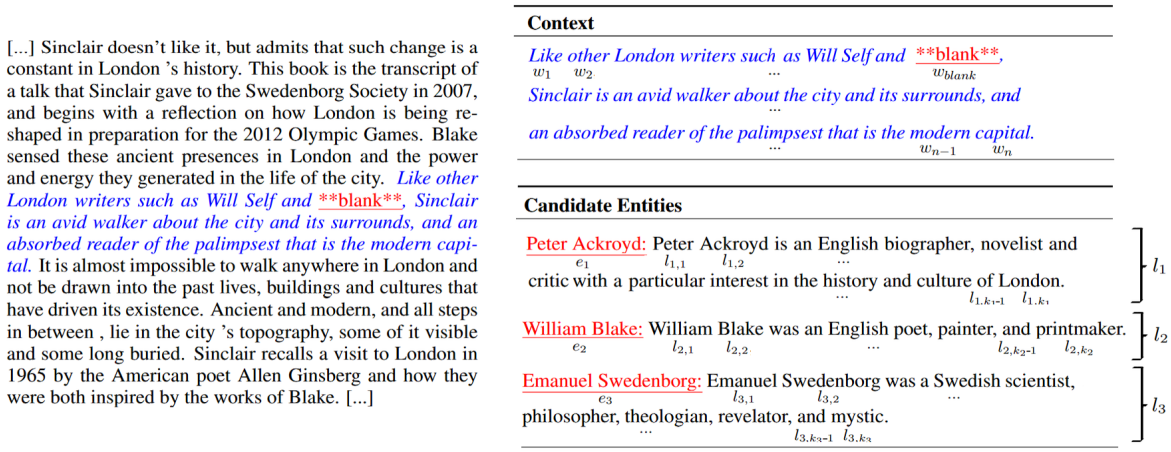

Figure 1: An example from the *Wikilinks Rare Entity Prediction* dataset. Shown is a paragraph from the dataset, along with the context (in blue italics) and the missing entity (in red underline). We also visually show the notation that we use for the remainder of this paper. The correct answer here is Peter Ackroyd.

the corpus, we split it into an ordered list of contexts $\mathcal{C} = \{\mathsf{C}_1, ..., \mathsf{C}_n\}$ where each context C_i $(1 \leq i \leq n)$ is a word sequence $(w_1, ..., w_m)$ where the special token **blank** is found. Let $\mathcal{E}$ be the set of candidate entities. For each missing entity, we want the model to select the correct entity $\tilde{e} \in \mathcal{E}$ to fill the blank slot. In our problem setting, the model also has access to the lexical resource $\mathcal{L} = \{\mathsf{L}_e \mid e \in \mathcal{E}\}$ where $\mathsf{L}_e = (l_{e_1}, ..., l_{e_k})$ is the lexical definition of entity e extracted from the knowledge base. Thus, the task of the model is to predict the correct missing entities for each empty slot in $\mathcal{D}$.

There are several possible ways to specify the candidate set $\mathcal{E}$. For instance, we could define $\mathcal{E}$ so that it includes all entities found in the corpus. However, given the extremely large amount of unique entities found in the dataset, this would render the task difficult to solve from both a practical and computational perspective. We present a simpler version of the task where $\mathcal{E}$ is the set of entities that are present in the document $\mathcal{D}$. Note that we can make the task arbitrarily more difficult by randomly sampling other entities from the entity vocabulary and adding them to candidate set.

We show an example from the *Wikilinks Entity Prediction* dataset in Figure 1, along with a visual guide to the notation from this section.

4 Model Architectures

In this section, we present two models that use the lexical definitions of entities to solve the proposed rare entity prediction problem. The basic building blocks of our models are recurrent neural networks (RNN) with long short-term memory (LSTM) units (Hochreiter and Schmidhuber, 1997). An RNN is a neural network with feedback connections that allows information from the past to be encoded in the hidden layer representation, thus is ideal for modeling sequential data (Dietterich, 2002) and most language related problems.

