A Deep Architecture for Semantic Parsing

Edward Grefenstette, Phil Blunsom, Nando de Freitas and **Karl Moritz Hermann**
Department of Computer Science
University of Oxford, UK
{edwgre, pblunsom, nando, karher}@cs.ox.ac.uk

Abstract

Many successful approaches to semantic parsing build on top of the syntactic analysis of text, and make use of distributional representations or statistical models to match parses to ontology-specific queries. This paper presents a novel deep learning architecture which provides a semantic parsing system through the union of two neural models of language semantics. It allows for the generation of ontology-specific queries from natural language statements and questions without the need for parsing, which makes it especially suitable to grammatically malformed or syntactically atypical text, such as tweets, as well as permitting the development of semantic parsers for resource-poor languages.

1 Introduction

The ubiquity of always-online computers in the form of smartphones, tablets, and notebooks has boosted the demand for effective question answering systems. This is exemplified by the growing popularity of products like Apple's Siri or Google's Google Now services. In turn, this creates the need for increasingly sophisticated methods for semantic parsing. Recent work (Artzi and Zettlemoyer, 2013; Kwiatkowski et al., 2013; Matuszek et al., 2012; Liang et al., 2011, *inter alia*) has answered this call by progressively moving away from strictly rule-based semantic parsing, towards the use of distributed representations in conjunction with traditional grammatically-motivated re-write rules. This paper seeks to extend this line of thinking to its logical conclusion, by providing the first (to our knowledge) entirely distributed neural semantic generative parsing model. It does so by adapting deep learning methods from related work in sentiment analysis (Socher et al., 2012; Hermann and Blunsom, 2013), document classification (Yih et al., 2011; Lauly et al., 2014; Hermann and Blunsom, 2014a), frame-semantic parsing (Hermann et al., 2014), and machine translation (Mikolov et al., 2010; Kalchbrenner and Blunsom, 2013a), *inter alia*, combining two empirically successful deep learning models to form a new architecture for semantic parsing.

The structure of this short paper is as follows. We first provide a brief overview of the background literature this model builds on in §2. In §3, we begin by introducing two deep learning models with different aims, namely the joint learning of embeddings in parallel corpora, and the generation of strings of a language conditioned on a latent variable, respectively. We then discuss how both models can be combined and jointly trained to form a deep learning model supporting the generation of knowledgebase queries from natural language questions. Finally, in §4 we conclude by discussing planned experiments and the data requirements to effectively train this model.

2 Background

Semantic parsing describes a task within the larger field of natural language understanding. Within computational linguistics, semantic parsing is typically understood to be the task of mapping natural language sentences to formal representations of their underlying meaning. This semantic representation varies significantly depending on the task context. For instance, semantic parsing has been applied to interpreting movement instructions (Artzi and Zettlemoyer, 2013) or robot control (Matuszek et al., 2012), where the underlying representation would consist of actions.

Within the context of question answering—the focus of this paper—semantic parsing typically aims to map natural language to database queries that would answer a given question. Kwiatkowski

Proceedings of the ACL 2014 Workshop on Semantic Parsing, pages 22–27,
Baltimore, Maryland USA, June 26 2014.

et al. (2013) approach this problem using a multi-step model. First, they use a CCG-like parser to convert natural language into an underspecified logical form (ULF). Second, the ULF is converted into a specified form (here a FreeBase query), which can be used to lookup the answer to the given natural language question.

3 Model Description

We describe a semantic-parsing model that learns to derive quasi-logical database queries from natural language. The model follows the structure of Kwiatkowski et al. (2013), but relies on a series of neural networks and distributed representations in lieu of the CCG and λ-Calculus based representations used in that paper.

The model described here borrows heavily from two approaches in the deep learning literature. First, a noise-contrastive neural network similar to that of Hermann and Blunsom (2014a, 2014b) is used to learn a joint latent representation for natural language and database queries (§3.1). Second, we employ a structured conditional neural language model in §3.2 to generate queries given such latent representations. Below we provide the necessary background on these two components, before introducing the combined model and describing its learning setup.

3.1 Bilingual Compositional Sentence Models

The bilingual compositional sentence model (BiCVM) of Hermann and Blunsom (2014a) provides a state-of-the-art method for learning semantically informative distributed representations for sentences of language pairs from parallel corpora. Through the joint production of a shared latent representation for semantically aligned sentence pairs, it optimises sentence embeddings so that the respective representations of dissimilar cross-lingual sentence pairs will be weakly aligned, while those of similar sentence pairs will be strongly aligned. Both the ability to jointly learn sentence embeddings, and to produce latent shared representations, will be relevant to our semantic parsing pipeline.

