World Knowledge for Abstract Meaning Representation Parsing

Charles Welch[1], **Jonathan K. Kummerfeld**[1], **Song Feng**[2], **Rada Mihalcea**[1]
University of Michigan[1], IBM Research[2]
{cfwelch, jkummerf, mihalcea}@umich.edu, sfeng@us.ibm.com

Abstract

In this paper we explore the role played by world knowledge in semantic parsing. We look at the types of errors that currently exist in a state-of-the-art Abstract Meaning Representation (AMR) parser, and explore the problem of how to integrate world knowledge to reduce these errors. We look at three types of knowledge from (1) WordNet hypernyms and super senses, (2) Wikipedia entity links, and (3) retraining a named entity recognizer to identify concepts in AMR. The retrained entity recognizer is not perfect and cannot recognize all concepts in AMR and we examine the limitations of the named entity features using a set of oracles. The oracles show how performance increases if it can recognize different subsets of AMR concepts. These results show improvement on multiple fine-grained metrics, including a 6% increase in named entity F-score, and provide insight into the potential of world knowledge for future work in Abstract Meaning Representation parsing.

Keywords: semantic parsing, Abstract Meaning Representation, world knowledge, named entity recognition

1. Introduction

Abstract Meaning Representation (AMR), introduced by Banarescu et al. (2012), aims to capture the semantic meaning of sentences using directed acyclic graphs where nodes are labeled with *concepts* and edges are labeled with *relations*. An example is shown in Figure 1. A number of recent studies use AMR graphs for downstream tasks (Pan et al., 2015; Liu et al., 2015; Sachan and Xing, 2016; Burns et al., 2016) and growing amounts of annotated data enable the development of statistical parsing algorithms and standardized evaluations.

Several parsers have been created that generate an AMR graph given a sentence (Flanigan et al., 2014; Wang et al., 2015b; Damonte et al., 2016), but even the most recent results suggest that there is still significant room to improve the performance for this challenging task.

In this paper, we explore the role of world knowledge for the task of semantic parsing with AMR. The paper makes three main contributions. First, we examine the effect of different types of world knowledge for semantic parsing with AMR for the first time.[1] Second, we examine the upper bound on world knowledge using gold annotations, and provide new insights into the potential of world knowledge in computational approaches to AMR parsing. Finally, we show that we can improve the parsing score over a state-of-the-art parser, with improvement on multiple fine-grained evaluation metrics, including a 6% increase in named entity F-score.

2. Background

There are several semantic parsers built for AMR annotations (Flanigan et al., 2014; Zhou et al., 2016; Wang et al., 2015a; Damonte et al., 2016; Barzdins and Gosko, 2016; Misra and Artzi, 2016) using data released through the LDC (LDC2014T12, LDC2015E86, LDC2016E25, LDC2017T10). The 2014 dataset contains 13,000 sentences, which increased to almost 20,000 in the 2015 set. The 2016 and 2017 datasets contain around 39,000 sentences. The data consist of sentences from English broadcast conversations, weblogs, discussion forums, and newswire data and are annotated with AMR graphs. The datasets build off of each other and include corrections and extensions of the AMR specification (e.g., 2015 introduces wikification and new PropBank frames).

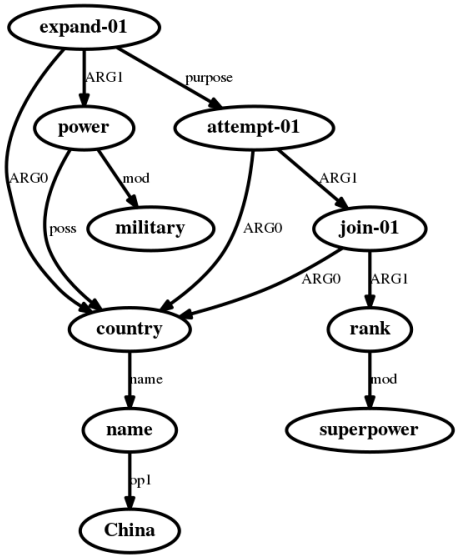

Figure 1: Example AMR graph for the sentence *"China is expanding its military power to attempt to join the ranks of the superpowers"*. Concepts are represented as nodes and relations as edges between those nodes.