LSTMs are an extension of RNNs which include a memory cell $\mathbf{c}_t$ alongside their hidden state representation $\mathbf{h}_t$. Reads and writes to the memory cell are controlled by a set of three gates that allow the model to either keep or discard information from the past and update their state with the current input. This allows LSTMs to model potentially longer dependencies and at the same time mitigate the vanishing and exploding gradient problems, which are quite common among regular RNNs (Bengio et al., 1994). In our experiments, we use LSTMs augmented with peephole connections (Gers et al., 2002).

We denote the output (i.e. the last hidden state) of an RNN f operating on a sequence S as $f(\mathsf{S})$, and subscript the t-th hidden state as $f_t(\mathsf{S})$.

4.1 Double Encoder (DOUBENC)

This model uses two jointly trained recurrent models, a lexical encoder $g(.)$ and a context encoder $f(.)$, and a logistic predictor P (see Figure 2).

The lexical encoder converts the definition of an entity into a vector embedding, while the context encoder repeats the same process for a given context to obtain its context embedding. These two embeddings are then used by P to predict if the

last state representation

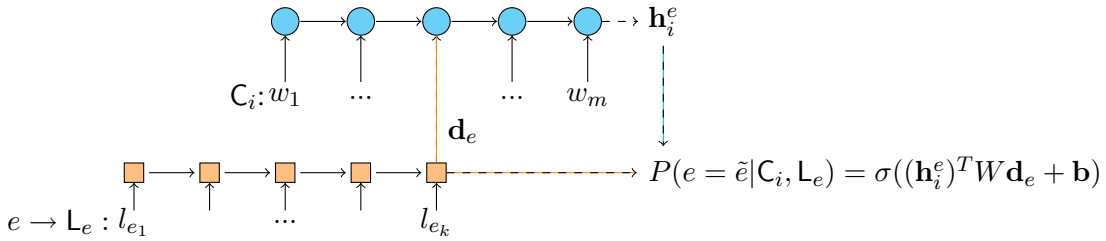

Figure 2: Our double encoder architecture. Each entity e has an associated lexical definition $\mathsf{L}_e = (l_{e_1}, l_{e_2}, ..., l_{e_k})$, which is fed through the lexical encoder g (orange squares) to provide an encoding $\mathbf{d}_e$. This definition embedding is then fed as the *blank* input token of context C_i to the context encoder f (blue circles), which provides a context embedding $\mathbf{h}_i^e$.

given entity-context pair is correct. Additionally, the blank in the context sentence is replaced by the encoded definition embedding to provide more information to f.

For an entity e in the candidate set $\mathcal{E}$ of document $\mathcal{D}$, we retrieve its corresponding lexical definition L_e, itself a sequence of words, to compute its encoding $g(\mathsf{L}_e) \equiv \mathbf{d}_e$.

For a given context C_i, we replace the embedding of the blank token with $\mathbf{d}_e$. Thus $\mathsf{C}_i = (w_1, ..., w_{blank}, ..., w_m)$ becomes $\mathsf{C}_i^e = (w_1, ..., \mathbf{d}_e, ..., w_m)$[2]. We then compute the context embedding of the new C_i^e, $f(\mathsf{C}_i^e) \equiv \mathbf{h}_i^e$.

After getting $\mathbf{h}_i^e$ and $\mathbf{d}_e$, we wish to compute the probability of candidate e being the correct entity $\tilde{e}$ missing in context C_i. This probability is the output of the predictor:

$$P(e = \tilde{e}|\mathsf{C}_i, \mathsf{L}_e) = \sigma((\mathbf{h}_i^e)^T W \mathbf{d}_e + \mathbf{b})$$

where σ is the sigmoid function, W and $\mathbf{b}$ are model parameters.

The cross term $(\mathbf{h}_i^e)^T W \mathbf{d}_e$ is a dot product between $\mathbf{h}_i^e$ and $\mathbf{d}_e$ that weighs the dimensions differently based on the learned parameters W. Similar prediction methods have been used successfully for question answering (Bordes et al., 2014; Yu et al., 2014) and dialogue response retrieval (Lowe et al., 2015).

