Cloze Question Answering using Semi-structured Knowledge and a Language Model

Ezra Winston[1], Bhuwan Dhingra[1], Kathryn Mazaitis[1], Graham Neubig[1], William W. Cohen[1,2]
[1]School of Computer Science, Carnegie Mellon University
[2]Google AI
{ewinston, bdhingra, krivard, gneubig, wcohen}@cs.cmu.edu

Abstract

Modern Question Answering (QA) systems rely on both knowledge bases (KBs) and unstructured text corpora as sources for their answers. While KBs generally offer more precise answers than unstructured text, they do not cover all domains and the cost of constructing them in novel domains is high. In this paper we consider a third setting: QA using text and semi-structured knowledge, in the form of a corpus of entity-labeled documents. We propose CASE, a hybrid of an RNN language model and an entity co-occurrence model, where the entity co-occurrence model is learned from the entity-labeled corpus. We test the model on several cloze (fill-in-the-blank) QA datasets: QUASAR-S, a corpus of tagged posts from Stack Overflow; SPADES, a dataset entity-linked to Freebase; and three new datasets we collect from community-QA websites analogous to QUASAR-S (and make publicly available). On all benchmarks, the proposed model shows large accuracy gains over strong baselines that do not incorporate a KB; where a KB is available, the model is competitive with the state-of-the-art KB-based method, despite using only semi-structured knowledge.

1 Introduction

Factoid QA is the task of providing short factual answers to questions posed in natural language. Recent systems for factoid QA typically use either Knowledge-Bases (KBs) (Yu et al. 2017; Bordes et al. 2015; Bordes, Chopra, and Weston 2014; Yih et al. 2015; Berant et al. 2013), unstructured text (Chen et al. 2017; Dhingra, Mazaitis, and Cohen 2017; Wang et al. 2017a; 2017b), or both (Das et al. 2017; Gardner and Krishnamurthy 2017; Xu et al. 2016a; 2016b). While KB approaches benefit from structured information, QA tasks which require domain-specific knowledge present a unique challenge since relevant knowledge bases are often lacking (Dong et al. 2014). Text-based approaches which query unstructured sources have improved greatly with recent advances in machine reading and comprehension, but effective combination of search and reading systems is an active research challenge (Chen et al. 2017).

Semi-structured text offers a promising source of knowledge for QA tasks which lack relevant KBs. This paper focuses on answering cloze (fill-in-the-blank) questions using semi-structured knowledge consisting of a background corpus of documents d each tagged with context entities c. We introduce a new task, *QA with context entities*, which is well-suited to make use of such background corpora. For this task, in addition to the standard question sentence q and answer entity a, there is a set of context entities c_q about which the question is asked. For example, QUASAR-S (Dhingra, Mazaitis, and Cohen 2017) consists of cloze-style questions about computer programming, collected from the community-QA website Stack Overflow. Each question q is derived from a definition of software entity c_q, and the answer is another software entity a (Figure 1, (top)). The dataset also includes a large background corpus of community-generated posts; users have tagged each post d with a set of context entities c, and potential answer entities a within documents have been automatically annotated (Figure 1, (bottom)).

Question

Tag c_q	**django**
Sentence q	*django is an open-source web application framework written in* _____
Answer a	**python**

Post Excerpt

Tags c	**django**, **subprocess**, **wsgi**
Sentence d	*So how can I maximize performance in python for this kind of scenario?*
Answer a	**python**

Figure 1: Example from QUASAR-S. Questions are clozes constructed from Stack Overflow tag definitions. The background corpus contains excerpts from tagged posts.

To our knowledge, ours is the first work exploring the QA with context entities task. However, this task occurs often in practical settings. Beyond the community-QA setting such as Stack Overflow that we focus on, other examples include customer questions in the context of a particular product, or questions posted by an editor in the context of a particular Wikipedia page. Also, as we demonstrate on the SPADES dataset, general QA tasks can be viewed as an instances of this task, whenever potential context entities can be identified in the question by entity-linking methods.

To effectively leverage such semi-structured information, we propose CASE (*Context-Adjusted Syntax Embeddings*), a hybrid of a recurrent neural network language model (RNN-LM) $P(a|q)$ that predicts an answer entity a from the surrounding question text, and a co-occurrence model $P(a|c_q)$ which predicts a from the context entities. We test several instantiations of this framework, and find that a simple context model based on co-occurrence statistics performs best.

CASE obtains state-of-art results on QUASAR-S, outperforming both search-and-read methods and an RNN-LM baseline. To understand why, we analyze the predictions of each component and the output embeddings of the RNN-LM component. Our analysis demonstrates a useful division of labor: the RNN-LM picks out the "type" of the answer entity based on question syntax, while the context model picks out the semantically meaningful entity based on co-occurrence counts.

While CASE is tailored for cases where no KB is available, we demonstrate that it can perform well on general cloze QA tasks as well, when context entities can be extracted from question sentences. This allows us to compare CASE with existing methods for cloze QA that rely on KBs. We evaluate on SPADES (Bisk et al. 2016), a dataset tailored for QA using Freebase (Bollacker et al. 2008). SPADES consists of cloze questions from sentences containing two or more Freebase entities (Figure 3). We treat all entities in question q as context entities c_q, and create a background corpus from the training questions by treating each as a documents d with the Freebase entities it contains as context c. Using only this small, semi-structured knowledge source, CASE is competitive with a state-of-the-art method which uses Freebase.

