Cross-lingual Learning of an Open-domain Semantic Parser

Kilian Evang
University of Groningen
The Netherlands
`k.evang@rug.nl`

Johan Bos
University of Groningen
The Netherlands
`johan.bos@rug.nl`

Abstract

We propose a method for learning semantic CCG parsers by projecting annotations via a parallel corpus. The method opens an avenue towards cheaply creating multilingual semantic parsers mapping open-domain text to formal meaning representations. A first cross-lingually learned Dutch (from English) semantic parser obtains f-scores ranging from 42.99% to 69.22% depending on the level of label informativity taken into account, compared to 58.40% to 78.88% for the underlying source-language system. These are promising numbers compared to state-of-the-art semantic parsing in open domains.

1 Introduction

Scarceness of manually annotated corpora for training dependency parsers has led researchers to explore more indirect forms of supervision, such as cross-lingual learning, where annotations in one language are used in training a system for another language. Semantic parsers, which map sentences directly to logically interpretable meaning representations, equally suffer from a lack of annotated training corpora, despite recent efforts like the Groningen Meaning Bank (Basile et al., 2012) or AMR Bank (Banarescu et al., 2013). The lack is especially pronounced for languages other than English.

   This paper aims to show that cross-lingual learning can help create semantic parsers for new languages with little knowledge about those languages and minimal human intervention. We present a method that takes an existing (source-language) semantic parser and parallel data and learns a semantic parser for the target language. Our method is in principle applicable to all parsers producing interpreted derivations (i.e., parse trees) of Combinatory Categorial Grammar (CCG; Steedman 2001) . It is independent of the concrete meaning representation formalism used, as long as meaning representations are assembled in the standard CCG way using the lambda calculus. We evaluate our method by applying it to English as source language, Dutch as target language and Discourse Representation Theory (DRT; Kamp and Reyle 1993) as meaning representation formalism, and measuring the performance of the obtained Dutch semantic parser.

2 Related Work

Semantic parsing has been tackled from a wide variety of angles. Systems that add a semantic interpretation component to an existing supervised syntactic parser (Curran et al., 2007; Le and Zuidema, 2012; Lewis and Steedman, 2013) have wide coverage but require much syntactically annotated training data. Other approaches are restricted to relatively narrow linguistic domains but manage to do without strong syntactic supervision. Forms of supervision used include sentence/meaning representation pairs (Wong and Mooney, 2007; Zettlemoyer and Collins, 2007) and even weaker forms of supervision (Clarke et al., 2010; Liang et al., 2011; Kwiatkowski et al., 2013; Goldwasser and Roth, 2011; Chen and Mooney, 2011; Krishnamurthy and Mitchell, 2012; Reddy et al., 2014; Artzi and Zettlemoyer, 2011; Poon, 2013). Only recently have approaches not relying on explicit syntactic supervision successfully been applied to
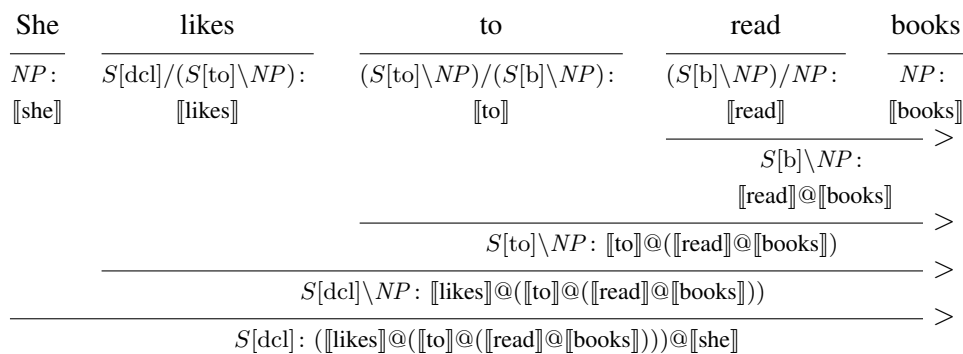
She	likes	to	read	books
NP:	$S[\text{dcl}]/(S[\text{to}]\backslash NP)$:	$(S[\text{to}]\backslash NP)/(S[\text{b}]\backslash NP)$:	$(S[\text{b}]\backslash NP)/NP$:	NP:
$[\![\text{she}]\!]$	$[\![\text{likes}]\!]$	$[\![\text{to}]\!]$	$[\![\text{read}]\!]$	$[\![\text{books}]\!]$

$$S[\text{b}]\backslash NP:\ [\![\text{read}]\!]@[\![\text{books}]\!] \quad >$$

$$S[\text{to}]\backslash NP:\ [\![\text{to}]\!]@([\![\text{read}]\!]@[\![\text{books}]\!]) \quad >$$

$$S[\text{dcl}]\backslash NP:\ [\![\text{likes}]\!]@([\![\text{to}]\!]@([\![\text{read}]\!]@[\![\text{books}]\!])) \quad >$$

$$S[\text{dcl}]:\ ([\![\text{likes}]\!]@([\![\text{to}]\!]@([\![\text{read}]\!]@[\![\text{books}]\!])))@[\![\text{she}]\!] \quad >$$

