Orange @ UMR Parsing Shared Task

Johannes Heinecke, Munshi Asadullah
Orange Research, 22300 Lannion
johannes.heinecke@orange.com, munshi.asadullah@orange.com

Abstract

Uniform Meaning Representation (UMR) is a novel meaning representation formalism emanating from Abstract Meaning Representation (AMR). Since it is more complex than AMR, including document level annotation it is more difficult to create a parsing pipeline which can predict an UMR document from a set of consecutive sentences. The UMR Parsing Shared Task was created to compare different approaches. We decided to use a 2-step approach to predict sentence level and document level annotation. Since the available data was limited, we opted for a multilingual model, even though unlike AMR, in UMR the concepts of the meaning graph are not drawn from a single source, but from language dependend resources. Our final score was 19.35%, 0.08 points behind the best participant (19.43%).

1. Introduction

Uniform Meaning Representation (UMR, Van Gysel et al., 2021) is an evolution and extension of Abstract Meaning Representation (AMR, Banarescu et al., 2013). In addition to semantic graphs present in AMR, UMR not only adds information not represented in AMR graphs like number and person, and alignments between tokens of the sentence and instances of the semantic graph, but also includes a document level annotation to represent the temporal relations of states and events (before, after, overlap, …) mentioned in a set of sentences. Finally UMR annotates coreferential entities and events crossing the sentence boundary. UMR is conceived to be multilingual. This means that the semantic graph contains concepts not only drawn from the English PropBank (Kingsbury and Palmer, 2002; Palmer et al., 2005) as in AMR, but from the language of the sentence.

UMR data comes in documents which contain a set of consecutive sentences, their semantic graphs, alignments and temporal and coreferential links between sentences. Optionally UMR files may also contain information about POS and even morphemes of the words of the sentences and translations.

The UMR Parsing Shared Task (Štěpánek et al., 2026) proposes to create UMR data from a (segmented and tokenised) text documents. Test data was in the six languages for which training data was available and a "surprise" language, Italian.

2. Data

The UMR data provided by the organisers contained UMR documents in 6 languages: Arapaho[1] (arp), Chinese (zh), Czech (cs), English (en), Latin (la) and Navajo[2] (nv), cf. Tables 1 and 2. Apart from Czech the data does not contain many sentences and a large part of it is considered "dirty", i.e. synthetic data without final human validation. For Latin no "clean" data was available at all.

	docs.	sent.	words	chars.	sents/ doc
arp	1	53	274	2 029	53.0
cs	6	103	1 587*	8 846	17.2
en	4	180	2 023	9 488	45.0
nv	1	5	60	494	5.0
zh	20	557	15 375	41 292	27.9

Table 1: Available "clean" training. *For Chinese the character count means number of ideograms.

	docs.	sent.	words	chars.	sents/ doc
arp	2	292	1 627	12 915	146.0
zh	1	1 435	18 656	42 129	1 435.0
cs	6 516	159 906	2 537 317	14 100 350	24.5
en	571	29 872	292 023	1 493 470	52.3
la	3	1 049	19 139	117 420	349.7
nv	3	337	2 663	19 709	112.3

Table 2: Available "dirty" training

3. Our Approach

In the past we got state-of-the-art results in multilingual AMR parsing (Heinecke and Shimorina, 2022) by finetuning Flan-T5-base (Chung et al., 2022) or mT5-base (Xue et al., 2021), the latter for languages other than English. Since we did not succeed in improving our scores by using larger

[1] Algonquian language spoken in the US states of Wyoming and Oklahoma

[2] Athabascan language spoken in the South West of the USA

Proceedings of The Seventh International Workshop on Designing Meaning Representations (DMR 2026) @ LREC 2026, pages 148–154
11 May 2026. ©ELRA Language Resources Association (ELRA), 2026

origin / destination	dirty dev	clean dev	dirty train	clean train
arp	1	0	1	1
cs	651	1	5 865	5
en	57	1	514	3
la	1	n/a	2	n/a
na	1	0	0	3
zh	0	1	1	19

Table 3: Split of "clean" and "dirty" documents into train and dev

LLM (like various version of Qwen2.5 (0.5B, 1.5B, 3B and 7B) and Qwen3 8B), unlike Chun and Xue (2024) we decided to try a two-staged approach for this task. A first step to get all intra-sentence information (the semantic graph, alignments, temporal relations and modal attribute roles), and a second step which tries to predict all inter-sentence related information (mainly temporal relations and coreferences).

