Word alignment in Discourse Representation Structure parsing

Christian Obereder
TU Wien
e11704936@student.tuwien.ac.at

Gábor Recski
TU Wien
gabor.recski@tuwien.ac.at

Abstract

Discourse Representation Structures (DRS) are formal representations of linguistic semantics based on Discourse Representation Theory (DRT, Kamp et al., 2011) that represent meaning as conditions over discourse referents. State-of-the-art DRS parsers learn the task of mapping text to DRSs from annotated corpora such as the Parallel Meaning Bank (PMB, Abzianidze et al., 2017). Using DRS in downstream NLP applications such as Named Entity Recognition (NER), Relation Extraction (RE), or Open Information Extraction (OIE) requires that DRS clauses produced by a parser be aligned with words of the input sentence. We propose a set of methods for extending such models to learn DRS-to-word alignment in two ways, by using learned attention weights for alignment and by adding alignment information from the PMB to the training data. Our results demonstrate that combining the two methods can achieve an alignment accuracy of over 98%. We also perform manual error analysis, showing that most remaining alignment errors are caused by one-off mistakes, many of which occur in sentences with multi-word expressions.

1 Introduction

Discourse Representation Structures (DRS) are formal representations of linguistic semantics based on Discourse Representation Theory (DRT) (DRT, Kamp et al., 2011) that represent meaning as conditions over discourse referents. State-of-the-art DRS parsers learn the task of mapping text to DRSs from annotated corpora such as the Parallel Meaning Bank (PMB, Abzianidze et al., 2017). Using DRS in downstream NLP applications such as Named Entity Recognition (NER), Relation Extraction (RE), or Open Information Extraction (OIE) requires that DRS clauses produced by a parser be aligned with words of the input sentence. Figure 1 shows an example DRS encoding the meaning of the sentence *The eagle is white*, complete with DRS-to-word alignment information.

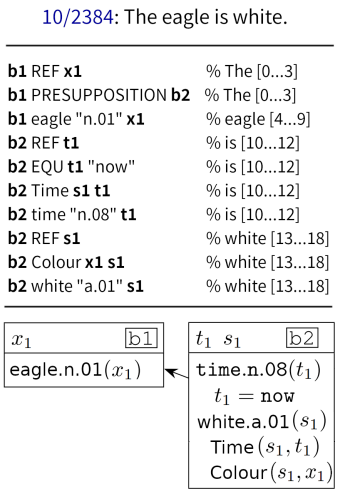


Figure 1: DRS in box- and clause-format for the sentence *The eagle is white*, with DRS-to-word alignments, from the PMB 3.0.0 corpus. 10/2384 is the ID of the sample in the PMB.

Unlike rule-based parsers such as Boxer (Bos, 2015), modern end-to-end parsers such as Neural-DRS (van Noord et al., 2018) do not generate this alignment. We propose a set of methods for extending such models to learn DRS-to-word alignment in two ways, by using learned attention weights for alignment and by adding alignment information from the PMB to the training data. Our results demonstrate that combining the two methods can achieve an alignment accuracy of over 98%. We also perform manual error analysis, showing that most remaining alignment errors are caused by one-off mistakes, many of which occur in sentences with multi-word expressions. The remainder of this paper is structured as follows. Section 2 summarizes related work on DRS parsing and attention-based alignment. Section 3 presents our main methods, Section 4 describes the experimental setup. Section 5 presents our experimental results, Sec-

tion 6 describes results of our manual error analysis. All software used in our experiments is released under an MIT license and is available on GitHub[1].

2 Related Work

Recent work on DRS parsing involves the training of a variety of deep learning architectures on ground truth data created using a combination of automatic rule-based parsing with the Boxer parser (Bos, 2015) and manual error correction. Such systems include various structure-aware encoder-decoder models (Liu et al., 2018, 2019), an RNN-based parser of DAG-grammars (Fancellu et al., 2019), as well as sequence-to-sequence models (van Noord et al., 2018) that were recently used with pretrained language models and character embeddings to achieve some additional improvement in parsing performance (van Noord et al., 2020). It is this latter set of models, implemented as part of the NeuralDRS[2] codebase, that this paper extends to include the learning of DRS-to-word alignment (see Section 3 for details).