Figure 1: Example CCG derivation.

x1 p1 e1

female(x1)
like.v.02(e1)
Experiencer(e1, x1)
Stimulus(e1, p1)

p1:

x2 e2

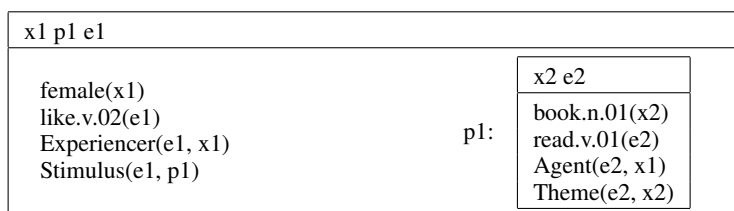book.n.01(x2)
read.v.01(e2)
Agent(e2, x1)
Theme(e2, x2)

Figure 2: Example DRS for the sentence in Figure 1.

more open-domain sentences (Vanderwende et al., 2015; Artzi et al., 2015). Ours is, to the best of our knowledge, the first such work using cross-lingual learning.

Cross-lingual learning has previously been applied to different NLP tasks, notably part-of-speech tagging and dependency parsing. For dependency parsing, the task most similar to ours, three families of approaches can be distinguished. In **annotation projection**, existing annotations of source-language text are automatically projected to target-language translations in a parallel corpus; the result is used to train a target-language system (Hwa et al., 2005; Tiedemann, 2014; Rasooli and Collins, 2015; Johannsen et al., 2016; Agić et al., 2016). In **model transfer**, parsers for different languages share some of their model parameters, thereby using information from annotations in multiple languages at the same time. (Zeman and Resnik, 2008; Ganchev et al., 2009; McDonald et al., 2011; Naseem et al., 2012; Täckström et al., 2013). The **translation approach** pioneered by Tiedemann et al. (2014) is similar to annotation projection, but instead of relying on existing translations, it automatically translates the data and synchronously projects annotations to the translation result. Our approach falls within the annotation projection family, with the new challenge that entire CCG derivations with logical interpretations need to be transferred.

3 Combinatory Categorial Grammar

Combinatory Categorial Grammar (Steedman, 2001) is a grammar formalism widely used for semantic parsing due to its suitability to statistical parsing (Clark and Curran, 2007) and its transparent syntax-semantics interface. Every constituent has a *category*—either a basic one (S for sentence, N for noun, NP for noun phrase, PP for prepositional argument) or a functional one such as $S\backslash NP$ for verb phrase, indicating that a constituent can combine with a noun phrase to its left to yield a sentence. Smaller constituents are combined into larger ones according to a handful of combinatorial rules such as application and composition. Every constituent also has a semantics, its *interpretation*, which is a term of the lambda calculus. Crucially, the combinatorial rules specify precisely how the interpretation of each non-lexical constituent is computed from the interpretations of the constituents that combine to form it. An example derivation (CCG parse tree) for an English sentence is shown in Figure 1.

The lambda calculus and thus CCG is applicable to any kind of meaning representation, as long as it can be constructed compositionally. In this paper, we use a flavor of Discourse Representation Theory (DRT; Kamp and Reyle 1993) and a corresponding semantic lexicon introduced by Bos (2009) which

$$\llbracket\text{likes}\rrbracket = \quad \lambda t.\lambda s.\lambda m.(s@\lambda x.((t@\lambda y.(y@x))@\lambda z.(\;\boxed{\begin{array}{l} e \\ \hline \text{like.v.01}(e) \\ \text{Experiencer}(e, x) \\ \text{Stimulus}(e, z) \end{array}}\; + (m@e))))$$

$$\llbracket\text{to}\rrbracket = \quad \lambda b.\lambda x.\lambda m.(\;\boxed{\begin{array}{l} p \\ \hline p{:}((b@x)@\lambda y.\;\boxed{}\;) \end{array}}\; + (m@p))$$

$$\llbracket\text{graag}\rrbracket = \lambda x.\llbracket\text{likes}\rrbracket@(\llbracket\text{to}\rrbracket@x) = \lambda b.\lambda x.\lambda m.(y@\lambda x.(\;\boxed{\begin{array}{l} e\ p \\ \hline \text{like.v.01}(e) \\ \text{Experiencer}(e, x) \\ \text{Stimulus}(e, z) \\ \\ p{:}((b@\lambda i.(i@x))@\lambda j.\;\boxed{}\;) \end{array}}\; + (m@e)))$$