To summarize, our contributions are as follows. (1) We introduce the *QA with context entities* task and propose CASE, a hybrid language/context model for this task. (2) We show that CASE outperforms RNN-LM and "search-and-read" baselines, improving the previous state-of-the-art accuracy on QUASAR-S by more than 11%, and we compare several instantiations of CASE of varying complexity, and show that a simple, easy-to-implement model performs best on multiple benchmarks. (3) We demonstrate that on the SPADES dataset, where no background text corpus is available, CASE still obtains results comparable to (2.7% less than) state-of-the-art KB methods, using only co-occurrence data derived from the training corpus; combining CASE with the previous state-of-the-art further improves accuracy. (4) We release three new datasets similar to QUASAR-S, constructed from smaller community QA sites Math Exchange, TeX Exchange, and English Exchange, and show that CASE outperforms LM baselines in all cases. (5) Finally, we provide qualitative analysis of the entity embeddings produced by CASE, showing that they more faithfully capture entity "type" information.

2 Background

Problem Definition

Prior work has focused on QA given a KB, document, or document corpus. Here, we focus on incorporating semi-structured data that comes in the form of context entities, in settings where no KB is available. In particular we focus on cloze-style QA and refer to this task as *Cloze QA with context entities*. A cloze question q is of the form $w_1, \ldots, \_blank\_, \ldots w_n$ and the task is to identify the answer entity $a \in \mathcal{A}$ from answer vocabulary $\mathcal{A}$ which replaces *_blank_*. For our task, each question q is also accompanied by a set context entities c_q. To answer these question, we are given a semi-structured corpus consisting of (context entities, document) pairs, $S = \{(c, d)\}$, where $c = \{c_1, \ldots, c_m\}$ is a set of context entities. In this paper, we also assume that some documents d contain annotated entities from the answer vocabulary $\mathcal{A}$. An instance of the task is a tuple (q, c_q, a, S) where a is the correct missing entity, and we wish to model $P(a|q, c_q, S)$. Figure 1 shows how these variables are instantiated for QUASAR-S.

Related tasks

Past work has studied several variants of question-answering (QA). In *knowledge-based QA* (KB QA) the goal is to model $P(a|q, \mathcal{K})$, the probability of answer a given question q and KB $\mathcal{K}$. In *Reading comprehension* (RC) one models $P(a|q, d)$, the probability of answer a given q and a document d containing the answer. RC systems can be extended to more general tasks in the *search and read* setting, where one models $P(a|q, D)$, the probability of a given question q and document corpus $D = \{d_1, \ldots, d_N\}$ (where D may come from a search engine). These problems all relate to *QA with context entities*, which uses a model $P(a|q, c_q, S)$ that predicts the probability of a given question q, context entities c_q, and an entity-tagged corpus $S = \{(c, d)\}$. Cloze QA can also be approached as *language modeling* where, given a sequence $s = w_1, \ldots, w_{k-1}$ (and sometimes also $s' = w_{k+1}, \ldots w_K$), one models $P(w_k|s)$.

3 CASE Models

We propose to use a language model $f(q, a) \propto P(a|q)$ together with a *context-entity* model $g(c, a) \propto P(a|c)$ to model answer probabilities $P(a|c, q)$. We make an independence assumption similar to that of Naive Bayes, modeling the question and context entities as independent given the answer:

$$P(q, c|a) = P(q|a)P(c|a).$$

This allows us to model the LM and the context-entity model separately and leads to the predictive distribution

$$\begin{aligned} P(a|q, c) &\propto P(a|q)P(a|c)/P(a) \\ &\propto f(q, a)g(c, a)/P(a). \end{aligned}$$

The best-performing model in our experiments is CASE-CC. We instantiate f as a bidirectional GRU network (Bi-GRU) following Dhingra, Mazaitis, and Cohen (2017). Let $W_1 \in \mathbb{R}^{H \times V}$ be a word embedding matrix where V is the

size of question word vocabulary $\mathcal{V}$ and H is the embedding dimension. Let $W_2 \in \mathbb{R}^{A \times 2H}$ be the output answer embedding matrix where A is the size of answer vocabulary $\mathcal{A}$. For predicting the entity at answer index i in question $q = w_1, \ldots, w_K$ we concatenate the forward and backward GRU outputs at that index:

$$x = [W_1 w_1, \ldots, W_1 w_K]$$
$$h = [fGRU(x)_{i-1}, bGRU(x)_{i+1}]$$
$$\log(f(q, \cdot)) = W_2 h$$

where the w_k are one-hot encoded and $fGRU(x)$ and $bGRU(x)$ are the sequential outputs of the forward and backward GRUs.