Most recent work on DRS parsing relies on the Parallel Meaning Bank (PMB, Abzianidze et al., 2017) for training and evaluation data. The PMB is a multilingual corpus containing sentences in English, German, Italian, and Dutch together with a variety of syntactic and semantic annotations. DRSs are generated for English using the Boxer parser and undergo various degrees of manual correction to create three subsets of the dataset. About 6,000 sentences have gold standard DRS annotations, another 67,000 constitute the silver dataset, these contain DRSs that have undergone at least one manual correction step, while about 120,000 sentences without any manual correction constitute the bronze portion of the dataset. Recent work has demonstrated that the inclusion of silver-quality annotation into the model training results in increased parsing performance (van Noord et al., 2018). Much related work on DRS parsing relies on the 2.1.0 and 2.2.0 versions of the PMB corpus (Abzianidze et al., 2019), we follow the more recent work of (van Noord et al., 2020) and use the 3.0.0 version in our experiments. DRS annotations in the PMB also contain alignment information, mapping nearly all DRS clauses to one or more tokens of the input text, as illustrated in Figure 1. We use this data both for model training and for evaluation of our main methods.

[1] https://github.com/GitianOberhuber/Neural_DRS_alignment

[2] https://github.com/RikVN/Neural_DRS

3 Methods

We propose a set of methods for extending the NeuralDRS parser architecture of van Noord et al. (2020) to include the task of DRS-to-word alignment, i.e. to map each DRS clause output by the parser to the word of the input sentence corresponding to the semantic information encoded by the DRS clause. The alignment information present in the PMB dataset (see Figure 1 and our discussion in Section 2) is used for both training and evaluation of our proposed models. The first method involves including the alignment data from PMB directly in the training data of the NeuralDRS system so that it learns to generate DRS-to-word alignments as part of its output. The second method involves using the attention scores computed by the NeuralDRS model to directly align DRS clauses in the output to words of the input. This method can be applied to the model trained using the original PMB data as well as the one trained on the modified version including word alignments. We show in Section 5 that it is the latter, combined method that achieves the highest accuracy on the DRS-to-word alignment task.

3.1 Alignment generation

Our first method involves creating a modified version of the training data that contains alignment information present in the PMB. For example, in case of the example sentence used in Figure 1, the string *b1 REF x1* would be replaced by *b1 REF x1 % The [0...3]* in the data. This data is then used to train the NeuralDRS system so that it learns to directly generate word alignments for each DRS clause. This approach does not guarantee that the model will output well-formed alignments, we therefore perform a simple form of fuzzy matching. For each generated word that is not a perfect match to one of the input words we choose the one with the lowest Levenshtein distance (Levenshtein, 1966).

3.2 Attention-based alignment

Our second method maps generated DRS clauses to input tokens using the attention scores calculated by the NeuralDRS model. Attention mechanisms in sequence-to-sequence models learn weighted alignments between input and output tokens. Given any alignment model a that maps pairs of encoder

and decoder states we can define the alignment scoring function as

$$e_{t't} = a(s_{t'-1}, h_t)$$

where h_t is the encoder hidden-state at timestep t and $s_{t'-1}$ is the decoder hidden-state at timestep $t'-1$. Then for some timestep t' the context-vector $c_{t'}$ can be calculated as

$$c_{t'} = \sum_{t=1}^{T} \alpha_{t't} h_t,$$

where the weight $\alpha_{t't}$ is calculated as

$$\alpha_{t't} = \frac{exp(e_{t't})}{\sum_{k=1}^{T} exp(e_{t'k})}$$

Our attention-based alignment method maps each output token to the input token with the largest alignment score. Formally, given an input sequence $x = \{x_1, ..., x_T\}$ and corresponding (encoder-) timesteps $\tau = \{1, ..., T\}$, for each decoder timestep t' we calculate

$$\underset{t \epsilon \tau}{\text{argmax}} \quad \frac{exp(e_{t't})}{\sum_{k=1}^{T} exp(e_{t'k})}.$$

Since our goal is to align DRS clauses, which consist of multiple output tokens, we calculate average scores over all tokens belonging to a given DRS clause.

The original NeuralDRS architecture uses dot-product attention (Luong et al., 2015), which defines the alignment score a as $h_t^\intercal s_{t'}$. For our attention-based alignment method we use both dot-product attention and bilinear attention, the latter of which defines a as $h_t^\intercal W s_{t'}$, where W is a learned matrix of weights. Our experiments show that the use of bilinear attention leads to improved alignment accuracy (see Section 5).