Since UMR data differs considerably from AMR data, we did not use any of the AMR 3.0 data (Knight et al., 2020) and tried to build the entire pipeline using only the provided training data, including the "dirty" data. This means that we also refrained from using the corefences annotated partially in AMR 3.0. Another reason for not using AMR 3.0 data is the fact that it is only available for English, whereas in UMR the concepts of the graphs are drawn from the language of the sentence.

To train the 2 models for our two steps, we split the data in dev and train as shown in Table 4. In general all the clean dataset is used to finetune our models. If there is enough clean data, we put at least a clean document in the dev dataset. No document is in both, dev and train at the same time.

After playing with different finetunings we added some preprocessing

- replace relations `:refer-number` and `:refer-person` by `:person` and `:person` respectively,
- put all literals between quotes including special tokens (`3rd`, `full-affirmative`) (excepting numerical values, - or +),
- deleting `:wiki` relations (present 7 times in in the clean train set of Czech, twice Arapaho and Chinese, once in Navajo and 46 occurences in English). In the "dirty" training set only english and Navajo contain this relation. In addition, the `:wiki` relation points to wikidata entities in Czech and to Wikipedia pages in the other languages, for our multilingual this proved to be too complicated taking into account the available time.

dev

lang.	docs.	sents.	words	chars.
ar	1	59	329	3 040
cs	652	14 228	219 116	1 254 202
en	58	833	14 252	78 234
la	1	327	6 229	37 155
nv	1	50	344	2 474
zh	2	63	1 860	4 978

train

lang.	docs.	sents.	words	chars.
arp	2	286	1 572	11 904
cs	5 870	145 781	2 319 788	12 854 994
en	517	29 219	279 794	1 424 724
la	2	722	12 910	80 265
nv	3	292	2 379	17 729
zh	19	1 929	32 171	78 443

Table 4: number of documents, sentences etc. after our split in dev (top) and train (bottom)

We also detected some inconsistencies which we did not correct since the time frame of the shared task was too dense:

- named entities like `:op1 "Obama"` are included in the alignments in English (clean) but not in Chinese (clean)
- concepts like `s1x / %Rcp` found in Czech (dirty) (Left-overs from the translation of data taken from the Prague Dependency Treebank (PDT, Hajič et al. (2020))[3] UMR)

The exact size of the files per language after split is shown in Table 4.

Even though the size of the datasets is very small, we did not try to create synthetic data as proposed by Gamba et al. (2025) (starting from Universal Dependencies treebanks).

3.1. Sentence graph

The first step in our pipeline tries to predict all intra-sentence information: the graph, the alignments and the document level annotations unless a variable from another sentence is involved. To do so, we merge the graph from the training data and add all intra-sentence document level annotation by creating instances for tokens like "root" or "document-creation-time". Since the relations used in the document level annotation are different than the relations in the graph, they can be identified and extracted during the inference (cf. example in Fig. 1). We then finetuned 3 models

[3] https://ufal.mff.cuni.cz/prague-dependency-treebank

```
(v / publication-91
  :ARG1 (v2 / landslide-01
    :ARG3 (v3 / and
      :op1 (v4 / die-01
        :ARG1 (v5 / person
          :quant "200")
        :aspect "state")
      :op2 (v6 / fear-01
        :ARG1 (v7 / miss-01
          :ARG1 (v8 / person
            :quant "1500")
          :aspect "state")
        :aspect "state")
      :aspect "process")
    :place (v9 / country
       :name (v10 / name
         :op1 "Philippines"))
    :before-of
        (v11 / document-creation-time
           :overlap v6)
    :overlap v4
    :overlap v7)
  :root (v12 / root
    :modal (v13 / author
      :full-affirmative v2
      :full-affirmative v4
      :full-affirmative v6
      :partial-affirmative v7))
```

Figure 1: Sentence graph enriched by intra-sentence relations taken from document level annotation in bold. Temporal relations are bold and underlined, modal attribute roles are bold only (taken from clean training data `english_umr-0001.umr`)

to predict an enriched UMR graph from a simple sentence. Two models were only trained in monolingual data, even if it was very little: 1) Flan-T5-base 2) mt5-Base. The third model was trained on training data of all languages combined. To avoid a dominance of the Czech data, we took only 10% of the "dirty" Czech data. Fig. 2 details our training pipeline for the first step