We also experimented with only feeding $\mathbf{h}_i^e$ to the predictor, without the cross term, and found this slows down training the lexical encoder. While $\mathbf{h}_i^e$ is a function of $\mathbf{d}_e$, using $\mathbf{d}_e$ in the cross term $(\mathbf{h}_i^e)^T W \mathbf{d}_e$ provides a much shorter gradient path from the loss to the lexical encoder through $\mathbf{d}_e$, thus allowing both modules to learn at the same pace.

[2]We mix the word / vector notation here since each word w is replaced by its corresponding word embedding vector.

Given a context, the model outputs a probability for each entity $e \in \mathcal{E}$. Entities in the candidate set are then ranked against each other according to their predicted probabilities. The entity with the highest probability is considered as the most plausible answer for the missing entity in the current context. We consider the model to make an error if that entity is not $\tilde{e}$.

4.2 Hierarchical Double Encoder (HIERENC)

The double encoder architecture mentioned above considers each context independently. However, since each document consists of a sequence of contexts, the knowledge carried by other contexts in $\mathcal{C}$ could also provide useful information for the decision process of C_i. To that end, we propose a hierarchical model structure by adding a LSTM network r, which we call the temporal network (see Figure 3), on top of the double encoder architecture. Since a document is a sequence of C_is, each *time step* of this network consists of one such context, and thus is indexed with i.

Since we already have a context encoder f, we reuse the output of $f(\mathsf{C}_i^e)$ as the input of r at time step i. More specifically, we combine the embeddings generated by f into a single one via averaging: $\bar{\mathbf{h}}_i = \frac{1}{|\mathcal{E}|} \sum_{e' \in \mathcal{E}} \mathbf{h}_i^{e'}$, which then serves as the input to the temporal network for context C_i. Note that alternatively, one could aggregate information about the past predictions through other means like policies or soft attention. However, this would introduce extra complexities to the learning process. As such, we use averaging to that end.

Finally, at each time step i, the temporal network outputs an embedding $r_i(\mathsf{C}_1, ..., \mathsf{C}_n) \equiv \mathbf{r}_i$. We use this temporal embedding to predict the probability of the context-entity pair with a

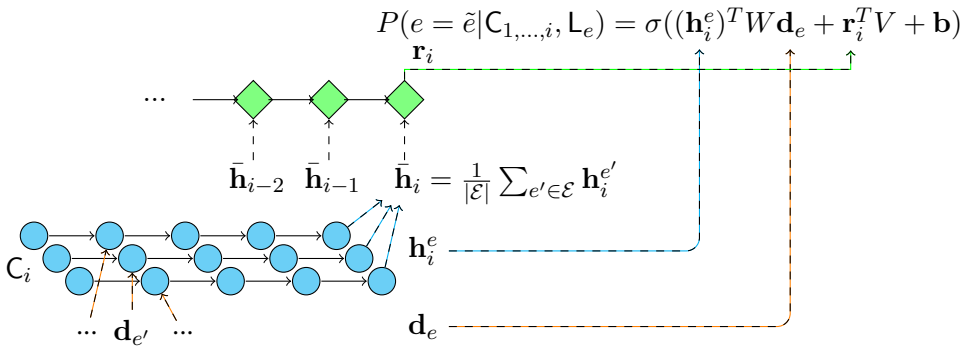

Figure 3: Our hierarchical encoder architecture. Each entity e is encoded as $\mathbf{d}_e$, at each time step, $\mathbf{h}_i^e$ is computed for each e. $\bar{\mathbf{h}}_i$ is the average encoding, which is fed as input to the temporal network r (green diamonds). The temporal network produces $\mathbf{r}_i$, which is used to compute $P(e = \tilde{e}|\mathsf{C}_{1,\ldots,i}, \mathsf{L}_e)$.

slightly altered logistic predictor:

$$P(e = \tilde{e}|\mathsf{C}_{1,\ldots,i}, \mathsf{L}_e) = \sigma((\mathbf{h}_i^e)^T W \mathbf{d}_e + \mathbf{r}_i^T V + \mathbf{b})$$

where W, V and $\mathbf{b}$ are model parameters. The entities in candidate set are again ranked against each other based on their probabilities.

