Bootstrapping Multilingual AMR with Contextual Word Alignments

Janaki Sheth** **Young-Suk Lee*** **Ramón Fernandez Astudillo*** **Tahira Naseem***

Radu Florian* **Salim Roukos*** **Todd Ward***

**Perelman School of Medicine, UPenn, Philadelphia, PA, USA
*IBM Research, Yorktown Heights, NY, USA
Janaki.Sheth@Pennmedicine.upenn.edu, ramon.astudillo@ibm.com
{ysuklee, tnaseem, raduf, roukos, toddward}@us.ibm.com

Abstract

We develop high performance multilingual Abstract Meaning Representation (AMR) systems by projecting English AMR annotations to other languages with weak supervision. We achieve this goal by bootstrapping transformer-based multilingual word embeddings, in particular those from cross-lingual RoBERTa (XLM-R large). We develop a novel technique for foreign-text-to-English AMR alignment, using the contextual word alignment between English and foreign language tokens. This word alignment is weakly supervised and relies on the contextualized XLM-R word embeddings. We achieve a highly competitive performance that surpasses the best published results for German, Italian, Spanish and Chinese.

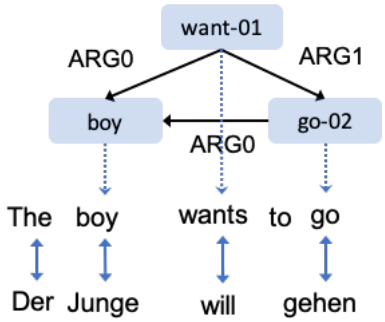

Figure 1: AMR graph for *The boy wants to go* and its German translation *Der Junge will gehen*. Implicit alignments between the English text and AMR concepts are denoted by dotted arrows. Explicit alignments between English and German texts are denoted by solid arrows.

1 Introduction

Abstract Meaning Representation graphs are rooted, labeled, directed, acyclic graphs representing sentence-level semantics (Banarescu et al., 2013). In the example shown in Figure 1, the sentence *The boy wants to go* is parsed into an AMR graph. The nodes of the AMR graph represent the AMR concepts, which may include normalized surface symbols e.g. *boy*, Propbank frames (Kingsbury and Palmer, 2002) e.g. *want-01*, *go-02* as well as other AMR-specific constructs. Edges in an AMR graph represent the relations between concepts. In this example *:arg0*, *:arg1* correspond to standard roles of Propbank.

One distinctive aspect of AMR annotation is the lack of explicit alignments between nodes in the graph and words in the sentences. Since such alignments are essential for training many of present-day AMR parsers, there have been various efforts to link the AMR concepts to their corresponding span of words (Flanigan et al., 2014; Pourdamghani et al., 2014; Lyu and Titov, 2018; Chen and Palmer, 2017). A significant emphasis of this paper is on deriving these alignments for multilingual AMR parsers.

Even though by nature AMR is biased towards English, recent work has evaluated the potential of AMR to work as an interlingua. Hajič et al. (2014) and Xue et al. (2014) categorize and propose refinements for divergences in the annotation between English and Chinese as well as Czech AMRs. Anchiêta and Pardo (2018) import the corresponding AMR annotation for each sentence from the English annotated corpus and revisit the annotation to adapt it to Portuguese. However, Damonte and Cohen (2018) show that it may be possible to use the original AMR annotations devised for English as representation for equivalent sentences in other languages without any modification despite the translation divergence. This defines the problem of multilingual AMR parsing that we seek to address in this paper - given a sentence in a foreign language, recover the AMR graph originally designed for its English translation. We implement

* This research was done during an internship at IBM Research AI.

Proceedings of the 16th Conference of the European Chapter of the Association for Computational Linguistics, pages 394–404
April 19 - 23, 2021. ©2021 Association for Computational Linguistics

multilingual AMR parsers for German, Spanish, Italian and Chinese.

In this paper we propose that transformer-based multilingual word embeddings can be a useful tool for addressing the problem of multilingual AMR parsing. Besides using contextual word embeddings as input token embeddings, we leverage them for *annotation projection*, where existing AMR annotations for English are projected to a target language by using contextual word alignments. In our experiments, we employ XLM-RoBerta large (Conneau et al., 2019) as the multilingual pre-trained transformer model. We show that our proposed procedure achieves competitive results as some of the classical methods for text-to-AMR alignment. Furthermore, such a procedure is easily scalable to the 100 languages that XLM-R is trained on.

We also combine different techniques for concept alignments and AMR parser training which significantly improve performance over the base models. For concept alignment, we combine the proposed contextual word alignments with previously established alignment techniques utilizing matching rules tailored to AMR as well as machine translation aligners (Flanigan et al., 2014; Pourdamghani et al., 2014). For AMR parser training, we pre-train an AMR parser on the treebanks of different languages simultaneously and subsequently finetune on each language. This is analogous to the techniques used for silver data pre-training (Konstas et al., 2017; van Noord and Bos, 2017) in AMR parsing and multi-lingual pre-training (Aharoni et al., 2019) in machine translation.