Since Flan-T5 is mainly pretrained on English data, for all other languages the (monolingual) fine-tuning of mT5 resulted in much better results. In lack of the test-data we used our dev split to identify the best model. However, for the languages for which only very little data is available (Arapaho, Latin and Navajo) the multilingual model performed far better. Only for Czech with a big train set the monolingual model was better (Table 5). Note that we used the dev dataset to determine the best option. This is not optimal, especially since our dev set contains documents from the "dirty" dataset, but the official test set only contains clean data.

	monolingual		multilingual
	mT5	Flan-T5	mT5
Arapaho	36.12	34.85	**40.97**
Chinese	41.71	11.67	**49.70**
Czech	**86.30**	68.14	77.14
English	58.73	59.73	**60.86**
Latin	33.48	31.81	**48.95**
Navajo	31.76	27.77	**41.15**

Table 5: Training results to identify which model performs best for which language. Except for Czech, the multilingual model based on mT5 outperforms the other versions

A different problem is the alignment of instances of the sentence graph with the words (tokens) of the sentence. We have regarded this problem with the least priority due to time reasons. There exist aligners for AMR (such as AMRlib[4]), but we did not use this approach for two reasons: The obvious absence of training data and the fact that alignment in AMR is different from the one in UMR. In AMR tokens are not only aligned to instances, but also to literals (names, quantities) or to relations (e.g. the English preposition "by" can be aligned to an `:ARG0` relations in a Passive Voice construction).

3.2. Alignments

In order to have at least some – basic – alignments we opted for a guessing method: We use a Lemmatizer (UDParse[5]), trained on data from the Universal Dependencies project (UD)[6] and then map lemmas to instances of concepts. This fails of course when two different instances of the same concept appear in the sentence. In case of named entities, where the name itself is an attribute in the sentence graph, we add a postprocessing to find the instance of the concept having the name. For Navajo and Arapaho, no UD data is available. In this case our fallback is mapping forms to concepts. Due to the complex morphology of these two languages, the recall is very low.

3.3. Document level annotations

We employed a different approach to infer inter-sentence document level annotations. The training data suggested that most inter-sentence relations between instances are rarely more than six sentences apart. To finetune our model we used Qwen3 8B. We ran 3 experimental setups to resolve inter-sentence annotations, i.e. co-reference and temporal relations. For all 3 setups the data

[4] https://github.com/bjascob/amrlib
[5] https://github.com/Orange-OpenSource/UDParse
[6] https://universaldependencies.org

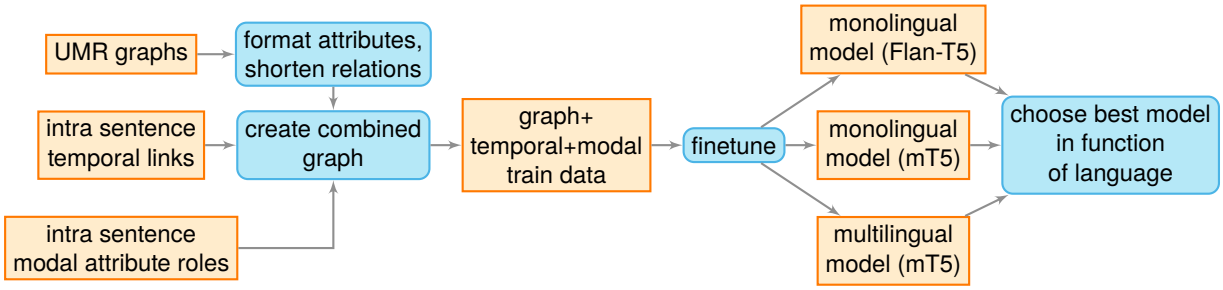

Figure 2: Schema of the training of the first step. For the multilingual model we combined all data of our train split (but only 10% of the Czech data)

was generated by creating sentence pairs for a maximum span of 6 on a sliding window, within a document as these relations do not exist on inter-document level. Each data point thus consists of the first sentence and its corresponding UMR graph, the second sentence and and its corresponding UMR graph. For setup 1 and 2 inter-sentence co-references were not exploited and for setup 3 we also provided the distance between the sentences to the training and prediction. We quickly realized the overwhelming number of data points without any co-references to exploit thus it produced a model that did not produce any positive outputs. So we introduced filtering and rejected all documents without any inter-sentence relations for the second setup. Although, improved, the best F value achieved on the validation data was 0.06. For the third setup, we kept the filter and included inter-sentence relations and balanced the number of positive and negative samples. On validation data we achieved 0.48 F value. A schema of the training flow is shown in Fig. 3.