Figure 3: Examples of some lexical interpretations (two English, one Dutch).

uses a neo-Davidsonian event semantics with VerbNet/LIRICS semantic roles (Bonial et al., 2011) and WordNet 3.0 senses (Fellbaum, 1998). The meaning representation (discourse representation structure; DRS) for our example sentence is shown in Figure 2. Lexical interpretations of some words are shown in Figure 3.

4 Method

We start with a parallel corpus of sentence pairs whose source-language part has been annotated with semantic CCG derivations as in Figure 1 by the source-language system. We use this annotation in two ways: first, to induce a target-language lexicon in a first step called **category projection**. Secondly, we use it as a form of indirect supervision: we assume that the source-language system works mostly correctly, and that if two sentences are translations of each other, they should have the same interpretation—thus we can train the target-language parser to produce the same interpretations as the source-language parser. To this end, we try to find target-language derivations resulting in the same interpretations as the source-language ones, based on the target-language candidate lexical items found in category projection. We call this second step **derivation projection**. The derivations thus found are used to train a statistical parsing model for the target language. We call this third step **parser learning**.

All three steps make use of a shift-reduce CCG parser similar to that of Zhang and Clark (2011). Parse actions are SHIFT-C-I, COMBINE-C, UNARY-C (where C is the category placed on top of the stack by shifting or applying a binary/unary rule and I is the interpretation of the (multi)word placed on top of the stack by shifting), SKIP for skipping words as semantically empty, FINISH for marking a parse as complete and IDLE for keeping complete parses on the agenda while others are still incomplete (Zhu et al., 2013).

We now describe the three steps in more detail.

4.1 Step 1: Category Projection

Category projection assigns candidate categories and interpretations to target-language (multi)words in the training data. It thereby also induces the target-language lexicon that we use in subsequent steps. It serves as a cross-lingual alternative to the two traditional main strategies of inducing CCG lexicons for semantic parsing, namely hand-written, language-specific lexical templates (Zettlemoyer and Collins, 2005) and higher-order unification constrained by search heuristics (Kwiatkowksi et al., 2010).

Category projection first word-aligns the training corpus—we use the n best alignments found by GIZA++ (Och and Ney, 2003) with default settings. The result is a large, noisy set of translation units. From each contiguous translation unit, we try to induce a candidate lexical item. Figure 4 shows an
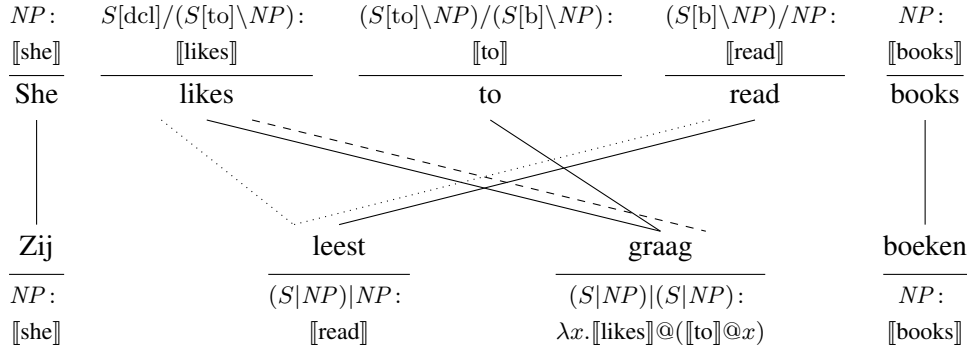
Figure 4: Category projection: word alignments induce candidate categories and interpretations for target-language words.

example sentence pair: at the top there are the lexical items from the English derivation in Figure 1, each with a CCG category and interpretation. The interpretations are here represented using the $[\![\cdot]\!]$ notation but are actually a complex λ-terms with DRT-based meaning representations (for short: a λ-DRS) as exemplified in Figure 3. The lines in the center represent possible word alignments, with correct translation units drawn as solid lines and incorrect ones as dashed or dotted ones. At the bottom there is the Dutch sentence with induced candidate lexical items. For the sake of the example, we only show one candidate lexical item per word, those induced from the correct translation units.