The BiCVM model shown in Fig. 1 assumes vector composition functions g and h, which map an ordered set of vectors (here, word embeddings from $\mathcal{D}_A, \mathcal{D}_B$) onto a single vector in $\mathbb{R}^n$. As stated above, for semantically equivalent sentences a, b across languages $\mathcal{L}_A, \mathcal{L}_B$, the model aims to minimise the distance between these composed representations:

$$E_{bi}(a, b) = \|g(a) - h(b)\|^2$$

In order to avoid strong alignment between dissimilar cross-lingual sentence pairs, this error is combined with a noise-contrastive hinge loss, where $n \in \mathcal{L}_B$ is a randomly sampled sentence, dissimilar to the parallel pair $\{a, b\}$, and m denotes some margin:

$$E_{hl}(a, b, n) = [m + E_{bi}(a, b) - E_{bi}(a, n)]_+ ,$$

where $[x]_+ = max(0, x)$. The resulting objective function is as follows

$$J(\theta) = \sum_{(a,b)\in\mathcal{C}} \left(\sum_{i=1}^{k} E_{hl}(a, b, n_i) + \frac{\lambda}{2}\|\theta\|^2 \right),$$

with $\frac{\lambda}{2}\|\theta\|^2$ as the L_2 regularization term and $\theta{=}\{g, h, \mathcal{D}_A, \mathcal{D}_B\}$ as the set of model variables.

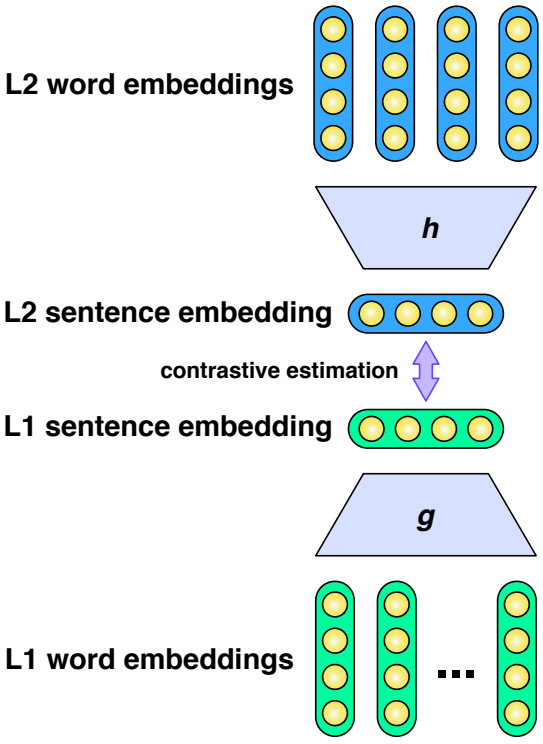

Figure 1: Diagrammatic representation of a BiCVM.

While Hermann and Blunsom (2014a) applied this model only to parallel corpora of sentences, it is important to note that the model is agnostic concerning the inputs of functions g and h. In this paper we will discuss how this model can be applied to non-sentential inputs.

3.2 Conditional Neural Language Models

Neural language models (Bengio et al., 2006) provide a distributed alternative to n-gram language models, permitting the joint learning of a prediction function for the next word in a sequence given the distributed representations of a subset of the last $n-1$ words alongside the representations themselves. Recent work in dialogue act labelling (Kalchbrenner and Blunsom, 2013b) and in machine translation (Kalchbrenner and Blunsom, 2013a) has demonstrated that a particular kind of neural language model based on recurrent neural networks (Mikolov et al., 2010; Sutskever et al., 2011) could be extended so that the next word in a sequence is jointly generated by the word history and the distributed representation for a conditioning element, such as the dialogue class of a previous sentence, or the vector representation of a source sentence. In this section, we briefly describe a general formulation of conditional neural language models, based on the log-bilinear models of Mnih and Hinton (2007) due to their relative simplicity.