4 Experiments

Each of our experiments extends the single-encoder BERT-based model described by van Noord et al. (2020) and made available on GitHub[3]. We train models with two datasets, the original PMB data and the alignment-augmented data, the latter allows models to directly generate DRS-to-word alignments, as described in Section 3.1. Both types of models are also used to extract DRS-to-word alignments from their attentions weights, as described in Section 3.2.

[3] https://github.com/RikVN/Neural_DRS

All experiments are conducted using the English data of the 3.0.0 release of the PMB. The train portion of the gold data as well as all of the silver data is used for initial model training, followed by fine-tuning only on the gold data. Fine-tuning is performed five times with different random seeds, initial training is performed only once. To save resources, the maximum number of epochs (for both initial training and fine-tuning) was limited to 4. Models are implemented using the open-source AllenNLP framework (Gardner et al., 2018). Data preprocessing follows the original system described in van Noord et al. (2020), postprocessing of model outputs to produce final alignments is performed as described in Section 3. Model hyperparameters are shown in Appendix A.

For each of our models we evaluate both parsing quality and alignment accuracy. For measuring parsing performance we rely on the methodology of van Noord et al. (2020). This involves finding the optimal mapping from variable names used by predicted DRSs to those used in the ground truth, then calculating the precision, recall, and F-score of predicted DRS-clauses, ignoring *REF* clauses that serve to introduce variables and would always count as true positives, inflating scores unnecessarily. For measuring alignment accuracy we only consider correctly predicted DRS-clauses (including *REF* clauses) and define accuracy as the ratio of such clauses that have been aligned to the correct input word. Since we expect parsing errors to negatively affect the system's ability to align correctly predicted DRS-clauses, we also calculate alignment accuracy on the subset of sentences for which the DRS was parsed perfectly, i.e. those DRS where all clauses have been correctly predicted. The ratio of such sentences varies between 33% and 37% across parsing models. Furthermore, when comparing predictions to ground truth alignments we treat the following two cases exceptionally:

Multi-word tokens The PMB data contains some multi-word tokens, corresponding to named entities or other multi-word expressions, and represented in the corpus as e.g. *10~a.m.* The NeuralDRS pipeline does not have access to this analysis and processes the words *10* and *a.m* separately. If the PMB aligns a DRS clause to such a token, we consider our predicted alignment correct if and only if it maps the clause to one of the words of the multi-word token.

Multiple alignments A small fraction of DRS clauses in the PMB corpus is aligned with more than one input word. We consider these correctly aligned if our prediction corresponds to one of the multiple ground truth alignments.

5 Results

Table 1 shows all evaluation results on both the dev and test portions of the PMB 3.0.0 dataset. We observe that bilinear attention outperforms dot-product attention by a large margin when used to directly capture DRS-to-word alignment, as described in Section 3.2. In Appendix B we also provide visual comparison of the two types of attention that illustrates this difference. The end-to-end approach (Section 3.1) of training a model with DRS data augmented with alignment information from the PMB and using this model to generate the DRS-to-word alignment is superior to the attention-based methods. However, the highest accuracy is achieved by the combination of the two methods, i.e. using the attention weights of the end-to-end model for direct DRS-to-word alignment.

When evaluating on the subset of sentences which have been perfectly parsed, alignment accuracy increases considerably and is nearly perfect for both the end2end and combined approaches. This is in line with our expectation that errors in aligning correctly predicted DRS clauses typically occur around parsing errors. Since about two thirds of all sentences contain at least one parsing error, the combined approach is clearly the most practical choice for performing DRS-to-word alignment. We also measure the performance of each model on the DRS parsing task, but since we trained each model with a lower number of epochs to save resources, it is unsurprising that these figures are somewhat below the performance of the original NeuralDRS model (van Noord et al., 2020).

6 Error analysis

We perform manual analysis of alignment errors made by the end-to-end and combined approaches. For each model, sample outputs of approx. 40 sentences each were extracted from both the original dev set and the one filtered to contain only correctly parsed sentences. Here we describe only the most common error types of each approach.