For the context model g we use unsmoothed **C**o-occurrence **C**ounts calculated from the entity-labeled training set. Specifically, given context entities $c = \{c_1, \ldots, c_m\}$ we compute

$$g(c, a) = \text{avg}_i \#(a, c_i)/\#(c_i).$$

In other words, for each context entity, we compute the empirical probability of co-occurrence with the answer entity, and then average over context entities in the context entity set. Finally, answer predictions are

$$P(\cdot|q, c) = \sigma\left(\log(f(q, \cdot)) - \log(g(c, \cdot)) - b\right)$$
$$\propto f(q, \cdot) g(c, \cdot) / \exp(b)$$

where b is a learned bias and σ denotes softmax. While the LM f learns to predict the answer based on the surrounding sentence, the context model g makes predictions based on context entities. This division of labor allows the LM to focus more on local syntactic features while relying on the context model for topical/semantic information.

We also experimented with several alternative entity context models g. For the CASE-AE model, we let $\log(g(c, \cdot)) = \text{avg}_i W c_i$, the **A**verage of context entity **E**mbeddings, where the c_i are one-hot encoded and W is a learned context entity embedding matrix. We also evaluated a context model based on the self-attentional **S**et **E**ncoder suggested by Vinyals, Bengio, and Kudlur (2015) for encoding unordered sets. We call this model CASE-SE. Specifically,

$$q_t = GRU(q^*_{t-1}) \qquad r_t = \sum\nolimits_i a_{i,t} c_i$$
$$d_{i,t} = \langle W c_i, q_t \rangle \qquad q^*_t = [q_t \; r_t]$$
$$a_{i,t} = \sigma(d_{\cdot,t}) \qquad \log(g(c, \cdot)) = W q^*_m$$

where this process is repeated for $t = 0, \ldots, m$ steps, i.e. we take a number of self attention steps equal to the number of context entities.

4 Experiments

Datasets

We conduct experiments on several datasets. QUASAR-S and the new STACKEX datasets, which we created, contain background corpora of unstructured text, and also context that can be used to inform the entity-context model. To compare with methods that explicitly use a KB, we also evaluate performance on SPADES, where questions are designed to be answerable using Freebase. Table 1 shows dataset statistics.

MATHEX

the **z-transform** is a discrete analogue to the laplace-transform in that it maps a time domain signal into a representation in complex frequency plane

TEXEX

align is an environment provided by math packages that permits multiple related equations to be aligned at a common reference point

ENGLISHEX

hyperbaton is any deliberate and dramatic departure from standard word-order

Figure 2: Examples from STACKEX. Context entities are shown in bold, answer entities underlined.

QUASAR-S (Dhingra, Mazaitis, and Cohen 2017) A large cloze-style QA dataset created from the website Stack Overflow (SO), consisting of questions and a background corpus in the computer programming domain. QUASAR-S has the unique feature of requiring deep domain expertise in software, a domain without a rich KB, making it challenging for KB QA. Neither human experts in a closed-book setting (i.e. without access to the background corpus) nor human non-experts in an open-book setting (i.e. with search access to the background corpus) can answer more than 50% of questions, probably because even domain experts are not familiar with all SO topics. However, Dhingra, Mazaitis, and Cohen (2017) note that this is not necessarily an upper bound for automated systems, which could access more background knowledge than any one individual. The difficulty of the task is emphasized by the fact that neither RNN-LM nor the state-of-the-art Gated Attention reader (in a search-and-read setting) obtains more than 70% of human performance. The 37k cloze questions are constructed from the definitions of SO tags by replacing occurrences of software entities with *_blank_*. The SO tag being defined in question q becomes the context entity c_q. The background corpus consists of 27M sentences from the top 50 question and answer threads for each of 4,874 software entities. Each post d is tagged with 1-5 tags which become the document context c. Figure 1 shows an example question and relevant background sentences.

STACKEX[1] To test how CASE performs in settings with less background data, we construct three new QA datasets analogous to QUASAR-S. Following the methodology of Dhingra, Mazaitis, and Cohen (2017), we extract questions and post excerpts from Math Exchange, TeX Exchange, and English Exchange. Each site contains definitions of entity tags in the corresponding domain as well as tagged posts (see Figure 2). As for QUASAR-S, we construct cloze questions from the tag definitions and use the posts as semi-structured knowledge training corpora. Since each of these

[1] Available at [URL in camera-ready]

	QUASAR-S	SPADES	MATHEX	TEXEX	ENGLISHEX
Training Qns	31,049	190,972*	-	-	-
Val Qns	3,174	4,763	365	317	61
Test Qns	3,139	9,309	389	328	68
Background Exs	17.8 mil†	-	123,179	158,624	58,077
Context Entities	44,375	53,961	1600	1542	939
Answer Entities	4,875	53,961‡	320	242	61

Table 1: Statistics of QUASAR-S, STACKEX, and SPADES. *Each entity in the 79,247 original training questions is replaced to produce a new training question; †Each entity in the 26.6 mil. SO posts is replaced to produce a training example; ‡While 1.8 million entities are present in the SPADES Freebase extract, we restrict prediction to entities appearing in the training questions.

Question q : *Google acquired _____ which was founded in Palo Alto*

Context entities c_q : **Google**, **Palo Alto**

Answer entity a : **Nest**

Figure 3: Example question from SPADES

Stack Exchange sites defines only a few hundred tags, we split questions evenly into validation and test sets and do not hold out a training question set.