Inducing a candidate lexical item from a translation unit works as follows: if the translation unit contains only one source-language word, it provides the category and interpretation for the (multi)word on the target side, as is the case for *She/Zij*, *read/leest* and *books/boeken*. Since slash directions are language-specific, we change all categories to have vertical slashes, which can apply in either direction. We also remove English-specific category features such as $[b]$ and $[dcl]$, distinguishing bare and declarative verb phrases. For example, $(S[b]\backslash NP)/NP$ becomes $(S|NP)|NP$.

If the translation unit contains more than one source-language word, this string is parsed using CCG combinatory rules, and if successful, the resulting category and interpretation are used as a lexical item for the word on the target side. For example, the verb *likes* and the particle *to* combine via forward composition into one category and interpretation for the Dutch adverb *graag*, which expresses the same meaning in a syntactically different way. The full resulting λ-DRS for *graag* is shown in Figure 3.

Unaligned target words get a special category *skip*.

The final target-language lexicon $\mathcal{L}$ contains the lexical items thus induced, but only those that are at least a cutoff factor c times as frequent as the most frequent candidate for this target-language word/part-of-speech combination.

4.2 Step 2: Derivation Projection

The lexical items assigned to target-language words in category projection give rise to a space of possible CCG derivations. The space is large and noisy, partly because of the pervasive syntactic ambiguity of natural language, partly because we use more than one word alignment in category projection. In derivation projection, the task is to filter out only the "correct" derivations so we can then train on these. We regard as "correct" any derivation that results in the same interpretation for the whole sentence as the source-language derivation.

For finding the "correct" derivations, we use the method of Zhao and Huang (2015) of running the parser in forced decoding mode: we use a beam of unlimited width but prune away parse items where, based on their interpretations, we can rule out that they could lead to a "correct" derivation. For instance, in our example, an item with interpretation $[\![read]\!]@[\![she]\!]$ would be pruned because it cannot be part of $([\![likes]\!]@([\![to]\!]@([\![read]\!]@[\![books]\!])))@[\![she]\!]$. To make this check tractable, we treat English lexical interpretations such as $[\![read]\!]$ as atomic.

The forced decoding uses a local lexicon, using only lexical items induced from the same sentence pair.

$$
\begin{array}{cccc}
\text{Zij} & \text{leest} & \text{graag} & \text{boeken} \\
\hline
NP: & (S|NP)|NP: & (S|NP)|(S|NP): & NP: \\
[\![\text{she}]\!] & [\![\text{read}]\!] & \lambda x.[\![\text{likes}]\!]@([\![\text{to}]\!]@x) & [\![\text{books}]\!]
\end{array}
$$

$$
\cfrac{\cfrac{(S|NP)|NP:}{\lambda x.[\![\text{likes}]\!]@([\![\text{to}]\!]@([\![\text{read}]\!]@x))}\ \mathbf{B}<_{\times}}{\cfrac{S|NP:\ [\![\text{likes}]\!]@([\![\text{to}]\!]@([\![\text{read}]\!]@[\![\text{books}]\!]))}{S:\ ([\![\text{likes}]\!]@([\![\text{to}]\!]@([\![\text{read}]\!]@[\![\text{books}]\!])))@[\![\text{she}]\!]}\ <}\ >
$$

Figure 5: Derivation projection: combinatory rules are applied to find a derivation with the same interpretation as the source-language sentence.

The combinatory rule instances used are extracted from all English training derivations, but to allow for different word orders, we verticalize all slashes and for binary rule instances add mirror-image versions, e.g., the backward application instance $NP\ S\backslash NP \Rightarrow S$ generates $NP\ S|NP \Rightarrow S$ and $S|NP\ NP \Rightarrow S$.

If we cannot find any "correct" derivation, this means we did not get the word alignments inducing the lexical items needed to find one. This can be due to translations being idiomatic, loose or informative (Bos, 2014). In such cases, our assumption that the interpretation for source and target sentence should be the same breaks down, and we would not want to use this training example anyway. In this sense, derivation projection also has the function of cleaning the training set.

Despite the pruning, for some sentences the search space is prohibitively large, so we restrict the size of the parser agenda to 256, a number that still allows us to run this step in reasonable time. If this limit is exceeded or if we do not find a complete derivation with the target interpretation, we discard the sentence. If we do find one or more—such as the one in Figure 5—the sentence becomes part of the training data for the following step.

4.3 Step 3: Parser Learning

For statistical parsing, we use an averaged perceptron with a hash kernel (Bohnet, 2010) and the same feature templates as Zhang and Clark (2011), plus, for shift actions, a feature uniquely identifying a lexical item including the (multi)word, its part(s) of speech and the chosen category and interpretation. The parser uses the full global lexicon $\mathcal{L}$. The same grammar as in derivation projection is used.