Incorrect words The end-to-end approach will map some DRS-clauses to a word not present in the input sentence. Sometimes these are synonyms of the expected word, e.g. in the sentence *Is hexane toxic?*, the parser maps the clauses aligned with *toxic* to the word *poisonous*. In other examples the model produces ("hallucinates") unrelated words, e.g. in the sentence *Tom is addicted to heroin* the correctly predicted DRS clause `b2 heroin "n.01" x2` is mapped to the nonexistent input word *sobs*. These errors are often propagated across multiple DRS clauses aligned with the same input word, this way they are responsible for the majority of all errors made by the end-to-end approach on our samples.

One-off errors Unlike the end-to-end method, the combined approach is guaranteed to map each DRS clause to an existing input word. The majority of errors made by this approach are one-off mistakes, i.e clauses are mapped to a word adjacent to the one it is actually aligned with. Further inspection reveals that such errors often occur in sentences that either contain multi-word tokens (e.g. the DRS parse of the sentence *Mr. Ford is all right now* correctly contains the clause `b2 all_right "a.01" s1` but it is erroneously mapped to *now*) or multiple words mapped to a single word sense (e.g. from the sentence *I chopped a tree down.* the parser correctly generates the clause `b1 chop_down "v.01" x1` but then incorrectly maps the last clause `b1 tree "n.01" x3` to the last word *down*.

7 Conclusion

We have proposed two methods for extending a state-of-the-art DRS parser to perform DRS-to-word alignment and have shown that their combination achieves over 98% alignment accuracy on correctly predicted DRS clauses. Manual error analysis indicates that end-to-end generation of word alignment, which on its own achieves less than 96% accuracy, propagates errors caused by erroneously generated words across multiple DRS clauses. The combined approach of using attention scores, on the other hand, guarantees that each clause is mapped to existing input words and reduces the errors of the end-to-end approach by more than half. Additional error analysis suggests that multi-word expressions may be a major source of remaining alignment errors.

Ethical considerations

The main motivation of the present work is to enable the use of semantic parsing in complex NLP pipelines that rely on the information encoded in

	Dev			Test		
Method	**All sens**	**Corr. DRS**	**DRS F1**	**All sens**	**Corr. DRS**	**DRS F1**
Noord et al.	-	-	**87.58** $\pm$ 0.19	-	-	**88.53** $\pm$ 0.26
Attention (dot-prod.)	82.15 $\pm$ 0.91	83.33 $\pm$ 0.91	86.69 $\pm$ 0.25	82.15 $\pm$ 0.88	83.09 $\pm$ 1.00	87.10 $\pm$ 0.52
Attention (bilinear)	86.34 $\pm$ 0.59	88.08 $\pm$ 0.54	86.40 $\pm$ 0.48	86.36 $\pm$ 0.60	87.40 $\pm$ 0.81	87.17 $\pm$ 0.41
End-to-end	95.84 $\pm$ 0.19	**99.56** $\pm$ 0.09	84.89 $\pm$ 0.30	95.93 $\pm$ 0.19	**99.68** $\pm$ 0.12	85.74 $\pm$ 0.46
Combined (bilinear)	**98.49** $\pm$ 0.13	99.33 $\pm$ 0.08	84.89 $\pm$ 0.30	**98.46** $\pm$ 0.11	99.44 $\pm$ 0.11	85.74 $\pm$ 0.46

Table 1: DRS-to-word alignment performance of the proposed methods. *All sens* is alignment accuracy on the full English dev- and test-set of PMB 3.0.0, *Corr. DRS* uses the subset of sentences for which predicted DRSs are fully correct. DRS F1 is the parsing performance of each model. Attention-based alignment methods are based on model weights, as described in Section 3.2. The *end-to-end* method uses alignments generated by the model, as described in Section 3.1. The combined method uses the attention weights from the model trained to perform end-to-end alignment. All figures are mean values over 5 runs.

DRS structures to perform information extraction tasks such as Relation Extraction or Open Information Extraction with rule-based or hybrid methods. Partially or fully symbolic IE models can effectively expose and mitigate risks associated with black box models such as unintended model bias (Bender et al., 2021; De-Arteaga et al., 2019; Nadeem et al., 2021), lack of explainability of model decisions (Jain and Wallace, 2019), and vulnerabilities against adversarial attacks (Kour et al., 2023).