SPADES (Bisk et al. 2016) A set of 93k cloze-style questions constructed from sentences from ClueWeb09 which have been automatically annotated with Freebase entities by Gabrilovich, Ringgaard, and Subramanya (2013). Specifically, the sentences in SPADES contain two or more Freebase entities that are linked by at least one relation path in Freebase. Das et al. (2017) provide strong memory-network baselines which leverage both Freebase and training questions as knowledge sources. Unlike QUASAR-S, there are no explicit tags present. We therefore take the Freebase entities in each question sentence q (usually one) as the context entities c_q. Also, SPADES has no background text corpus $S = \{(c, d)\}$. We instead use the training questions as a small background corpus, considering each question as a document d and its annotated Freebase entities as the context entities c. Figure 3 shows an example question.

Experimental Setup

Across all CASE-CC experiments we instantiate LM f as a BiGRU following the baseline from Dhingra, Mazaitis, and Cohen (2017). Training is conducted using a learning rate of 0.001 annealed by 50% after each epoch. We use the Adam (Kingma and Ba 2014) optimizer with default hyperparameters ($\beta_1 = 0.9$, $\beta_2 = 0.999$, $\varepsilon =$ 1e-8), with 100-dimensional sentence word embeddings pretrained using skip-gram word2vec (Mikolov and Zweig 2012). For context model g we use co-occurrence counts as described above. The BiGRU is trained with the co-occurrence counts model present.[2]

QUASAR-S and STACKEX While the goal is to predict answers on the question set constructed from tag definitions, we first train on the much larger post corpus. We create a training example for each occurrence of an answer entity in a post by replacing that entity with *_blank_* and treating it as the answer. We use the post tags as the context entities c. For QUASAR-S, since the model is trained on posts and evaluated on the question set, we fine-tune the model on the training questions. We follow an approach similar to that used by Chu, Dabre, and Kurohashi (2017) for neural translation transfer learning: after training on the large post corpus until convergence, we then train on a 50/50 mix of training questions and posts. This procedure avoids overfitting to the much smaller set of training questions. On the other STACKEX datasets, where no training questions exist, we train only on the post corpora and do not conduct fine-tuning.

On QUASAR-S we compare to the baselines reported in Dhingra, Mazaitis, and Cohen (2017) and to two other ways of incorporating context (BiGRU-PT, CBiGRU) described below in Section 4. On STACKEX we replicate the LM baselines using code obtained from the author, but omit the GA reader baseline since this requires a set of training clozes which are absent form STACKEX. On both datasets we also compare to the model CC consisting of only the co-occurrence counts model g, ignoring question sentences.

SPADES We follow the same experimental procedure as for QUASAR-S and use the training/validation/test split from Das et al. (2017). We compare to the ONLYTEXT, ONLYKB, and UNISCHEMA models of Das et al. (2017). In addition to CASE-CC, we train hybrid models that add the co-occurrence counts as a bias to the output softmax of the ONLYKB and UNISCHEMA models. For these models we use the code, parameters, and training procedures of UNISCHEMA but train the model with the co-occurrence bias present. Finally, we compare to a model CC consisting of the co-occurrence model g only.

Results

QUASAR-S Results and baselines are reported in Table 2. The fine-tuned CASE-CC obtains an accuracy of 45.2%, a gain of 11.6% over the best previously reported results of Dhingra, Mazaitis, and Cohen (2017), obtained by BiGRU (33.6%). Dhingra, Mazaitis, and Cohen (2017) also report performance of several search-and-read methods, the best of which uses the neural gated-attention (GA) reader. When the answer is present in a retrieved document, the GA reader

[2]Source code is available at [URL in camera-ready].

Method	Val. Acc.	Test Acc.
Human Performance		
Expert (CB)	0.468	-
Non-Expert (OB)	0.500	-
Language Models		
3-gram LM	0.148	0.153
4-gram LM	0.161	0.171
5-gram LM	0.165	0.174
BiGRU LM	0.345	0.336
Search + Read		
WD (SD)	0.100	0.107
MF-e (SD)	0.134	0.136
MF-i (SD)	0.159	0.159
GA (SD)	0.315	0.316
WD (LD)	0.082	0.093
MF-e (LD)	0.128	0.136
MF-i (LD)	0.159	0.159
GA (LD)	0.318	0.321
New Models		
CC	0.128	0.139
CASE-AE	0.314	0.327
CASE-SE	0.330	0.329
BiGRU-PT-5	0.326	0.335
BiGRU-PT-1	0.336	0.342
C-BiGRU	0.342	0.352
BiGRU + ft	0.385*	0.380*
CASE-CC	0.413*	0.413*
CASE-CC + ft	**0.449***	**0.452***

Table 2: Performance comparison on QUASAR-S. Results other than *New Models* and notation are from Dhingra, Mazaitis, and Cohen (2017). ft: fine-tuning on questions; LD: long documents; SD: short documents; GA: gated-attention reader; MF-i, MF-e, WD: search-and-read methods using heuristics to extract answer from retrieved documents; OB: open-book; CB: closed book. *Accuracy gain over next best is significant at the $p < 0.05$ level under an exact McNemar's paired test.

identifies the correct answer 48.3% of the time, but the 65% search accuracy limits overall accuracy to 31.6%. CASE-CC nearly matches the accuracy of the GA reader component alone. The CASE-CC accuracy approaches that of human experts in a closed-book setting (46.8%), and is only 4.8% behind that of non-expert humans in an open-book setting (50.0%). Lastly, we find that fine-tuning on questions improves the performance of both the BiGRU and CASE-CC by about 5%. We report negative results of the other context models below.