A log-bilinear language model is a neural network modelling a probability distribution over the next word in a sequence given the previous $n-1$, i.e. $p(w_n|w_{1:n-1})$. Let $|V|$ be the size of our vocabulary, and R be a $|V| \times d$ vocabulary matrix where the R_{w_i} demnotes the row containing the word embedding in $\mathbb{R}^d$ of a word w_i, with d being a hyper-parameter indicating embedding size. Let C_i be the context transform matrix in $\mathbb{R}^{d \times d}$ which modifies the representation of the ith word in the word history. Let b_{w_i} be a scalar bias associated with a word w_i, and b_R be a bias vector in $\mathbb{R}^d$ associated with the model. A log-bilinear model expressed the probability of w_n given a history of $n-1$ words as a function of the energy of the network:

$$E(w_n; w_{1:n-1}) = -\left(\sum_{i=1}^{n-1} R_{w_i}^T C_i\right) R_{w_n} - b_R^T R_{w_n} - b_{w_n}$$

From this, the probability distribution over the next word is obtained:

$$p(w_n|w_{1:n-1}) = \frac{e^{-E(w_n;w_{1:n-1})}}{\sum_{w_n} e^{-E(w_n;w_{1:n-1})}}$$

To reframe a log-bilinear language model as a conditional language model (CNLM), illustrated in Fig. 2, let us suppose that we wish to jointly condition the next word on its history and some variable β, for which an embedding r_β has been obtained through a previous step, in order to compute $p(w_n|w_{1:n-1}, \beta)$. The simplest way to do this additively, which allows us to treat the contribution of the embedding for β as similar to that of an extra word in the history. We define a new energy function:

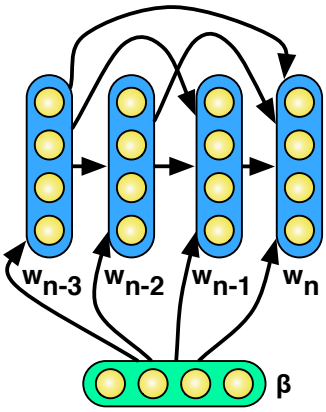

Figure 2: Diagrammatic representation of a Conditional Neural Language Model.

$$E(w_n; w_{1:n-1}, \beta) = -\left(\left(\sum_{i=1}^{n-1} R_{w_i}^T C_i\right) + r_\beta^T C_\beta\right) R_{w_n} - b_R^T R_{w_n} - b_{w_n}$$

to obtain the probability

$$p(w_n|w_{1:n-1}, \beta) = \frac{e^{-E(w_n;w_{1:n-1},\beta)}}{\sum_{w_n} e^{-E(w_n;w_{1:n-1},\beta)}}$$

Log-bilinear language models and their conditional variants alike are typically trained by maximising the log-probability of observed sequences.

3.3 A Combined Semantic Parsing Model

The models in §§3.1–3.2 can be combined to form a model capable of jointly learning a shared latent representation for question/query pairs using a BiCVM, and using this latent representation to learn a conditional log-bilinear CNLM. The full model is shown in Fig. 3. Here, we explain the final model architecture both for training and for subsequent use as a generative model. The details of the training procedure will be discussed in §3.4.

The combination is fairly straightforward, and happens in two steps at training time. For the

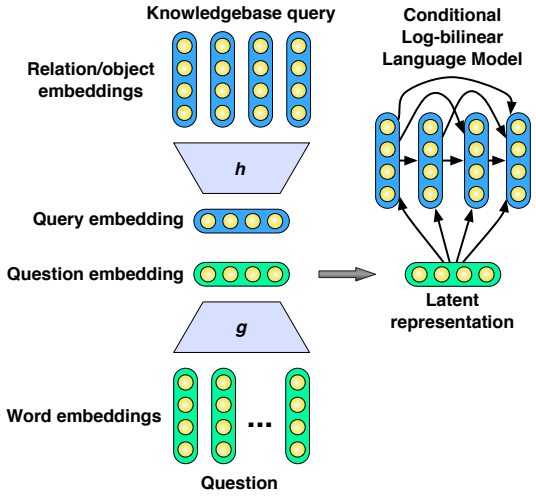

Figure 3: Diagrammatic representation of the full model. First the mappings for obtaining latent forms of questions and queries are jointly learned through a BiCVM. The latent form for questions then serves as conditioning element in a log-bilinear CNLM.

first step, shown in the left hand side of Fig. 3, a BiCVM is trained against a parallel corpora of natural language question and knowledgebase query pairs. Optionally, the embeddings for the query symbol representations and question words are initialised and/or fine-tuned during training, as discussed in §3.4. For the natural language side of the model, the composition function g can be a simple additive model as in Hermann and Blunsom (2014a), although the semantic information required for the task proposed here would probably benefit from a more complex composition function such as a convolution neural network. Function h, which maps the knowledgebase queries into the shared space could also rely on convolution, although the structure of the database queries might favour a setup relying primarily on bi-gram composition.

Using function g and the original training data, the training data for the second stage is created by obtaining the latent representation for the questions of the original dataset. We thereby obtain pairs of aligned latent question representations and knowledgebase queries. This data allows us to train a log-bilinear CNLM as shown on the right side of Fig. 3.