4. Inference of the official test data

The test data provided by the organisers consists of 30 documents with in total 1019 sentences (cf. Table 6). In addition to the languages already available for training, one document was in Italian (it).

lang.	docs.	sents.	words	chars.	sents./ doc
arp	2	55	274	2 076	27.5
cs	5	220	4 048	26 042	44.0
en	5	195	4 092	22 084	39.0
it	1	100	2 212	12 135	100.0
la	1	50	889	5 554	50.0
nv	1	163	1 194	12 911	163.0
zh	15	236	6 467	37 998	15.7
total	30	1 019	19 176	23 668	

Table 6: test data size

The data flow is as shown in Fig. 4. As said above, the models of the first step predict the sentence graph with the temporal and modal document level annotations (unless the relation contains an instance of a preceding sentence). The graph is then split into the UMR graph and the document level annotations. The sentence and the UMR graph is the input for the second step (dashed box in Fig. 4). Alignments are added at the same moment (cf. section 3.2). Unfortunately, we could not generate any inter-sentence document level annotations. We suppose that our approach was not adapted to the scarcity of the training data.

The final step is the generation of the UMR output file. Here we also delete quotes around special tokens like `3rd` or `full-affirmative` and rename the relations `:number` and `:person` back to their official form `:refer-number` and `:refer-person` respectively. Due the the little training data, our mT5 finetuned models tend to output duplicate relations, which we also remove. Since formal errors in the UMR file will mean a score of 0, we used the provided validation script to spot formal error (missing instances in alignments, cycles in graphs) and repair them automatically.

5. Results on official test data

Our results are shown in Tables 7 and 8. Unsurprisingly both Arapaho documents score very badly, most likely due to the little data available for training and absence of any Arapaho data in the pretrained mT5 mode which we finetuned. Interestingly the Navajo document scores much better, even though the clean training data was even smaller than for Arapaho (but dirty training data was slightly bigger). Interestingly our simple aligner worked rather well for Chinese (70%) and English (71.4%) but failed completely for Italian (18.7%). Even the morphologically complex languages of Arapaho (40/7%) and Navajo (48.2%) scored better. The modal and temporal document level annotation were 0 for all languages but Chinese. And we failed to predict a single coreference.