STACKEX As for QUASAR-S, we compare CASE-CC to LM baselines. Results are reported in Table 4. On MATHEX and TEXEX, CASE-CC obtains nearly twice the BiGRU accuracy. On ENGLISHEX, where the BiGRU already obtains 47.1% accuracy, CASE-CC obtains +14.7%.

SPADES Results and baselines are reported in Table 3. CASE-CC, trained only on entity co-occurrences in the question text, obtained better accuracy (35.8%) than both the BiGRU (19.9%) and the memory-network ONLYTEXT model of Das et al. (2017), which creates a knowledge base using training question text as facts. CASE-CC performs nearly as well as the memory-network ONLYKB model (38.6%), which uses Freebase facts, or the UNISCHEMA model (39.9%), which uses both text and Freebase facts. The co-occurrence only model CC obtains a surprising 27.9% accuracy. Using co-occurrence counts as a bias in the ONLYKB and UNISCHEMA models improve both by about 2.5%, with the best model UNISCHEMA+CC obtaining 42.3% accuracy.

Method	Val. Acc.	Test Acc.
Text-only Models		
BiGRU	0.184	0.190
ONLYTEXT†	0.253	0.266
CC	0.270	0.279
Bisk et al. (2016)	0.327	-
CASE-CC	0.362*	0.358*
Knowledge-base Models		
ONLYKB†	0.391	0.385
ENSEMBLE†	0.394	0.386
UNISCHEMA†	0.411	0.399
ONLYKB+CC	0.415*	0.403*
UNISCHEMA+CC	**0.427**	**0.423***

Table 3: Performance comparison on SPADES. †(Das et al. 2017). *Accuracy gain over next best (or non-CC version for UNISCHEMA and ONLYKB) is significant at the $p < 0.05$ level under an exact McNemar's paired test.

Discussion

The strong performance of CASE confirms that QA can take advantage of semi-structured text corpora in specialized domains where no KB exists. By incorporating co-occurrence counts, CASE-CC obtains significant accuracy gains across all five datasets. On SPADES, using only the training questions as knowledge source, CASE-CC does almost as well as methods which use Freebase.

CASE outperforms both BiGRU and search-and-read baselines on QUASAR-S. In the first case, we attribute this to the fact that CASE effectively incorporates context entities while BiGRU does not. In addition, the RNN in CASE can focus more on syntactic/type information, leaving the context model g to handle context/semantic information; we explore this further in Section 5. As noted, search-and-read methods such as the GA reader baseline have trouble combining the search and read components. However, CASE even approaches accuracy of the GA-reader component alone on examples where the correct answer is in the retrieved context. This is likely due to training data requirements: while CASE was trained directly on the 17 mil. post corpus, GA-reader was trained on only the 30k training questions, instead using the posts as the source for querying.

Performance on STACKEX follows a similar trend to

	MATHEX		TEXEX		ENGLISHEX	
Method	**Val. Acc.**	**Test Acc.**	**Val. Acc.**	**Test Acc.**	**Val. Acc.**	**Test Acc.**
CC	0.216	0.198	0.290	0.320	0.311	0.279
3-gram LM	0.200	0.162	0.215	0.216	0.426	0.309
4-gram LM	0.216	0.152	0.233	0.228	0.443	0.294
5-gram LM	0.214	0.147	0.233	0.225	0.426	0.309
BiGRU	0.320	0.242	0.404	0.360	0.557	0.471
CASE-CC	**0.523***	**0.452***	**0.574***	**0.622***	**0.721***	**0.618***

Table 4: Performance comparison on STACKEX datasets. *Accuracy gain over next best is significant at the $p < 0.05$ level under an exact McNemar's paired test.

Question	**CASE-CC**	**BiGRU from CASE**	**CC**
antivirus software is software used to prevent detect and remove malware	malware, antivirus, heuristics	duplicates, malware, scrollbars	antivirus, malware, server
fps is a measure of frame-rate the rate at which ...	frame-rate, cpu, video	data-transfer, execution-time, frame-rate	video, frame-rate, cpu
ffserver is a streaming server for both audio and video	server, video, codec	endpoint, connection -manager, interface	ffmpeg, video, server

Table 5: QUASAR-S examples where CASE-CC gets the correct answer but BiGRU baseline does not. Context entities are shown in bold, answer entities underlined. Ranked predictions are show for CASE-CC and for it's BiGRU and CC components, neither of which make correct predictions individually.

QUASAR-S. STACKEX datasets appear easier overall, which we attribute to the number of candidate answer entities (4,875 for QUASAR-S vs 320, 242, and 61 for the STACKEX datasets). However, the small training sets and lack of training questions (training only uses posts) work in the opposite direction, making these datasets harder.