The parser uses beam search. If at some point during training on one example there is no item on the beam anymore that could lead to one of the "correct" derivations found in derivation projection, the parser aborts training on this example and performs an early perceptron update (Collins and Roark, 2004).

5 Experimental Setup

To ensure that derivation projection can find a large number of high-quality derivations, we need training data with a large proportion of "literally" translated sentences. By this we do not mean that the translation has to be syntactically isomorphic—our projection approach can actually deal with a wide range of such syntactic divergences (cf. Dorr, 1993), such as the *likes to/graag* example. But translations should not be informative or loose, as this changes their meaning. More literal translations than in freely occurring text can be found in resources aimed at human language learners (who are faced with a similar task as our system: learning to understand an unknown language, helped by example sentences translated into a familiar one). One such resource is tatoeba.org, based on the Tanaka corpus (Tanaka, 2001). We used 13,122 English-Dutch sentence pairs from Tatoeba as training data, 1,639 for development and 1,641 as final test set, of which a random sample of 150 sentences was manually annotated to serve as a gold standard.

In preliminary experiments, we tried out different values for n where we use the n-best alignments per direction to extract candidate lexical item from. Too low and derivation projection may not find a derivation for many target-language sentences for lack of needed candidate lexical entries. Too high and the agenda gets polluted with spurious parse items, and derivation projection aborts due to the agenda limit. We found that for our training set, the percentage of examples for which we find at least one target-language derivation peaked at $n = 3$ with 57.7%.

The source language system whose output we use for supervision is the C&C/Boxer system (Curran et al., 2007), which takes English sentences and produces discourse representation structures. We use SVN repository version 2444, giving the options `--modal true --nn true --roles verbnet` to Boxer and making some minor modifications to its code to better match our annotation scheme for adjectives, adverbs, semantic roles and modals.

For statistical parsing, we initialize all model weights to 0 and use a beam width of 16. The Dutch sentences are POS-tagged using TreeTagger (Schmid, 1995). For decoding, the input is only POS-tagged Dutch development/test sentences. We use the same lexicon as for training, but to deal with unseen content words, an abstract version of each lexical entry is created where the synset ID in its λ-DRS is replaced by the $\_\_UNKNOWN\_\_$ symbol. The parser then selects between all categories occurring with the POS tag, with the most common abstract interpretation for each category. The output is a CCG derivation—or, since the parser can fall back to fragmentary output, a sequence thereof—each of which has a DRS interpretation.

6   Evaluation Setup

For evaluation, we follow the approach proposed by Allen et al. (2008): meaning representations are converted to graphs and an alignment between system output and gold graph vertices is found that maximizes the number of labeled edges in a maximum common subgraph. An instantiation of this evaluation metric for Abstract Meaning Representations, SMATCH (Cai and Knight, 2013), is now commonly used. We use the instantiation for DRSs that was first introduced by Le and Zuidema (2012).

We first measure how closely the output of our system for Dutch resembles that of C&C/Boxer for English on the development/testing portion of our parallel corpus. This gives an idea of how well our system has learned to imitate the existing system, but has two problems: first, it does not say much about the quality of the output because that of C&C/Boxer is not free from errors, it is not a gold standard. Secondly, the data contains idiomatic, informative and loose translations, in which case we *want* the outputs of both systems to differ.

Therefore, we also measure how closely the outputs of C&C/Boxer and our system resemble a gold standard of 150 sentence/DRS pairs from the testing portion, for their respective input languages. Since DRSs are complex structures not easily created in completely manual annotation, we resorted to hand-correcting automatically produced ones to obtain the gold standard. This was done as follows: Two annotators independently corrected 50 DRSs produced by C&C/Boxer so that the DRSs represented the meaning of the Dutch annotations. Inter-annotator agreement at this point as measured by the evaluation metric was 67%. Instances of disagreement were identified, with 29% related to WordNet senses, 22% to semantic roles, 16% to other relations such as prepositional ones, 13% to the rendering of Dutch idioms using English WordNet senses, 9% to modal and logical operators such as implication and negation, and 11% to other structural issues such as nested DRSs. In an adjudication phase, both annotators resolved the differences together and agreed on a common gold standard. A single annotator then corrected another 100 Boxer DRSs, which were then checked by the other annotator, and differences were again resolved through discussion. One annotator finally created an adapted version of all 150 DRSs where in case of non-literal translations, the annotation matches the English rather than Dutch sentence.