Once trained, the models can be fully joined to produce a generative neural network as shown in Fig. 4. The network modelling g from the BiCVM

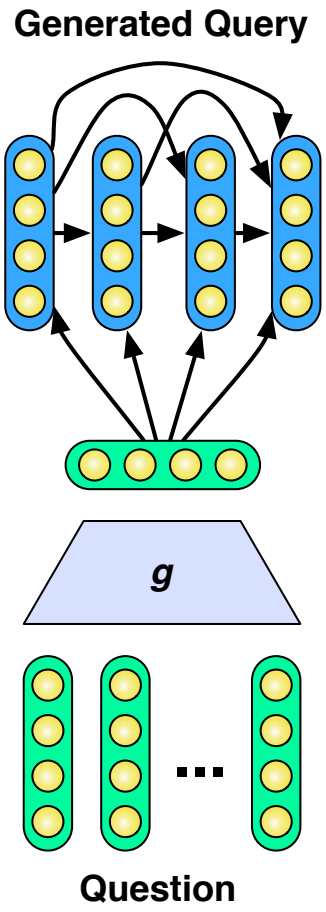

Figure 4: Diagrammatic representation of the final network. The question-compositional segment of the BiCVM produces a latent representation, conditioning a CNLM generating a query.

takes the distributed representations of question words from unseen questions, and produces a latent representation. The latent representation is then passed to the log-bilinear CNLM, which conditionally generates a knowledgebase query corresponding to the question.

3.4 Learning Model Parameters

We propose training the model of §3.3 in a two stage process, in line with the symbolic model of Kwiatkowski et al. (2013).

First, a BiCVM is trained on a parallel corpus C of question-query pairs $\langle Q, R \rangle \in C$, using composition functions g for natural language questions and h for database queries. While functions g and h may differ from those discussed in Hermann and Blunsom (2014a), the basic noise-contrastive optimisation function remains the same. It is possible to initialise the model fully randomly, in which

case the model parameters θ learned at this stage include the two distributed representation lexica for questions and queries, $\mathcal{D}_Q$ and $\mathcal{D}_R$ respectively, as well as all parameters for g and h.

Alternatively, word embeddings in $\mathcal{D}_Q$ could be initialised with representations learned separately, for instance with a neural language model or a similar system (Mikolov et al., 2010; Turian et al., 2010; Collobert et al., 2011, *inter alia*). Likewise, the relation and object embeddings in $\mathcal{D}_R$ could be initialised with representations learned from distributed relation extraction schemas such as that of Riedel et al. (2013).

Having learned representations for queries in $\mathcal{D}_R$ as well as function g, the second training phase of the model uses a new parallel corpus consisting of pairs $\langle g(Q), R\rangle \in C'$ to train the CNLM as presented in §3.3.

The two training steps can be applied iteratively, and further, it is trivial to modify the learning procedure to use composition function h as another input for the CNLM training phrase in an autoencoder-like setup.

4 Experimental Requirements and Further Work

The particular training procedure for the model described in this paper requires aligned question/knowledgebase query pairs. There exist some small corpora that could be used for this task (Zelle and Mooney, 1996; Cai and Yates, 2013). In order to scale training beyond these small corpora, we hypothesise that larger amounts of (potentially noisy) training data could be obtained using a boot-strapping technique similar to Kwiatkowski et al. (2013).

To evaluate this model, we will follow the experimental setup of Kwiatkowski et al. (2013). With the provisio that the model can generate freebase queries correctly, further work will seek to determine whether this architecture can generate other structured formal language expressions, such as lambda expressions for use in textual entailement tasks.

Acknowledgements

This work was supported by a Xerox Foundation Award, EPSRC grants number EP/I03808X/1 and EP/K036580/1, and the Canadian Institute for Advanced Research (CIFAR) Program on Adaptive Perception and Neural Computation.

References

Yoav Artzi and Luke Zettlemoyer. 2013. Weakly supervised learning of semantic parsers for mapping instructions to actions. *Transactions of the Association for Computational Linguistics*, 1(1):49–62.

Yoshua Bengio, Holger Schwenk, Jean-Sébastien Senécal, Fréderic Morin, and Jean-Luc Gauvain. 2006. Neural probabilistic language models. In *Innovations in Machine Learning*, pages 137–186. Springer.

Qingqing Cai and Alexander Yates. 2013. Large-scale Semantic Parsing via Schema Matching and Lexicon Extension. In *Proceedings of the Annual Meeting of the Association for Computational Linguistics (ACL)*.