We find that both the BiGRU and co-occurrence components of CASE are required for good performance. Table 5 shows examples that CASE gets right but the BiGRU baseline gets wrong. Ranked predictions of both LM and CC components of CASE are shown, indicating that neither component alone obtains correct answers. The relative importance of each component varies between datasets. On SPADES, where the training set is much smaller than QUASAR-S, the LM contributes +7.9% accuracy over the CC-only baseline, compared with +31.3% on QUASAR-S. When comparing CC-only to the BiGRU-only baseline, the BiGRU outperforms CC on QUASAR-S (13.9% and 33.6% accuracy, respectively), but the opposite is found on SPADES (27.9% vs 19.0%). BiGRU performance can be attributed to the difference in training data size (17.8 mil for QUASAR-S vs 190,972 for SPADES). Also, CC performs surprisingly well on SPADES, perhaps because the restriction to sentences that correspond to some Freebase relation, together with the fact that the sentences are drawn from a broad corpus with much factual redundancy, results in many repeated entity pairs: for example, given context entity Barack Obama, the answer is United States in 38% of examples.

Negative Results

Neither of the two other entity context models for g, CASE-AE and CASE-SE, showed improvement over the BiGRU baseline (Table 2). In both cases, the model had difficulty learning context entity embeddings. We hypothesize that this is due in part to the highly non-uniform frequency of tags in the posts corpus, compared with the uniform distribution of tags in the test questions which come from definitions. This does not present a problem for the co-occurrence counts model, which captures the relationship between context and answer entities without learning embeddings. Weighting training loss by inverse tag frequency may correct for this and is the subject of future work.

We also experimented with other ways of incorporating context beyond the CASE framework. CBiGRU is similar to CLSTM (Ghosh et al. 2016). Instead of inputting embedding $W_1 w_i$ to the BiGRU, we input $[Wc \;\; W_1 w_i]$ where Wc is an embedding for a tag entity c. BiGRU-PT extends each training sentence by prepending either 1 or up to 5 context entities to the beginning of each training sentence, potentially allowing the BiGRU to condition it's computation based on this context. We found that these methods of incorporating context did not improve over BiGRU (Table 2). As with CASE-AE and CASE-SE, CBiGRU could not learn good context entity embeddings. That BiGRU-PT did not improve performance matches our intuition, since RNNs have trouble remembering context from the beginning of a sequence.

5 Analysis of Embeddings

We observe that by modeling context and question sentence separately, CASE factors entity representations into a semantic/contextual component given by context and a syntactic/type component given by the sentence. To assess the extent of this property we analyze the output entity embed-

Seed	CASE-CC	BiGRU
ipod	ipod-touch, ipad, apple-tv	ipad, itunes, 3g
xcode	eclipse, visual-studio, xamarin-studio	cocoapods, gdb, rubymine
intellij-idea	netbeans, phpstorm, eclipse	spring-mvc, java-ee, rubymine
unit-testing	debugging, profiling, refactoring	integration-testing, tdd, dependency-injection
linear-regression	logistic-regression, random-forest, least-squares	logistic-regression, machine-learning, time-series

Table 6: NNs in the CASE-CC and BiGRU output embedding space. Entities of the same type as seed are underlined.

dings learned by CASE-CC. To obtain (noisy) ground-truth types for SO entities, we link entities to Wikidata (Vrandečić and Krötzsch 2014) via the links to Wikipedia in Stack Overflow tag definitions. We choose 20 groups of entities such as *Programming Languages* and *Network Protocols*. SPADES types are obtained from Freebase.

To compare CASE-CC output embeddings to those of the BiGRU baseline, we use each to predict type using 1-nearest-neighbor with cosine distance. Consistent with our expectations, CASE-CC embeddings obtain better accuracy (QUASAR-S: 63.3%, SPADES: 77.9%) than those of BiGRU (QUASAR-S: 57.4%, SPADES: 71.3%). Qualitatively, we also observe many instances in which the nearest neighbors in CASE-CC embedding space are of the same type (e.g both Java IDEs) while nearest neighbors in BiGRU embedding space may be only semantically related (e.g. a Java IDE and a Java web framework) (Table 6).

6 Related Work

Question Answering

Memory networks have proven effective for reasoning over KBs, documents, or jointly over both (Das et al. 2017; Miller et al. 2016; Bordes et al. 2015). While no KBs are available for our task, we do compare to the text-only memory network baseline of Bordes et al. (2015) on SPADES. Recently, the incompleteness of even the largest KBs (Dong et al. 2014) has motivated QA using unstructured text corpora such as Wikipedia instead of a KB. These text-based approaches often follow the *search-and-read* paradigm, involving a search stage, in which relevant documents are retrieved, and a reading stage, in which retrieved passages are read for the correct answer (Chen et al. 2017; Dhingra, Mazaitis, and Cohen 2017; Wang et al. 2017b). Much research has focused primarily on the reading stage (e.g. Choi et al.; Cui et al.; Dhingra et al.; Kadlec et al.; Seo et al.; Wang and Jiang; Xiong, Zhong, and Socher (2016; 2016; 2016; 2016; 2016; 2016; 2016)), with many datasets developed for the reading comprehension task (e.g. Joshi et al.; Nguyen et al.; Rajpurkar et al.; Hermann et al. (2017; 2016; 2016; 2015)). The search-and-read approach is conceivably applicable to the QA with context entities task, but was shown to perform poorly on QUASAR-S; to answer the domain-specific questions in this task, trading off between query recall and reading accuracy proves difficult (Dhingra, Mazaitis, and Cohen 2017), and RNN-LM performs best.