5 Experiments

5.1 Setup

We randomly partition the data into training, validation and test sets. The training set consists of approximately 80% of the total documents, the validation and test sets comprise about 10% each.

In our experiments, the context windows are defined as the sentences where the special ****blank**** tokens are found; the lexical definitions for each entity are the first sentences of their Freebase descriptions. We experimented with different configurations of defining contexts and entity definitions, such as expanding the context windows by including sentences that come before and after the one where blank is found, as well as taking more than one sentence out of the entity description. However, results on validation set show that increasing the context window size and the definition size had very little impact on accuracies, but drastically increased the training time of all models. We thus chose to use only the immediate sentence of the context and the first sentence of the entity description.

To train our models, we use the correct missing entity for each blank as the positive example and all other entities in the candidate set as the negative examples, which we found to be more beneficial empirically than using only a subset of rest of the candidate set. During the testing phase, we present each entity in the candidate set to our models and record the probabilities output by the models. The entity with the highest probability is chosen as the model prediction. For all gradient-based methods, including both baseline models and our proposed models, the learning objective is to minimize the binary cross-entropy of the training data.

We measure the performance on our entity prediction task using the *accuracy*; that is, the number of correct entity predictions made by the model divided by the total number of predictions. This is equivalent to the metric of Recall@1 that is often used in information retrieval.

5.2 Baselines

In order to demonstrate the effects of using lexical resources as external knowledge for solving the task, we present three sets of baselines: one set of simple baselines (RANDOM and FREQENT), one LSTM-based model that only relies on the contexts and does not utilize the definitions (CONTENC), and another set of models that do make use of the entity definitions but in a simplistic fashion (TF-IDF + COS and AVGEMB + COS).

RANDOM For each context in a given document, this baseline simply selects an entity from the candidate set uniformly at random as its prediction.

FREQENT Under this baseline, we rank all entities in the candidate set by the number of times that they appear in the document. For each blank in the document, we always choose the entity with the highest frequency in that document as the prediction. Note that this baseline has access to extra information compared to the other models, in par-

ticular the total number of times each entity appears in the document.

CONTENC Instead of using their definitions, entities are treated as regular tokens in vocabulary. Thus for a particular entity e, the context sequence $\mathsf{C}_i = (w_1, ..., w_{blank}, ..., w_m)$ becomes $(w_1, ..., w_e, ..., w_m)$. We feed the sequence C_i into the context encoder and as usual take the last hidden state as the context embedding $\mathbf{h}_i^e$. Thus given C_i and $e \in E$, the probability of e being the correct missing entity is:

$$P(e = \tilde{e}|\mathsf{C}_i) = \sigma((h_i^e)^T W + \mathbf{b})$$

where again σ is the sigmoid function, W and $\mathbf{b}$ are model parameters. This model is essentially a language model baseline, that does not make use of the external a priori knowledge.

TF-IDF + COS This method takes the term frequency-inverse document frequency (TF-IDF) vectors of the context and the entity definition as their corresponding embeddings. The aggregations of contexts and definitions are treated as their own corpora, and two separate TF-IDF transformers are fitted. Candidate entities are ranked by the *cosine similarity* between their definition vectors and the context vector. The entity with the highest cosine similarity score is chosen as the prediction.

AVGEMB + COS This baseline computes the context embedding by taking the *average* of some pre-trained word embeddings. The entities' embeddings are computed in the same way. In our experiments, we choose to use the published *GloVe* (Pennington et al., 2014) pre-trained word embeddings. Same as above, the prediction is made by considering the cosine similarity between the context embedding and the entity embeddings.