The evaluation of these AMR parsers is typically based on the SMATCH F1 tool (Cai and Knight, 2013) which measures the overlap of concept-relation-concept triples in a generated AMR graph as compared to the gold graph. In addition, Damonte et al. (2016) introduced finer-grained evaluations of the subtasks of AMR parsing, which

[1]The code modifications and features are available at `https://github.com/cfwelch/amr_world_knowledge`.

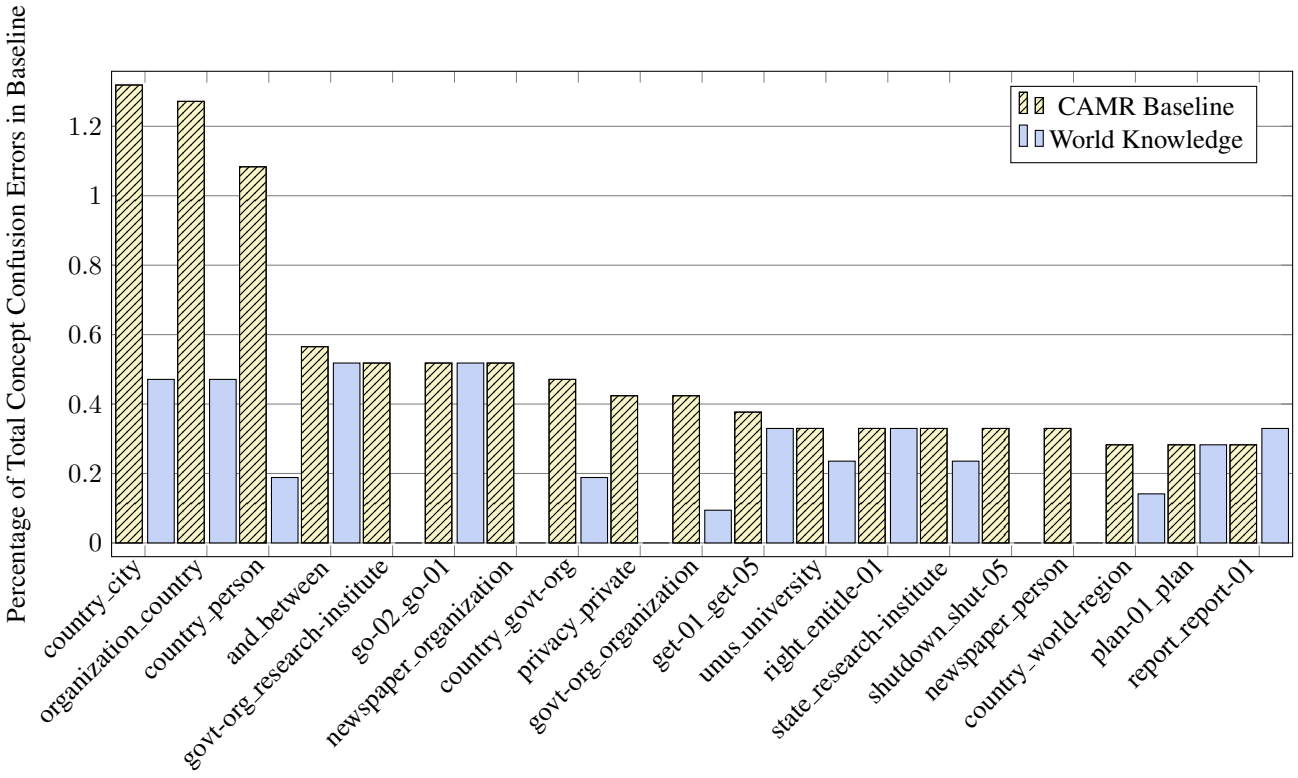

Figure 2: The top 10% of concepts incorrectly identified by the CAMR baseline compared to the percentage of concept confusion errors of each of these types after the addition of world knowledge. The first concept listed is the gold label and the second is the concept it was incorrectly labeled with. We use govt as an abbreviation for 'government'. The numbers after concept names represent their word senses. For instance, get-01 means 'to come into posession', whereas get-05 means 'to get to a state'.

measure parser effectiveness in terms of capturing named entities, concepts, negations, word sense disambiguations and semantic roles.