Language Modeling

RNN-based language models have shown increasingly good performance (see Chung et al. (2014) for a comparison). However, RNN-LMs have trouble modeling long-range context as well as predicting rare words (Ahn et al. 2016; Gulcehre et al. 2016). We find that explicitly incorporating predictions based on context entities is critical for the QA-with-context task, since the correct answer entity can be largely dictated by these. When a KB is present, recent RNN-LMs (Ahn et al. 2016; Yang and Mitchell 2017) address these issues by selectively incorporating KB facts. Where no KB is present, several approaches for incorporating more general long-range context in RNN-LMs have also emerged. Following the terminology of Wang and Cho (2015), these approaches either employ *early-fusion*, in which a context vector is concatenated with each RNN input (Ghosh et al. 2016; Mikolov and Zweig 2012), or *late fusion*, in which a context vector is used as a bias before the output nonlinearity of the RNN-LM (Lau, Baldwin, and Cohn 2017; Dieng et al. 2016; Wang and Cho 2015). We compare to one baseline inspired by the early-fusion method CLSTM (Ghosh et al. 2016)). The CASE model is an instance of late-fusion, adding a bias to the RNN output in logit space, prior to softmax. However, it differs from existing models in that the bias is computed based on context entities, rather than topics inferred from the document. Our incorporation of co-occurrence counts with an RNN-LM is related to the hybrid neural/n-gram LMs of Neubig and Dyer (2016), but here again the n-gram models are not based on context entities.

7 Conclusions and Future Work

In this paper, we demonstrated that semi-structured background data can be used to obtain large performance improvements on several cloze QA datasets. The hybrid of a LM and a simple entity co-occurrence model is both effective and easy to implement. CASE shows potential for domain-specific QA tasks such as QUASAR-S, where relevant KBs are not available and search-and-read systems face difficulties. We also see potential to incorporate other data sources into the context entity model, such as HTML web tables. In addition, using more expressive models of context may improve performance. Finally, we showed that CASE embeddings encode type/syntax information. The application of these embeddings to other tasks warrants further investigation.

References

Ahn, S.; Choi, H.; Pärnamaa, T.; and Bengio, Y. 2016. A neural knowledge language model. *arXiv:1608.00318*.

Berant, J.; Chou, A.; Frostig, R.; and Liang, P. 2013. Semantic parsing on freebase from question-answer pairs. In *EMNLP*.

Bisk, Y.; Reddy, S.; Blitzer, J.; Hockenmaier, J.; and Steedman, M. 2016. Evaluating induced ccg parsers on grounded semantic parsing. *arXiv:1609.09405*.

Bollacker, K.; Evans, C.; Paritosh, P.; Sturge, T.; and Taylor, J. 2008. Freebase: A collaboratively created graph database for structuring human knowledge. In *SIGMOD*.

Bordes, A.; Usunier, N.; Chopra, S.; and Weston, J. 2015. Large-scale simple question answering with memory networks. *arXiv:1506.02075*.

Bordes, A.; Chopra, S.; and Weston, J. 2014. Question answering with subgraph embeddings. *arXiv:1406.3676*.

Chen, D.; Fisch, A.; Weston, J.; and Bordes, A. 2017. Reading wikipedia to answer open-domain questions. *arXiv:1704.00051*.

Choi, E.; Hewlett, D.; Lacoste, A.; Polosukhin, I.; Uszkoreit, J.; and Berant, J. 2016. Hierarchical question answering for long documents. *arXiv:1611.01839*.

Chu, C.; Dabre, R.; and Kurohashi, S. 2017. An empirical comparison of simple domain adaptation methods for neural machine translation. *arXiv:1701.03214*.

Chung, J.; Gulcehre, C.; Cho, K.; and Bengio, Y. 2014. Empirical evaluation of gated recurrent neural networks on sequence modeling. *arXiv:1412.3555*.

Cui, Y.; Chen, Z.; Wei, S.; Wang, S.; Liu, T.; and Hu, G. 2016. Attention-over-attention neural networks for reading comprehension. *arXiv:1607.04423*.

Das, R.; Zaheer, M.; Reddy, S.; and McCallum, A. 2017. Question answering on knowledge bases and text using universal schema and memory networks. In *ACL*.

Dhingra, B.; Liu, H.; Yang, Z.; Cohen, W. W.; and Salakhutdinov, R. 2016. Gated-attention readers for text comprehension. *arXiv:1606.01549*.

Dhingra, B.; Mazaitis, K.; and Cohen, W. W. 2017. Quasar: Datasets for question answering by search and reading. *arXiv:1707.03904*.