No comparable systems for Dutch as input language and DRS as meaning representation language exist yet. To demonstrate the effect of learning the parsing model, we picked a simple baseline that assigns each target-language word the semantic representation most frequently associated with aligned English words and outputs the resulting, very fragmented graph.

Table 1: Development set (non-gold-standard) f-score depending on lexical cutoff factor c and training iterations T (i.e., number of passes over the entire training data).

n=3

lexical cutoff factor (c)	T=0	1	2	3	4	5	6	7	8	9	10
0.5	33.55	47.34	47.18	47.46	47.35	47.32	47.34	47.41	47.57	47.52	47.51
0.2	32.15	47.55	48.76	49.12	49.11	49.40	49.27	49.54	49.73	49.69	49.60
0.1	29.33	47.03	48.32	48.40	48.71	48.85	49.01	49.10	49.13	49.20	49.11
0.05	27.82	46.04	47.11	48.42	48.38	48.64	48.97	48.23	48.56	48.28	48.47
0.02	25.17	44.66	46.35	46.83	47.31	47.48	47.41	47.43	47.49	47.92	47.86
0.01	23.49	44.27	46.91	46.90	47.28	47.51	47.52	47.97	47.56	47.54	47.81

training iterations (T)

Table 2: Gold-standard match f-score for Boxer, our baseline and our best cross-lingually trained model.

Language	English	Dutch	
System	C&C/Boxer	Baseline	Our system
Full	58.40	26.71	42.99
Ignoring WordNet senses	69.06	36.67	60.23
Ignoring VerbNet/LIRICS roles	64.51	27.57	47.82
Ignoring other relation labels	59.18	27.57	43.39
Ignoring all	78.88	39.04	69.22

7 Results and Discussion

Table 1 shows how the f-score of our system on the (non-gold-stard because automatically annotated) development corpus varies as a function of the lexical cutoff factor c and number of training iterations T. We used the model with the highest score ($c = 0.1$, $T = 10$) for final testing. Table 2 shows the results, comparing the performance of our cross-lingually learned system on Dutch against the baseline and against C&C/Boxer's performance on the English versions of the same sentences.

C&C/Boxer obtains an f-score of 58.40% on the gold standard. Although the data and the formalism are not directly comparable, we note that this f-score is close to those of current state-of-the-art open-domain semantic parsers for English, e.g. those that participated in the recent Abstract Meaning Representation shared task (May, 2016). A large part of the errors comes from misidentifying word senses and semantic roles. "Sloppy" evaluations in which we treat all word senses, all roles and/or other (e.g. prepositional) relation labels as equal give markedly higher f-scores of up to 78.88%.

Our system for Dutch scores around 15% lower than the source-language system under the strict evaluation, at 42.99%. The gap narrows to around 10% under the sloppy evaluation, scoring 69.22%. The gap is expected for a number of reasons. For one, the English system has the advantage of a strong syntactic parser which was trained on a far larger number of sentences, which also had explicit syntactic annotation. The especially large gap under the strict evaluation can partially be explained by many unseen words in the test data, with the training data insufficiently large to learn a wide-coverage lexicon, while the system for English has access to the full WordNet lexicon.

For languages like Dutch, available resources could be exploited to address these problems. For example, one could improve a cross-lingually bootstrapped CCG parser by training it to recover the dependencies in a dependency treebank, e.g. Universal Dependencies (Nivre et al., 2016). Multilingual lexical databases like Open Multilingual Wordnet (Bond and Foster, 2013) could be exploited to attack unseen words. For truly low-resource languages where such resources are not available, parallel data could be mined in order to extend the target-language lexicon. This could work even with data that is currently too loosely translated or too syntactically complex to work well with our projection approach.

Although we optimized the hyperparameter $n = 3$ for the number of successfully projected derivations, Dutch derivations were found for only 58.35% of our training sentences, considerably reducing the amount of training data that is available for parser learning. To what extent this is due to non-literal translations being weeded out (cf. Section **??**), and to what extent failing derivation projections could be avoided (e.g. by considering other combinatory rules than those extracted from the English data) is an important question for future work.

8   Conclusions

Semantic parsing for open domains is a young and very dynamic research area that may shortly enable computers to make use of natural language on a new and significantly deeper level. With a field notoriously focused on English, how can other languages keep up with the developments?

In this paper, we have shown a possible avenue. We draw upon CCG's flexible notion of constituency and the language-independent nature of its combinatory rules to develop a lexicon induction technique that overcomes certain translation divergences between languages. We have then used cross-lingual supervision to train a semantic parser for Dutch at a far lower cost than the original English one, considering the cost of manually creating explicit syntactic annotation and a semantic lexicon.