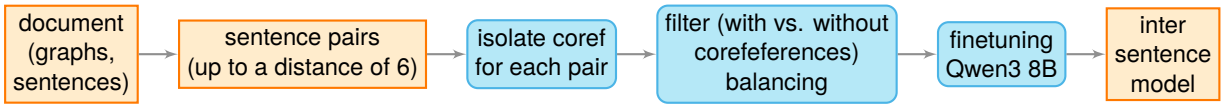

Figure 3: Finetuning of the inter-sentence document level annotations

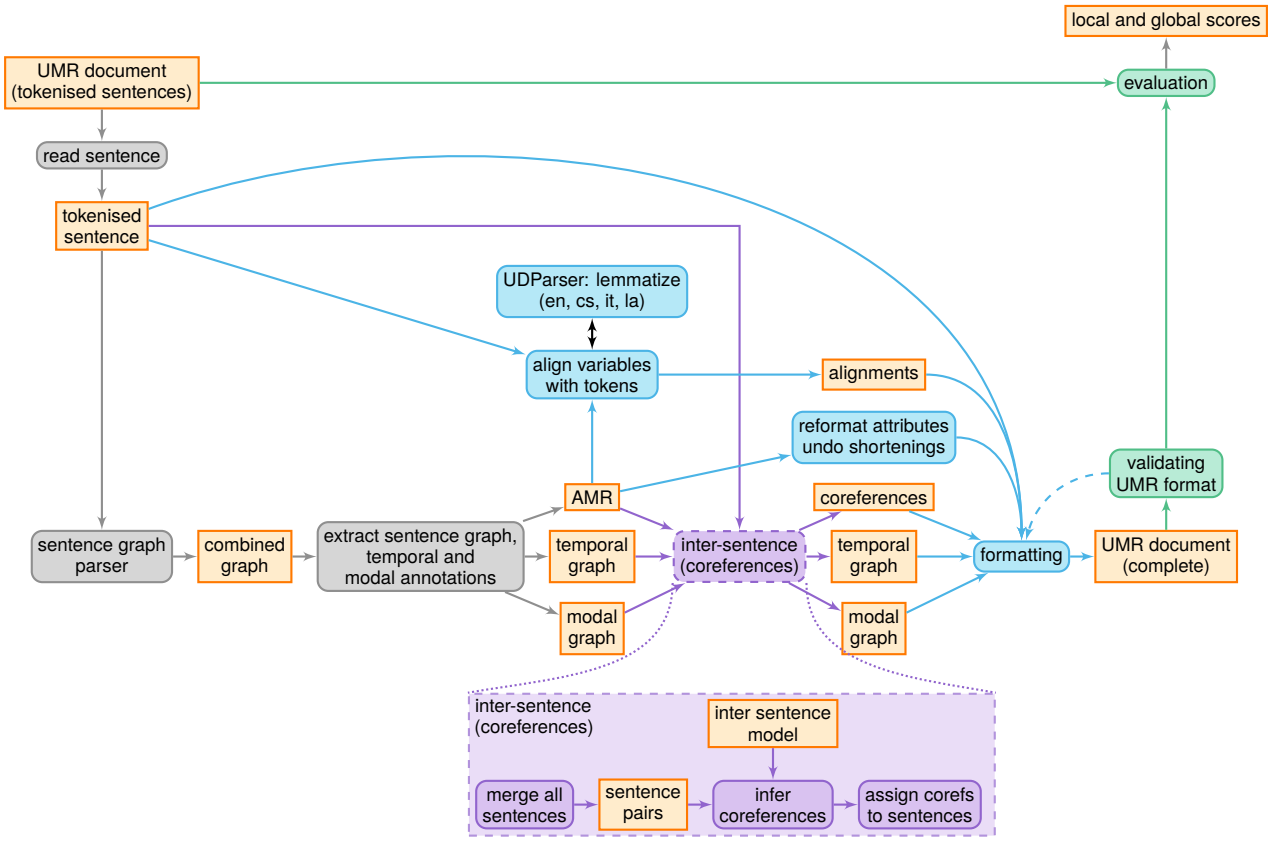

Figure 4: Flow diagramme for inference (the large dashed box at the bottom is a zoom on the inter-sentence inference)

In a post-Shared-Task experiment we corrected a bug in the prediction of coreference so that we finally got coreferences, but this did not change the global score.

English and Chinese score best in our case. Both languages are well represented in mT5. Additionally in the case of Chinese only one document of the training data was “dirty” which obviously improved the parsing result.

6. Conclusion and perspective

We presented our approach to the UMR shared task where we were able to finish in with a global score of 19.35%. Due to this, in absolute terms low score, the UMR parsing is currently not exploitable since there are too many errors, even in major languages like English, Chinese or Czech. Apart from needing more high-quality data, inconsistencies between languages and annotations in the existing data must be resolved.

7. Bibliographical References

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.

Jayeol Chun and Nianwen Xue. 2024. Uniform Meaning Representation Parsing as a Pipelined Approach. In *Proceedings of TextGraphs-17:*

testfile	F1 score	testfile	F1 score
arp-0005	5.64%	zh-0032	29.44%
arp-0003	8.76%	zh-0036	29.68%
cs-0004	13.75%	en-0006	29.97%
it-0000	13.94%	zh-0026	33.32%
cs-0000	14.45%	zh-0028	34.79%
cs-0003	15.50%	zh-0024	35.36%
en-0000	15.90%	zh-0027	35.72%
cs-0001	16.29%	zh-0030	35.99%
la-0001	17.24%	zh-0034	39.39%
cs-0002	20.65%	zh-0023	40.29%
nv-0004	21.55%	en-0008	41.09%
en-0007	24.18%	zh-0029	41.47%
en-0005	24.34%	zh-0031	41.67%
zh-0021	25.31%	zh-0035	45.54%
zh-0022	28.61%	zh-0033	48.21%

Table 7: Our results per test file, sorted by score (average: 27.49%)

language	F1 score
Arapaho	8.15%
Italian	13.94%
Czech	15.87%
Latin	17.24%
Average	19.00%
Navajo	21.55%
English	22.19%
Chinese	36.51%

Table 8: Our results per language, sorted by score (average: 19.35%)

Graph-based Methods for Natural Language Processing, pages 40–52, Bangkok, Thailand. Association for Computational Linguistics.