Dieng, A. B.; Wang, C.; Gao, J.; and Paisley, J. 2016. Topicrnn: A recurrent neural network with long-range semantic dependency. *arXiv:1611.01702*.

Dong, X.; Gabrilovich, E.; Heitz, G.; Horn, W.; Lao, N.; Murphy, K.; Strohmann, T.; Sun, S.; and Zhang, W. 2014. Knowledge vault: A web-scale approach to probabilistic knowledge fusion. In *ACM SIGKDD*. ACM.

Gabrilovich, E.; Ringgaard, M.; and Subramanya, A. 2013. Facc1: Freebase annotation of clueweb corpora, version 1 (release date 2013-06-26, format version 1, correction level 0).

Gardner, M., and Krishnamurthy, J. 2017. Open-vocabulary semantic parsing with both distributional statistics and formal knowledge.

Ghosh, S.; Vinyals, O.; Strope, B.; Roy, S.; Dean, T.; and Heck, L. 2016. Contextual lstm (clstm) models for large scale nlp tasks. *arXiv:1602.06291*.

Gulcehre, C.; Ahn, S.; Nallapati, R.; Zhou, B.; and Bengio, Y. 2016. Pointing the unknown words. *arXiv:1603.08148*.

Hermann, K. M.; Kocisky, T.; Grefenstette, E.; Espeholt, L.; Kay, W.; Suleyman, M.; and Blunsom, P. 2015. Teaching machines to read and comprehend. In *NIPS*.

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

Kadlec, R.; Schmid, M.; Bajgar, O.; and Kleindienst, J. 2016. Text understanding with the attention sum reader network. *arXiv:1603.01547*.

Kingma, D., and Ba, J. 2014. Adam: A method for stochastic optimization. *arXiv:1412.6980*.

Lau, J. H.; Baldwin, T.; and Cohn, T. 2017. Topically driven neural language model. *arXiv:1704.08012*.

Mikolov, T., and Zweig, G. 2012. Context dependent recurrent neural network language model. *SLT* 12:234–239.

Miller, A.; Fisch, A.; Dodge, J.; Karimi, A.-H.; Bordes, A.; and Weston, J. 2016. Key-value memory networks for directly reading documents. *arXiv:1606.03126*.

Neubig, G., and Dyer, C. 2016. Generalizing and hybridizing count-based and neural language models. *arXiv:1606.00499*.

Nguyen, T.; Rosenberg, M.; Song, X.; Gao, J.; Tiwary, S.; Majumder, R.; and Deng, L. 2016. Ms marco: A human generated machine reading comprehension dataset. *arXiv:1611.09268*.

Rajpurkar, P.; Zhang, J.; Lopyrev, K.; and Liang, P. 2016. Squad: 100,000+ questions for machine comprehension of text. *arXiv:1606.05250*.

Seo, M.; Kembhavi, A.; Farhadi, A.; and Hajishirzi, H. 2016. Bidirectional attention flow for machine comprehension. *arXiv:1611.01603*.

Vinyals, O.; Bengio, S.; and Kudlur, M. 2015. Order matters: Sequence to sequence for sets. *arXiv:1511.06391*.

Vrandečić, D., and Krötzsch, M. 2014. Wikidata: a free collaborative knowledgebase. *Communications of the ACM* 57(10):78–85.

Wang, T., and Cho, K. 2015. Larger-context language modelling. *arXiv:1511.03729*.

Wang, S., and Jiang, J. 2016. Machine comprehension using match-lstm and answer pointer. *arXiv:1608.07905*.

Wang, S.; Yu, M.; Guo, X.; Wang, Z.; Klinger, T.; Zhang, W.; Chang, S.; Tesauro, G.; Zhou, B.; and Jiang, J. 2017a. R ˆ3: Reinforced reader-ranker for open-domain question answering. *arXiv:1709.00023*.

Wang, S.; Yu, M.; Jiang, J.; Zhang, W.; Guo, X.; Chang, S.; Wang, Z.; Klinger, T.; Tesauro, G.; and Campbell, M. 2017b. Evidence aggregation for answer re-ranking in open-domain question answering. *arXiv:1711.05116*.

Xiong, C.; Zhong, V.; and Socher, R. 2016. Dynamic coattention networks for question answering. *arXiv:1611.01604*.

Xu, K.; Feng, Y.; Huang, S.; and Zhao, D. 2016a. Hybrid question answering over knowledge base and free text. In *COLING*.

Xu, K.; Reddy, S.; Feng, Y.; Huang, S.; and Zhao, D. 2016b. Question answering on freebase via relation extraction and textual evidence. *arXiv:1603.00957*.

Yang, B., and Mitchell, T. 2017. Leveraging knowledge bases in lstms for improving machine reading. In *ACL*.

Yih, W.-t.; Chang, M.-W.; He, X.; and Gao, J. 2015. Semantic parsing via staged query graph generation: Question answering with knowledge base. In *ACL*.

Yu, M.; Yin, W.; Hasan, K. S.; Santos, C. d.; Xiang, B.; and Zhou, B. 2017. Improved neural relation detection for knowledge base question answering. *arXiv:1704.06194*.