Ronan Collobert, Jason Weston, Leon Bottou, Michael Karlen, Koray Kavukcuoglu, and Pavel Kuksa. 2011. Natural language processing (almost) from scratch. *Journal of Machine Learning Research*, 12:2493–2537.

Karl Moritz Hermann and Phil Blunsom. 2013. The Role of Syntax in Vector Space Models of Compositional Semantics. In *Proceedings of the 51st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, Sofia, Bulgaria, August. Association for Computational Linguistics.

Karl Moritz Hermann and Phil Blunsom. 2014a. Multilingual Distributed Representations without Word Alignment. In *Proceedings of the 2nd International Conference on Learning Representations*, Banff, Canada, April.

Karl Moritz Hermann and Phil Blunsom. 2014b. Multilingual Models for Compositional Distributional Semantics. In *Proceedings of the 52nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, Baltimore, USA, June. Association for Computational Linguistics.

Karl Moritz Hermann, Dipanjan Das, Jason Weston, and Kuzman Ganchev. 2014. Semantic Frame Identification with Distributed Word Representations. In *Proceedings of the 52nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, Baltimore, USA, June. Association for Computational Linguistics.

Nal Kalchbrenner and Phil Blunsom. 2013a. Recurrent continuous translation models. In *Proceedings of the 2013 Conference on Empirical Methods in Natural Language Processing (EMNLP), Seattle, USA. Association for Computational Linguistics*.

Nal Kalchbrenner and Phil Blunsom. 2013b. Recurrent convolutional neural networks for discourse compositionality. *arXiv preprint arXiv:1306.3584*.

Tom Kwiatkowski, Eunsol Choi, Yoav Artzi, and Luke Zettlemoyer. 2013. Scaling semantic parsers with on-the-fly ontology matching. In *Proceedings of*

the 2013 Conference on Empirical Methods in Natural Language Processing, pages 1545–1556, Seattle, Washington, USA, October. Association for Computational Linguistics.

Stanislas Lauly, Alex Boulanger, and Hugo Larochelle. 2014. Learning multilingual word representations using a bag-of-words autoencoder. *CoRR*, abs/1401.1803.

Percy Liang, Michael I. Jordan, and Dan Klein. 2011. Learning dependency-based compositional semantics. In *Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies - Volume 1*, HLT '11, pages 590–599, Stroudsburg, PA, USA. Association for Computational Linguistics.

Cynthia Matuszek, Nicholas FitzGerald, Luke S. Zettlemoyer, Liefeng Bo, and Dieter Fox. 2012. A joint model of language and perception for grounded attribute learning. In *Proceedings of the 29th International Conference on Machine Learning, ICML 2012, Edinburgh, Scotland, UK, June 26 - July 1, 2012*.

Tomas Mikolov, Martin Karafiát, Lukas Burget, Jan Cernockỳ, and Sanjeev Khudanpur. 2010. Recurrent neural network based language model. In *INTERSPEECH*, pages 1045–1048.

Andriy Mnih and Geoffrey Hinton. 2007. Three new graphical models for statistical language modelling. In *Proceedings of the 24th international conference on Machine learning*, pages 641–648. ACM.

Sebastian Riedel, Limin Yao, Benjamin M. Marlin, and Andrew McCallum. 2013. Relation extraction with matrix factorization and universal schemas. In *Joint Human Language Technology Conference/Annual Meeting of the North American Chapter of the Association for Computational Linguistics (HLT-NAACL '13)*, June.

Richard Socher, Brody Huval, Christopher D. Manning, and Andrew Y. Ng. 2012. Semantic compositionality through recursive matrix-vector spaces. In *Proceedings of EMNLP-CoNLL*, pages 1201–1211.

Ilya Sutskever, James Martens, and Geoffrey E Hinton. 2011. Generating text with recurrent neural networks. In *Proceedings of the 28th International Conference on Machine Learning (ICML-11)*, pages 1017–1024.

Joseph Turian, Lev Ratinov, and Yoshua Bengio. 2010. Word representations: A simple and general method for semi-supervised learning. In *Proceedings of ACL*, Stroudsburg, PA, USA.

Wen-Tau Yih, Kristina Toutanova, John C. Platt, and Christopher Meek. 2011. Learning Discriminative Projections for Text Similarity Measures. In *Proceedings of the Fifteenth Conference on Computational Natural Language Learning*, CoNLL '11, pages 247–256, Stroudsburg, PA, USA. Association for Computational Linguistics.

John M. Zelle and Raymond J. Mooney. 1996. Learning to parse database queries using inductive logic programming. In *Proceedings of the National Conference on Artificial Intelligence*, pages 1050–1055.