At the time these experiments were performed many of the high performing parsers were based on JAMR or CAMR (Flanigan et al., 2014; Wang et al., 2015a). We chose to base our parser on CAMR, which was the highest performing entry on SemEval 2016 Task 8 and the parser on which most of the entries for the 2016 shared task were based (May, 2016; Wang et al., 2016). Comparing to previous extensions of CAMR, our work is the first attempt to integrate various forms of world knowledge as features for AMR parsing.

3. AMR Parsing with World Knowledge

We hypothesize that introducing world knowledge could potentially increase the overall performance of AMR parsers. In order to identify useful features and effective approaches to improve an existing AMR parser, we first examine the errors produced by AMR parsers. By looking at errors made by CAMR, we found that a significant number of concepts are either mislabeled or missing. The lighter, striped bars in Figure 2 show the most frequent concept identification errors, representing 10% of the overall concept identification errors that CAMR makes. For instance, as seen in this chart, the concept of *country* is incorrectly labeled as *city*, *organization* is incorrectly labeled as *country*, and *country* is incorrectly labeled as *person* as the top 3 most common errors. These 3 errors make up 3.7% of the total concept identification errors. Based on this error analysis, we chose to integrate world knowledge to reduce concept identification errors.

3.1. World Knowledge for AMR

We examine three types of world knowledge: semantic classes (WordNet classes), named entities, and encyclopedic knowledge (entity links to Wikipedia).

WordNet Classes and Supersenses. WordNet organizes words into semantic hierarchies, and therefore it can be used to abstract words to more general concepts (also referred to as *supersenses* (Miller, 1995)). We use WordNet in two ways.

First, we use a set of 45 WordNet supersenses, 26 of which are for nouns, as assigned by the lexicographers who developed WordNet. Every noun or verb in WordNet is subsumed by one of these supersenses.

Second, we abstract even further by taking advantage of the hypernym hierarchy for nouns. Given a word, we identify its synsets, then for each synset we generate a trace from that node to the root of the hierarchy following hypernym links. Next, we take the labels of nodes in the

traces three steps from the root, which keeps the number of classes small while still having meaningful abstractions. The resulting set of fifteen classes is {`group, thing, measure, change, object, substance, causal agent, relation, matter, horror, communication, psychological feature, set, process, attribute`}.

Named Entities. We use the Stanford named entity tagger (Finkel et al., 2005), and retrain it on the training set of LDC2014T12. To retrain the tagger we need spans in the AMR training sentences and their assigned labels.

To generate annotated data, we use the alignments automatically generated using the JAMR aligner for AMR graphs. The aligner creates token spans corresponding to the generation of concepts in the gold AMR graph. We then use the frequent concept labels as a set of classes for retraining the NE tagger. We consider two methods for choosing the classes for our named entity system. The first method is to list all the concept labels by their frequency and choose the top twenty, removing all the non-noun labels, and removing classes which have low F1 scores when retrained, meaning that the NE model cannot reliably recognize this entity type (we end up with nine classes). The second is to look at the most frequent occurrences of nodes that have name relations, and use this set of types as classes (excluding the *name* type itself).

Entity Linking to Wikipedia. We apply the TAGME entity linker (Ferragina and Scaiella, 2010), which takes a sentence as input and produces an annotated output showing which spans of tokens in the sentence correspond to Wikipedia entries. It also provides a confidence score for the linking. In our implementation, we only consider the entity links that have a confidence over 50%.