5.3 Hyperparameters

For the CONTENC baseline, we choose 300 as the size of hidden state for the encoder. For the DOUBENC and the HIERENC models, the size of hidden state for both the context encoder and the lexical encoder is set to 300. An RNN with 200 LSTM units is used as the temporal network in the hierarchical case. All three models are trained with stochastic gradient descent with Adam (Kingma and Ba, 2015) as our optimizer, with learning rates of 10^{-3} used for CONTENC and 10^{-4} used for DOUBENC as well as HIERENC. Models with the best performance on validation set are saved and used to test on test set.

Model	Accuracy Valid	Test
Fixed Baselines		
RANDOM	29.4%	30.1%
FREQENT	32.9%	33.1%
Without External Knowledge		
CONTENC	39.3%	39.6%
With External Knowledge		
TF-IDF + COS	29.2%	30.0%
AVGEMB + COS	35.5%	35.9%
DOUBENC	54.7%	54.0%
HIERENC	**57.3%**	**56.6%**

Table 3: Empirical results on *Wikilinks Entity Prediction* dataset for proposed baselines and models.

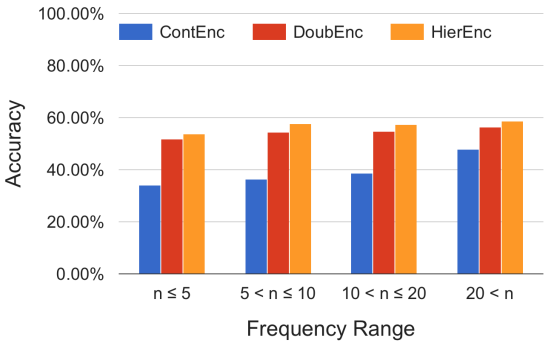

Figure 4: Accuracies of CONTENC, DOUBENC, and HIERENC on test set, for different frequency ranges; n is entity frequency in the entire corpus.

5.4 Results

Empirical results are shown in Table 3. We test our proposed model architectures (detailed in Section 4), along with baselines described in Section 5.2.

It is clear from Table 3 that models that only use contextual knowledge give relatively poor performance compared to the ones that utilize lexical resources. The large discrepancy between the context encoder and the double encoder shows that lexical resources play a crucial role in solving the task. The best result is achieved by the hierarchical double encoder, which confirms that knowing about previous contexts is indeed beneficial to the prediction at the current time step.

We performed statistical significance tests on the predictions from CONTENC, compared to the predictions made by DOUBENC and HIERENC respectively. Both tests yielded $p < 10^{-5}$. We also computed the p-value between DOUBENC and HI-

Context & Prediction
[...] We heard from Audrey Bomse, who is with the Free Gaza movement. She was in _____, Cyprus. [...]
CONTENC: Istanbul HIERENC: Larnaca
Candidate Set
Istanbul: Istanbul is the most populous city in Turkey, and the country's economic, cultural, and historical center.
Larnaca: Larnaca is a city on the southern coast of Cyprus and the capital of eponymous district.
Ben Macintyre: Ben Macintyre is a British author, historian, reviewer and columnist writing for The Times newspaper.
(Other candidate entities.....)

Table 4: An example from the test set, with the predictions made by CONTENC and HIERENC; HIERENC was able to successfully predict the correct missing entity, *Larnaca*.

ERENC, with $p \approx 0.003$. This suggests that the performance improvement achieved by the hierarchical model is statistically significant.

6 Discussion

Figure 4 provides a performance breakdown of test accuracies over various entity frequencies for CONTENC, DOUBENC, and HIERENC. As we can see, the biggest performance gap between the baseline and our two proposed models occurs when $n \leq 5$; as entity frequencies increase, the accuracy of CONTENC also increases. This confirms the value and necessity of lexical resources, especially when entities appear extremely infrequently. We also see that HIERENC outperforms DOUBENC consistently over all frequency ranges. This suggests that by propagating information from the past through temporal network, the hierarchical model is able to reason beyond the local context, thus achieve higher accuracies.