Bridging the gap between source and target language does come at an additional cost in performance. However, there are a number of possible ways to attack this gap in future work, including using target-language lexical resources if available, unsupervised mining of large amounts of parallel data for lexical entries, and also improving the parsing model itself with recent advances in CCG semantic parsing.

Dutch and English are relatively close cousins; in ongoing work we are investigating the applicability of our method to a number of Germanic and Romance languages (e.g., German and Italian) and so far have found no theoretical obstacles. To what extent applying it to less closely related language pairs than English/Dutch is harder empirically remains to be investigated. In any case, we are confident that the techniques presented in this paper can help develop multilingual semantic parsers without starting from scratch, software-wise and data-wise, for every new language.

Acknowledgments

References

Ž. Agić, A. Johannsen, B. Plank, H. Martínez Alonso, N. Schluter, and A. Søgaard. 2016. Multilingual projection for parsing truly low-resource languages. *TACL*, 4:301–312.

J.F. Allen, M. Swift, and W. de Beaumont. 2008. Deep Semantic Analysis of Text. In J. Bos and R. Delmonte, editors, *Semantics in Text Processing*. College Publications.

Y. Artzi and L. Zettlemoyer. 2011. Bootstrapping semantic parsers from conversations. In *Proceedings of EMNLP*.

Y. Artzi, K. Lee, and L. Zettlemoyer. 2015. Broad-coverage CCG semantic parsing with AMR. In *Proceedings of EMNLP*.

L. Banarescu, C. Bonial, S. Cai, M. Georgescu, K. Griffitt, U. Hermjakob, K. Knight, P. Koehn, M. Palmer, and N. Schneider. 2013. Abstract Meaning Representation for Sembanking. In *Proceedings of LAW VII & ID*.

V. Basile, J. Bos, K. Evang, and N. J. Venhuizen. 2012. Developing a large semantically annotated corpus. In *Proceedings of LREC*.

B. Bohnet. 2010. Top accuracy and fast dependency parsing is not a contradiction. In *Proceedings of Coling*.

F. Bond and R. Foster. 2013. Linking and extending and open multilingual wordnet. In *Proceedings of ACL*.

C. Bonial, W. J. Corvey, M. Palmer, V. Petukhova, and H. Bunt. 2011. A hierarchical unification of LIRICS and VerbNet semantic roles. In *Proceedings of ICSC*.

J. Bos. 2009. Towards a large-scale formal semantic lexicon for text processing. In *Proceedings of GSCL*.

J. Bos. 2014. Semantic annotation issues in parallel meaning banking. In *Proceedings of ISA-10*.

S. Cai and K. Knight. 2013. Smatch: an evaluation metric for semantic feature structures. In *Proceedings of the 51st Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, pages 748–752, Sofia, Bulgaria, August. Association for Computational Linguistics.

D. L. Chen and R. J. Mooney. 2011. Learning to interpret natural language navigation instructions from observations. In *Proceedings of AAAI*.

S. Clark and J. R. Curran. 2007. Wide-coverage efficient statistical parsing with CCG and log-linear models. *Computational Linguistics*, 33(4):493–552.

J. Clarke, D. Goldwasser, M. Chang, and D. Roth. 2010. Driving semantic parsing from the world's response. In *Proceedings of CoNLL*.

M. Collins and B. Roark. 2004. Incremental parsing with the perceptron algorithm. In *Proceedings of ACL*.

J. Curran, S. Clark, and J. Bos. 2007. Linguistically Motivated Large-Scale NLP with C&C and Boxer. In *Proceedings of ACL: Demo and Poster Sessions*.

B. J. Dorr. 1993. *Machine Translation: A View from the Lexicon*. The MIT Press.

C. Fellbaum, editor. 1998. *WordNet. An Electronic Lexical Database*. The MIT Press.

K. Ganchev, J. Gillenwater, and B. Taskar. 2009. Dependency grammar induction via bitext projection constraints. In *Proceedings of ACL-IJCNLP*.

D. Goldwasser and D. Roth. 2011. Learning from natural instructions. In *Proceedings of IJCAI*.

R. Hwa, P. Resnik, A. Weinberg, C. Cabezas, and O. Kolak. 2005. Bootstrapping parsers via syntactic projection across parallel texts. *Natural Language Engineering*, 11(03):311–325.

A. Johannsen, . Agi, and A. Sgaard. 2016. Joint part-of-speech and dependency projection from multiple sources. In *Proceedings of ACL*.

H. Kamp and U. Reyle. 1993. *From Discourse to Logic; An Introduction to Modeltheoretic Semantics of Natural Language, Formal Logic and DRT*. Kluwer.