3.2. Integration with the CAMR Parser

We integrate world knowledge into the CAMR parser (Wang et al., 2015b), which is one of the top performing AMR parsers. CAMR first generates a dependency parse of a sentence and then iterates over nodes in the dependency tree and decides at each point which of a set of transitions to take. To integrate world knowledge into this parser, we change the feature set generated by each node in the current context window. CAMR scores each possible transition at each step of the parse. The context window can contain the current buffer node, a node representing a potential edge with the current buffer node, and a potential parent node. For each of these nodes, we add features that either have a categorical or boolean value, reflecting the world knowledge available for that node.

4. Experiments

We perform two sets of experiments. In the first set, we automatically infer the values of the three types of world knowledge and use these as features in conjunction with the CAMR parser. In the second set of experiments we use gold standard NER annotations, which we then use to train CAMR to determine an upperbound on the performance of this parser when using world knowledge. We use max-margin learning with AdaGrad instead of the perceptron method in the original code. Previous work has shown this learning method to be effective for a variety of language processing tasks and we observe the same effect (Kummerfeld et al., 2015). In all these experiments, the original CAMR parser naturally constitutes our baseline.

Dataset and Evaluations Metrics. We evaluate our work in two ways: one is overall SMATCH performance (Cai and Knight, 2013), which most of the previous work adopts; the other is the finer-grained evaluation introduced by Damonte et al. (2016), which evaluates the quality of each subtask of AMR parsing.
We focus on a few particularly relevant metrics:

- UNLABELED is the SMATCH score computed on the predicted graphs ignoring all edge labels. It could help tell us if world knowledge helps improve performance on graph structure prediction.
- NO WSD is the score computed by ignoring the word senses after concepts (e.g. 'get-01' and 'get-05' would both become 'get').
- NAMED ENT(ITIES) is the score only checking if the named entity concepts are correct.
- NEGATIONS considers 'polarity' edges in the graph and computes the accuracy of negated concepts.
- CONCEPTS is the score looking only at concept labels and not edge labels.
- REENTRANCY is the score only of reentrant edges in the AMR graph.
- SRL is the score for semantic role labeling which only considers the edge labels.

The dataset we primarily experiment with is LDC2014T12, which was originally used by the CAMR system we extend and does not include Wikification. We experimented with LDC2015E86 and found lower performance comparing to LDC2014T12, which is consistent with recent findings (Zhou et al., 2016; Damonte et al., 2016). Our SMATCH is comparable with Damonte et al. (2016) and outperforms Wang et al. (2015b), however is 3% lower than the graph-based approach used by Flanigan et al. (2016).[2]

4.1. Automatic World Knowledge Augmentations

We augment the CAMR features with the three types of world knowledge described in Section 3. As seen in the left side of Table 1, the addition of world knowledge features leads to an increase in the overall SMATCH F-score as well as several of the finer-grained evaluations. The largest improvement is observed in the named entity subtask, by an absolute 6%. Interestingly, using all of our features in

[2]Flanigan et al. (2016) does not use the types of world knowledge integrated in this paper. We leave integration of world knowledge into the JAMR parser to future work.

		Retrained NER					Oracle NER			
METRIC	CAMR	EL	NE	WN	NE+WN	ALL	9-CLASS	NE	NON-NE	ALL
SMATCH	64.7	**66**	**66**	**66**	**66**	65	67	65	68	**69**
UNLABELED	70.8	**72**	**72**	**72**	**72**	71	73	71	**74**	**74**
NO WSD	65.7	**67**	**67**	**67**	**67**	66	68	66	69	**70**
NAMED ENT.	74.6	77	77	77	**81**	77	**79**	77	78	**79**
NEGATIONS	14.1	**16**	**16**	13	15	14	15	15	**16**	15
CONCEPTS	80.2	80	80	80	**81**	80	81	80	**85**	**85**
REENTRANCY	36.2	38	36	**40**	37	36	**37**	36	**37**	**37**
SRL	**59.1**	59	59	59	58	58	60	59	**61**	**61**

Table 1: Comparisons of CAMR modifications and improvement over the default CAMR model on LDC2014T12 are shown on the left side of the table as SMATCH F1 scores. We examine feature subsets for entity links (EL), named entities (NE), and WordNet features (WN). The right side shows experiments using gold NER concept labels. The highest numbers for each row of each side are in bold.

combination did not perform as well as the combination of named entity and WordNet features. The noise associated with the entity link feature may be the reason why this feature set does not contribute to the overall best classifier. Additionally, we saw 6% reduced false-positives (concepts identified in the parse that should not exist) when using world knowledge than in the CAMR baseline.