Table 4 shows an example found in the test set, along with the predictions from CONTENC and HIERENC. Even though the context encoder baseline was able to identify that the missing entity should be a city, it incorrectly predicted *Istanbul*. This is likely because *Istanbul* appears 86 times in the dataset, whereas *Larnaca* appears only twice in the test set, *and not at all in the training set*. It seems that, although the context encoder was able to derive a strong association between Istanbul and Middle Eastern geolocations, such knowledge was not learned for Larnaca because of the lack of examples. Conversely, the hierarchical double encoder was able to take both the context and the external knowledge into account and successfully predicted the correct missing city.

One interesting observation is the margin of difference in accuracy between the context encoder and the embedding average baseline. The context encoder, which is a relatively sophisticated context-only model, only slightly outperforms the simple embedding average baseline that has no learning component. This suggests that the entity definitions are valuable in solving such tasks even when it is used in a rather simplistic way.

As we discussed in Section 5.1, we found in initial experiments that using a large context window size (including sentences before and after the sentence where blank token is found) does not have any significant positive impact on the results. This may imply that words that are most informative about the missing entity in the blank are generally found in vicinity of the blank. It is also likely that more sophisticated models will be able to use the surrounding context information more effectively, leading to greater performance increases.

7 Conclusions

In this paper, we examined the use of external knowledge in the form of lexical resources to solve reading comprehension problems. Specifically, we propose the problem of rare entity prediction. In our *Wikilinks Rare Entity Prediction* dataset, the majority of the entities have very low frequencies across the entire corpus; thus, models that solely rely on co-occurrence statistics tend to under-perform. We show that models leveraging the Freebase descriptions achieve large performance gains, particularly when this information is incorporated intelligently as in our double encoder-based models.

For future work, we plan to examine the effects of other knowledge sources. In this paper, we use entity definitions as the source of external knowledge. However, Freebase also contains other types of valuable information, such as relational information between entities. Thus, one potential direction for future work would be to incorporate both relational information and lexical definitions.

We have demonstrated the crucial role that external knowledge plays in solving tasks with many

rare entities. We believe that incorporating external knowledge into other systems, such as dialogue agents, should also see similar positive results. We plan to explore the idea of external knowledge integration further in future research.

Acknowledgements

This work is supported by Samsung Advanced Institute of Technology (SAIT). We would like to thank the anonymous reviewers for their comments and suggestions. We would also like to thank Dzmitry Bahdanau, Tom Bosc, and Pascal Vincent for their discussions on related work, as well as Timothy O'Donnell for his feedback during the writing of this paper.

References

Sungjin Ahn, Heeyoul Choi, Tanel Pärnamaa, and Yoshua Bengio. 2016. A neural knowledge language model. *arXiv preprint arXiv:1608.00318* .

Dzmitry Bahdanau, Tom Bosc, Stanislaw Jastrzebski, Edward Grefenstette, Pascal Vincent, and Yoshua Bengio. 2017. Learning to compute word embeddings on the fly. *arXiv preprint arXiv:1706.00286* .

Justin L. Barrett and Melanie A. Nyhof. 2001. Spreading non-natural concepts: the role of intuitive conceptual structures in memory and transmission of cultural materials. *Journal of Cognition and Culture* 1(1):69–100.

Yoshua Bengio, Patrice Simard, and Paolo Frasconi. 1994. Learning long-term dependencies with gradient descent is difficult. *IEEE Transactions on Neural Networks* 5(2):157–166.

Kurt Bollacker, Colin Evans, Praveen Paritosh, Tim Sturge, and Jamie Taylor. 2008. Freebase: a collaboratively created graph database for structuring human knowledge. In *Proceedings of the 2008 ACM SIGMOD International Conference on Management of Data*. ACM, pages 1247–1250.

Antoine Bordes, Jason Weston, and Nicolas Usunier. 2014. Open question answering with weakly supervised embedding models. In *Joint European Conference on Machine Learning and Knowledge Discovery in Databases*. Springer, pages 165–180.