Limitations

This short paper presents experiments using a single dataset (PMB) and modifying a single architecture for semantic parsing (NeuralDRS). Furthermore, our conclusions are limited to the alignment task for a single type of semantic parsing formalism (DRS). In-depth investigation of the task of word alignment in semantic parsing should include experiments involving other common semantic parsing formalisms such as AMR (Banarescu et al., 2013) and UCCA (Abend and Rappoport, 2013), while experiments like those performed in this work should be repeated on multiple state-of-the-art sequence-to-sequence architectures for semantic parsing.

References

Omri Abend and Ari Rappoport. 2013. Universal Conceptual Cognitive Annotation (UCCA). In *Proceedings of the 51st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 228–238, Sofia, Bulgaria. Association for Computational Linguistics.

Lasha Abzianidze, Johannes Bjerva, Kilian Evang, Hessel Haagsma, Rik van Noord, Pierre Ludmann, Duc-Duy Nguyen, and Johan Bos. 2017. The Parallel Meaning Bank: Towards a multilingual corpus of translations annotated with compositional meaning representations. In *Proceedings of the 15th Conference of the European Chapter of the Association for Computational Linguistics: Volume 2, Short Papers*, pages 242–247, Valencia, Spain. Association for Computational Linguistics.

Lasha Abzianidze, Rik van Noord, Hessel Haagsma, and Johan Bos. 2019. The first shared task on discourse representation structure parsing. In *Proceedings of the IWCS Shared Task on Semantic Parsing*, Gothenburg, Sweden. Association for Computational Linguistics.

Laura Banarescu, Claire Bonial, Shu Cai, Madalina Georgescu, Kira Griffitt, Ulf Hermjakob, Kevin Knight, Philipp Koehn, Martha Palmer, and Nathan Schneider. 2013. Abstract Meaning Representation for sembanking. In *Proceedings of the 7th Linguistic Annotation Workshop and Interoperability with Discourse*, pages 178–186, Sofia, Bulgaria. Association for Computational Linguistics.

Emily M. Bender, Timnit Gebru, Angelina McMillan-Major, and Shmargaret Shmitchell. 2021. On the dangers of stochastic parrots: Can language models be too big? In *Proceedings of the 2021 ACM Conference on Fairness, Accountability, and Transparency*, FAccT '21, page 610–623, New York, NY, USA. Association for Computing Machinery.

Johan Bos. 2015. Open-domain semantic parsing with boxer. In *Proceedings of the 20th Nordic Conference of Computational Linguistics (NODALIDA 2015)*, pages 301–304, Vilnius, Lithuania. Linköping University Electronic Press, Sweden.

Maria De-Arteaga, Alexey Romanov, Hanna Wallach, Jennifer Chayes, Christian Borgs, Alexandra Chouldechova, Sahin Geyik, Krishnaram Kenthapadi, and Adam Tauman Kalai. 2019. Bias in bios: A case study of semantic representation bias in a high-stakes setting. In *Proceedings of the Conference on Fairness, Accountability, and Transparency*, FAT* '19, page 120–128, New York, NY, USA. Association for Computing Machinery.

Federico Fancellu, Sorcha Gilroy, Adam Lopez, and Mirella Lapata. 2019. Semantic graph parsing with recurrent neural network DAG grammars. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 2769–2778, Hong Kong, China. Association for Computational Linguistics.

Matt Gardner, Joel Grus, Mark Neumann, Oyvind Tafjord, Pradeep Dasigi, Nelson F. Liu, Matthew Peters, Michael Schmitz, and Luke Zettlemoyer. 2018. AllenNLP: A deep semantic natural language processing platform. In *Proceedings of Workshop for NLP Open Source Software (NLP-OSS)*, pages 1–6, Melbourne, Australia. Association for Computational Linguistics.

Sarthak Jain and Byron C. Wallace. 2019. Attention is not Explanation. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 3543–3556, Minneapolis, Minnesota. Association for Computational Linguistics.

Hans Kamp, Josef van Genabith, and Uwe Reyle. 2011. Discourse representation theory. In *Handbook of philosophical logic*, pages 125–394. Springer.

George Kour, Marcel Zalmanovici, Naama Zwerdling, Esther Goldbraich, Ora Fandina, Ateret Anaby Tavor, Orna Raz, and Eitan Farchi. 2023. Unveiling safety vulnerabilities of large language models. In *Proceedings of the Third Workshop on Natural Language Generation, Evaluation, and Metrics (GEM)*, pages 111–127, Singapore. Association for Computational Linguistics.