Hyung Won Chung, Le Hou, Shayne Longpre, Barret Zoph, Yi Tay, William Fedus, Yunxuan Li, Xuezhi Wang, Mostafa Dehghani, Siddhartha Brahma, Albert Webson, Shixiang Shane Gu, Zhuyun Dai, Mirac Suzgun, Xinyun Chen, Aakanksha Chowdhery, Alex Castro-Ros, Marie Pellat, Kevin Robinson, Dasha Valter, Sharan Narang, Gaurav Mishra, Adams Yu, Vincent Zhao, Yanping Huang, Andrew Dai, Hongkun Yu, Slav Petrov, Ed H. Chi, Jeff Dean, Jacob Devlin, Adam Roberts, Denny Zhou, Quoc V. Le, and Jason Wei. 2022. Scaling instruction-finetuned language models.

Federica Gamba, Alexis Palmer, and Daniel Zeman. 2025. Bootstrapping UMRs from Universal Dependencies for Scalable Multilingual Annotation. In *Proceedings of the 19th Linguistic Annotation Workshop (LAW-XIX-2025)*, pages 126–136, Vienna, Austria. Association for Computational Linguistics.

Jan Hajič, Eduard Bejček, Jaroslava Hlavacova, Marie Mikulová, Milan Straka, Jan Štěpánek, and Barbora Štěpánková. 2020. Prague dependency treebank - consolidated 1.0. In *Proceedings of the Twelfth Language Resources and Evaluation Conference*, pages 5208–5218, Marseille, France. European Language Resources Association.

Johannes Heinecke and Anastasia Shimorina. 2022. Multilingual Abstract Meaning Representation for Celtic Languages. In *Proceedings of the 4th Celtic Language Technology Workshop within LREC2022*, pages 1–6, Marseille. ELRA.

Paul Kingsbury and Martha Palmer. 2002. From TreeBank to PropBank. In *Proceedings of the Third International Conference on Language Resources and Evaluation*, pages 1989–1993, Las Palmas, Canary Islands - Spain. European Language Resources Association.

Martha Palmer, Daniel Gildea, and Paul Kingsbury. 2005. The Proposition Bank: An Annotated Corpus of Semantic Roles. *Computational Linguistics*, 31(1):71–106.

Jens E. L. Van Gysel, Meagan Vigus, Jayeol Chun, Kenneth Lai, Sarah Moeller, Jiarui Yao, Tim O'Gorman, Andrew Cowell, William Croft, Chu-Ren Huang, Jan Hajič, James H. Martin, Stephan Oepen, Martha Palmer, James James Pustejovsky, Rosa Vallejos, and Nianwen Xue. 2021. Designing a Uniform Meaning Representation for Natural Language Processing. *Künstliche Intelligenz*, 35:343–360.

Linting Xue, Noa Constant, Mihir Kale, Rami Al-Rfou, Aditya Siddhant, Aditya Barua, and Colin Raffel. 2021. mT5: A Massively Multilingual Pre-trained Text-to-Text Transformer. In *Conference of the North American Chapter of the Association for Computational Linguistics*, pages 483–498. Association for Computational Linguistics.

Jan Štěpánek, Daniel Zeman, Markéta Lopatková, Federica Gamba, Hana Hledíková, and Nianwen Xue. 2026. First shared task on UMR parsing. In *Proceedings of the Seventh International Workshop on Designing Meaning Representations*, Palma, Spain. ELRA.

8. Language Resource References

Kevin Knight and Bianca Badarau and Laura Baranescu and Claire Bonial and Madalina Bardocz and Kira Griffitt and Ulf Hermjakob and Daniel Marcu and Martha Palmer and Tim

O'Gorman and Nathan Schneider. 2020. *Abstract Meaning Representation (AMR) Annotation Release 3.0*. Linguistic Data Consortium. distributed via LDC: LDC2020T02, 3.0, ISLRN 676-697-177-821-8.