Danqi Chen, Jason Bolton, and Christopher D. Manning. 2016. A thorough examination of the cnn/daily mail reading comprehension task. In *Association for Computational Linguistics (ACL)*.

Thomas G. Dietterich. 2002. Machine learning for sequential data: a review. In *Structural, syntactic, and statistical pattern recognition*, Springer, pages 15–30.

Oren Etzioni, Michele Banko, and Michael J. Cafarella. 2006. Machine reading. In *AAAI*. volume 6, pages 1517–1519.

Daniel Fried and Kevin Duh. 2015. Incorporating both distributional and relational semantics in word representations. In *ICLR'15: International Conference on Learning Representations (workshop)*.

Felix A. Gers, Nicol N. Schraudolph, and Jürgen Schmidhuber. 2002. Learning precise timing with lstm recurrent networks. *Journal of Machine Learning Research* 3(August):115–143.

Karl Moritz Hermann, Tomas Kocisky, Edward Grefenstette, Lasse Espeholt, Will Kay, Mustafa Suleyman, and Phil Blunsom. 2015. Teaching machines to read and comprehend. In *Advances in Neural Information Processing Systems*. pages 1684–1692.

Felix Hill, Antoine Bordes, Sumit Chopra, and Jason Weston. 2016a. The goldilocks principle: reading children's books with explicit memory representations. In *ICLR 2016*.

Felix Hill, Kyunghyun Cho, Anna Korhonen, and Yoshua Bengio. 2016b. Learning to understand phrases by embedding the dictionary. *Transactions of the Association for Computational Linguistics (TACL)* .

Sepp Hochreiter and Jürgen Schmidhuber. 1997. Long short-term memory. *Neural Computation* 9(8):1735–1780.

Diederik Kingma and Jimmy Ba. 2015. Adam: a method for stochastic optimization. In *ICLR 2015*.

Teng Long, Ryan Lowe, Jackie Chi Kit Cheung, and Doina Precup. 2016. Leveraging lexical resources for learning entity embeddings in multi-relational data. In *ACL 2016*.

Ryan Lowe, Nissan Pow, Iulian Serban, and Joelle Pineau. 2015. The Ubuntu dialogue corpus: a large dataset for research in unstructured multi-turn dialogue systems. In *Proceedings of SIGDIAL, 2015*.

George A Miller. 1995. Wordnet: a lexical database for English. *Communications of the ACM* 38(11):39–41.

Jeffrey Pennington, Richard Socher, and Christopher D. Manning. 2014. GloVe: global vectors for word representation. In *EMNLP*. volume 14, pages 1532–1543.

Pranav Rajpurkar, Jian Zhang, Konstantin Lopyrev, and Percy Liang. 2016. SQuAD: 100,000+ questions for machine comprehension of text. In *Empirical Methods in Natural Language Processing (EMNLP), 2016*.

Alan Ritter, Sam Clark, Oren Etzioni, et al. 2011. Named entity recognition in tweets: an experimental study. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing*. Association for Computational Linguistics, pages 1524–1534.

Roger C. Schank and Robert Abelson. 1977. *Scripts, goals, plans, and understanding*. Erlbaum.

Sameer Singh, Amarnag Subramanya, Fernando Pereira, and Andrew McCallum. 2012. Wikilinks: a large-scale cross-document coreference corpus labeled via links to Wikipedia. Technical Report UM-CS-2012-015, University of Massachusetts, Amherst.

Adam Trischler, Tong Wang, Xingdi Yuan, Justin Harris, Alessandro Sordoni, Philip Bachman, and Kaheer Suleman. 2016. NewsQA: a machine comprehension dataset. *arXiv preprint arXiv:1611.09830* .

Lei Yu, Karl Moritz Hermann, Phil Blunsom, and Stephen Pulman. 2014. Deep learning for answer sentence selection. In *NIPS Deep Learning and Representation Learning Workshop, Montreal*.