J. Krishnamurthy and T. Mitchell. 2012. Weakly supervised training of semantic parsers. In *Proceedings of EMNLP-CoNLL*.

T. Kwiatkowksi, L. Zettlemoyer, S. Goldwater, and M. Steedman. 2010. Inducing probabilistic ccg grammars from logical form with higher-order unification. In *Proceedings of the 2010 Conference on Empirical Methods in Natural Language Processing*, pages 1223–1233. Association for Computational Linguistics.

T. Kwiatkowski, E. Choi, Y. Artzi, and L. Zettlemoyer. 2013. Scaling semantic parsers with on-the-fly ontology matching. In *Proceedings of EMNLP*.

P. Le and W. Zuidema. 2012. Learning compositional semantics for open domain semantic parsing. In *Proceedings of COLING*.

M. Lewis and M. Steedman. 2013. Combined distributional and logical semantics. *TACL*, 1:179–192.

P. Liang, M. I. Jordan, and D. Klein. 2011. Learning dependency-based compositional semantics. In *Proceedings of ACL*.

J. May. 2016. Semeval-2016 task 8: Meaning representation parsing. In *Proceedings of SemEval*, pages 1063–1073.

R. McDonald, S. Petrov, and K. Hall. 2011. Multi-source transfer of delexicalized dependency parsers. In *Proceedings of EMNLP*.

T. Naseem, R. Barzilay, and A.l Globerson. 2012. Selective sharing for multilingual dependency parsing. In *Proceedings of ACL*.

J. Nivre, M. de Marneffe, F. Ginter, Y. Goldberg, J. Haji, C.D. Manning, R. McDonald, S. Petrov, S. Pyysalo, N. Silveira, R. Tsarfaty, and D. Zeman. 2016. Universal dependencies v1: A multilingual treebank collection. In *Proceedings of LREC*.

F.J. Och and H. Ney. 2003. A systematic comparison of various statistical alignment models. *Computational Linguistics*, 29(1):19–51.

H. Poon. 2013. Grounded unsupervised semantic parsing. In *Proceedings of ACL*.

M.S. Rasooli and M. Collins. 2015. Density-driven cross-lingual transfer of dependency parsers. In *Proceedings of EMNLP*.

S. Reddy, M. Lapata, and M. Steedman. 2014. Large-scale semantic parsing without question-answer pairs. *TACL*, 2(1):377–392.

H. Schmid. 1995. Improvements in part-of-speech tagging with an application to German. In *Proceedings of the ACL SIGDAT-Workshop*.

M. Steedman. 2001. *The Syntactic Process*. The MIT Press.

O. Täckström, R. McDonald, and J. Nivre. 2013. Target language adaptation of discriminative transfer parsers. In *Proceedings of NAACL-HLT*.

Y. Tanaka. 2001. Compilation of a multilingual parallel corpus. In *Proceedings of PACLING*.

J. Tiedemann, Ž. Agić, and J. Nivre. 2014. Treebank translation for cross-lingual parser induction. In *Proceedings of CoNLL*.

J. Tiedemann. 2014. Rediscovering annotation projection for cross-lingual parser induction. In *Proceedings of COLING: Technical Papers*.

L. Vanderwende, A. Menezes, and C. Quirk. 2015. An AMR parser for English, French, German, Spanish and Japanese and a new AMR-annotated corpus. In *Proceedings of NAACL-HLT*.

W. Wong and R. Mooney. 2007. Learning synchronous grammars for semantic parsing with lambda calculus. In *Proceedings of ACL*.

D. Zeman and P. Resnik. 2008. Cross-language parser adaptation between related languages. In *Proceedings of the IJCNLP Workshop on NLP for Less Privileged Languages*.

L. Zettlemoyer and M. Collins. 2005. Learning to map sentences to logical form: Structured classification with probabilistic categorial grammars. In *Proceedings of the Twenty-First Conference Annual Conference on Uncertainty in Artificial Intelligence (UAI-05)*, pages 658–666, Arlington, Virginia. AUAI Press.

L. S. Zettlemoyer and M. Collins. 2007. Online learning of relaxed CCG grammars for parsing to logical form. In *Proceedings of EMNLP-CoNLL*.

Y. Zhang and S. Clark. 2011. Shift-reduce CCG parsing. In *Proceedings of ACL*.

K. Zhao and L. Huang. 2015. Type-driven incremental semantic parsing with polymorphism. In *Proceedings of NAACL-HLT*.

M. Zhu, Y. Zhang, W. Chen, M. Zhang, and J. Zhu. 2013. Fast and accurate shift-reduce constituent parsing. In *Proceedings of ACL*.