Overall, our approach confirms the hypothesis that world knowledge can help an AMR parser. Specifically focusing on the types of errors identified in Section 3, the darker shaded bars in Figure 2 show the errors obtained in the presence of world knowledge. We observe a clear decrease in error, e.g., the percentage of mistakes for "government organization and research institution" drops to zero.

4.2. Gold Standard World Knowledge

To gain additional insight into the role played by NER in the AMR parser, we examine what would happen if we had an NER model that was perfectly accurate for a given set of concept labels. We obtain these NER gold standard labels from the annotations available in the dataset that we use, after aligning the annotation graphs with the raw text. We look at four different scenarios: (1) one scenario where we have labels for the nine classes we used in Section 3.; (2) a second one where we use all named entity concept labels as features; (3) a third scenario using all non-named-entity concept labels as features; and (4) a fourth one where all concept labels of aligned tokens are used as features.

The first column in the right side of Table 1 shows the SMATCH score achieved by using the gold labels for our first method of generating NE tags, which limited our tagger to nine classes. In the second and third experiments we partition the set of labels into NEs and non-NEs. The NEs are the set of concept labels that have 'name' edges in the AMR training data in LDC2014T12. The gold IOB labels for these 252 classes are used to train the parser in experiment 2, and in experiment 3 we use all concepts that are not included in this set which includes about 10k types of labels. The NE score is lower than the 9-class score because NE does not include the 'name' label itself, which is a separate node in the AMR parse and gets aligned to a large number of tokens.

In the last experiment we assume that an NER system can be trained on all concept label types simultaneously and we train and test the parser using these labels as features. As expected, the performance increases significantly in this case. Interestingly, as the performance for most evaluation types increases, the named entity performance is highest when using the real output of the NER system. The gold NER outputs have no effect on the negation or reentrancy scores. Having correct labels intuitively has less to do with these two aspects of AMR graphs.

5. Discussion and Conclusions

Our experiments confirmed the hypothesis that some forms of world knowledge can improve existing AMR parsers. In particular, we found that the combination of named entities and WordNet features outperforms other methods on almost all metrics, except for slightly lower SRL and negation scores. The learning method itself, when properly tuned, gave us an improvement over the CAMR baseline. When looking at the concept confusions in Figure 2 we found that world knowledge helped reduce these errors. We also found false-positives reduced by 6% over the CAMR baseline. The entity link feature did not provide much improvement and it did not help the parser when combined with our other features.

Our analyses of gold standard NER features also revealed some of the limitations of using world knowledge with existing AMR parsers. The upperbound that we identified for this type of knowledge, while clearly above the performance of previous parsers, is still far below the expected performance when gold annotations are being used.

We also attempted to include other forms of world knowledge, encoded in the form of word embeddings (Mikolov et al., 2013) or node embeddings (Grover and Leskovec, 2016), which did not work as expected, and did not lead to improvements. This suggests that future research avenues in AMR parsing should instead focus on improvements in parsing algorithms or training data.

Acknowledgments

This material is based in part upon work supported by IBM under contract 4915012629. Any opinions, findings, conclusions or recommendations expressed above are those of the authors and do not necessarily reflect the views of IBM.

6. Bibliographical References

Banarescu, L., Bonial, C., Cai, S., Georgescu, M., Griffitt, K., Hermjakob, U., Knight, K., Koehn, P., Palmer, M., and Schneider, N. (2012). Abstract meaning representation (amr) 1.0 specification. In *https://www.isi.edu/ ulf/amr/help/amr-guidelines.pdf*, pages 1533–1544.