Vladimir I Levenshtein. 1966. Binary codes capable of correcting deletions, insertions, and reversals. *Soviet physics – doklady*, 10(8):707–710.

Jiangming Liu, Shay B. Cohen, and Mirella Lapata. 2018. Discourse representation structure parsing. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 429–439, Melbourne, Australia. Association for Computational Linguistics.

Jiangming Liu, Shay B. Cohen, and Mirella Lapata. 2019. Discourse representation parsing for sentences and documents. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 6248–6262, Florence, Italy. Association for Computational Linguistics.

Thang Luong, Hieu Pham, and Christopher D. Manning. 2015. Effective approaches to attention-based neural machine translation. In *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing*, pages 1412–1421, Lisbon, Portugal. Association for Computational Linguistics.

Moin Nadeem, Anna Bethke, and Siva Reddy. 2021. StereoSet: Measuring stereotypical bias in pretrained language models. In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, pages 5356–5371, Online. Association for Computational Linguistics.

Rik van Noord, Lasha Abzianidze, Antonio Toral, and Johan Bos. 2018. Exploring neural methods for parsing discourse representation structures. *Transactions of the Association for Computational Linguistics*, 6:619–633.

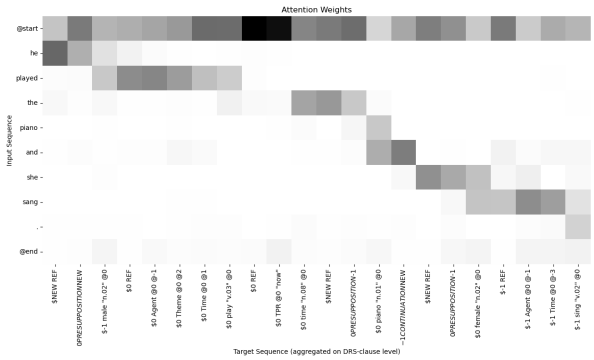
Rik van Noord, Antonio Toral, and Johan Bos. 2020. Character-level representations improve DRS-based semantic parsing even in the age of BERT. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 4587–4603, Online. Association for Computational Linguistics.

A Model parameters

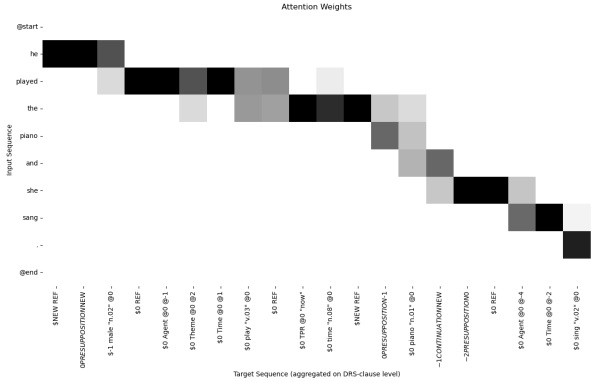
All hyperparameters used in the experiments described in Section 4 are shown in Table 2.

B Attention weights

Figure 2 compares dot-product and bilinear attention, illustrating the quantitative results in Section 5 that show the superior ability of bilinear attention to align generated DRS clauses with corresponding input words.



(a) dot-product attention



(b) bilinear attention

Figure 2: Visualization of dot-product and bilinear attention weights on a sample sentence from the PMB. Weights are aggregated on DRS-clause level, as described in Section 3.1

Input Embedding	
Type	bert-base-uncased
Size	768
Max. # source tokens	125
trainable	false
Target Embedding	
Type	pretrained GloVe
Size	300
Max. # tokens	1160
trainable	true
Encoder	
Type	biLSTM
Hidden Size	300
LSTM Layers	1
Attention	
Type	dot product / bilinear
normalize	true
matrix_dim	- / 600
vector_dim	- / 600
Decoder	
Type	LSTM
Hidden size	300
LSTM Layers	1
max_norm	3
scale_grad_by_freq	false
label_smoothing	0.0
beam_size	10
max decoding steps	1000
schedule sampling	0.2
Trainer	
batch size	12
optimizer	adam
learning rate	0.001
grad_norm	0.9
max_epochs	4

Table 2: Hyperparameters used in the experiments. Except for the values in red, all hyperparameters are equal to that of van Noord et al. (2020)