Barzdins, G. and Gosko, D. (2016). Riga at semeval-2016 task 8: Impact of smatch extensions and character-level neural translation on AMR parsing accuracy. In *Proceedings of SemEval*, pages 1143–1147.

Burns, G. A., Hermjakob, U., and Ambite, J. L. (2016). Abstract meaning representations as linked data. In *International Semantic Web Conference*, pages 12–20. Springer.

Cai, S. and Knight, K. (2013). Smatch: an evaluation metric for semantic feature structures. In *Proceedings of The 52nd Annual Meeting of the Association for Computational Linguistics*, pages 748–752.

Damonte, M., Cohen, S. B., and Satta, G. (2016). An incremental parser for abstract meaning representation. In *Proceedings of the 15th Conference of the European Chapter of the Association for Computational Linguistics*, pages 536–546.

Ferragina, P. and Scaiella, U. (2010). Tagme: on-the-fly annotation of short text fragments (by wikipedia entities). In *Proceedings of the 19th ACM international conference on Information and knowledge management*, pages 1625–1628. ACM.

Finkel, J. R., Grenager, T., and Manning, C. (2005). Incorporating non-local information into information extraction systems by gibbs sampling. In *Proceedings of the 43rd annual meeting on association for computational linguistics*, pages 363–370.

Flanigan, J., Thomson, S., Carbonell, J., Dyer, C., and Smith, N. A. (2014). A discriminative graph-based parser for the abstract meaning representation. In *Proceedings of the 52nd Annual Meeting of the Association for Computational Linguistics*, volume 1, pages 1426–1436.

Flanigan, J., Dyer, C., Smith, N. A., and Carbonell, J. (2016). CMU at semeval-2016 task 8: Graph-based AMR parsing with infinite ramp loss. In *Proceedings of SemEval*, pages 1202–1206.

Grover, A. and Leskovec, J. (2016). node2vec: Scalable feature learning for networks. In *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pages 855–864. ACM.

Kummerfeld, J. K., Berg-Kirkpatrick, T., and Klein, D. (2015). An empirical analysis of optimization for max-margin nlp. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing*, pages 273–279.

Liu, F., Flanigan, J., Thomson, S., Sadeh, N., and Smith, N. A. (2015). Toward abstractive summarization using semantic representations. In *Proceedings of The 54th Annual Meeting of the Association for Computational Linguistics*, pages 1077–1086.

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

Mikolov, T., Sutskever, I., Chen, K., Corrado, G. S., and Dean, J. (2013). Distributed representations of words and phrases and their compositionality. In *Advances in neural information processing systems*, pages 3111–3119.

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

Misra, D. K. and Artzi, Y. (2016). Neural shift-reduce CCG semantic parsing. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing*, pages 1775–1786.

Pan, X., Cassidy, T., Hermjakob, U., Ji, H., and Knight, K. (2015). Unsupervised entity linking with abstract meaning representation. In *Human Language Technologies: The 2015 Annual Conference of the North American Chapter of the ACL*, pages 1130–1139.

Sachan, M. and Xing, E. P. (2016). Machine comprehension using rich semantic representations. In *Proceedings of The 54th Annual Meeting of the Association for Computational Linguistics*, pages 486–492.

Wang, C., Xue, N., and Pradhan, S. (2015a). Boosting transition-based AMR parsing with refined actions and auxiliary analyzers. In *Proceedings of The 54th Annual Meeting of the Association for Computational Linguistics*, pages 857–862.

Wang, C., Xue, N., and Pradhan, S. (2015b). A transition-based algorithm for AMR parsing. In *Human Language Technologies: The 2015 Annual Conference of the North American Chapter of the ACL*, pages 366–375.

Wang, C., Pradhan, S., Xue, N., Pan, X., and Ji, H. (2016). CAMR at semeval-2016 task 8: An extended transition-based AMR parser. In *Proceedings of SemEval*, pages 1173–1178.

Zhou, J., Xu, F., Uszkoreit, H., Qu, W., Li, R., and Gu, Y. (2016). AMR parsing with an incremental joint model. In *Proceedings of SemEval*, pages 680–689.