Finally, we conduct a detailed error analysis of the multilingual AMR parsing. One of the major errors we have found involves synonymous concepts, which share the same meaning as the original concepts in English, but differ in spellings. While this error is mainly caused by the fact that the multilingual word embeddings bridge non-English input tokens to English concepts, it also highlights the highly lexical nature of Smatch scoring (Cai and Knight, 2013) which does not take synonymous concepts into consideration. We also elaborate upon error analysis of the direct comparison between our proposed annotation projection method using contextual word alignment and a previous baseline, using fast align.

The rest of the paper is organized as follows: In Section 2, we discuss related work. In Section 3, we present our main proposal on annotation projection based on contextual word alignments. In Section 4, we describe various combination approaches that improve the multilingual parser performances significantly. These include combining word-to-concept alignments, using multi-lingual treebanks and combining human-annotated and synthetic treebanks. In Section 5, we discuss experimental results. In Sections 6 and 7, we present detailed error analyses. We conclude the paper in Section 8.

2 Related work

Multilingual AMR. There have been significant advances in AMR parsing for languages other than English. Previous studies (Hajič et al., 2014; Xue et al., 2014; Migueles-Abraira et al., 2018; Sobrevilla Cabezudo and Pardo, 2019) investigated AMR annotations for a variety of different languages such as Chinese, Czech, Spanish and Brazilian Portuguese. Vanderwende et al. (2015) automatically parse the logical representation for sentences in Spanish, Italian, German and Japanese, which is then converted to AMR using a small set of rules.

While much of this work, along with studies such as Li et al. (2016); Anchiêta and Pardo (2018), produces AMR graphs whose nodes were labeled with words from the target language, Damonte and Cohen (2018) developed AMR parsers for English and used parallel corpora for annotation projection to train Italian, Spanish, German, and Chinese parsers that recover the AMR graph originally designed for the English translation. Their main results showed that the new parsers can overcome certain structural differences between languages.

Similar to Damonte and Cohen (2018), we also train multilingual AMR parsers by projecting English AMR annotation to target foreign languages (German, Spanish, Italian and Chinese), but we depart from their approach in the specifics of the annotation projection by exploring contextual word alignments directly derived from multilingual contextualized word embeddings. While both procedures utilize parallel corpora, the annotation projection of Damonte and Cohen (2018) requires additional supervised training of their statistical word aligner. Our proposed contextualized word alignment is however unsupervised in nature. Alternatively, a recent study by Blloshmi et al. (2020) showed that one may in fact not need alignment-based parsers for cross-lingual AMR, rather modelling concept identification as a *seq2seq* problem. In this paper, we will compare our results to both

Damonte and Cohen (2018) and Blloshmi et al. (2020).

Word vector alignment techniques. Traditional word alignment methods often use parallel corpora and IBM alignment models (Brown et al., 1990, 1993) as well as improved versions (Och and Ney, 2003; Dyer et al., 2013). More recently, there have been an advent of techniques that align vector representation of words from varying levels of supervision (Ruder et al., 2019). Often word vectors are learned independently for each language and then a mapping from source language vectors to target language vectors with a bilingual dictionary is developed (Mikolov et al., 2013; Smith et al., 2017; Artetxe et al., 2017). To reduce the need for bilingual supervision, the iterative method of starting from a minimal seed dictionary and alternating with learning the linear map was employed by a recent body of work (Conneau et al., 2018; Schuster et al., 2019; Artetxe et al., 2018).

The work most similar to ours is Cao et al. (2020) where the authors obtain contextual embedding alignments from multilingual BERT (Devlin et al., 2018; Pires et al., 2019) and subsequently improve the alignments via finetuning using supervised parallel corpora. Our contextual word alignment between two parallel sentences may be thought of as an adaptation of their contextual word retrieval task. However, we refrain from any finetuning of the contextual embeddings and show that the contextual word alignments from the off-the-shelf XLM-R model achieves results competitive to the word alignments by fast-align (see Damonte and Cohen (2018)). This suggests the potential for inexpensive, massive scaling of AMR parsing up to 100 languages on which XLM-R is trained.

3 Annotation projection

We adopt a transition-based parsing approach for AMR parsing following (Ballesteros and Al-Onaizan, 2017; Naseem et al., 2019; Fernandez Astudillo et al., 2020). These produce an AMR graph g from an input sentence s by predicting instead an action sequence a from s as a sequence to sequence problem. This action sequence applied to a state machine M produces then the desired target graph as $g = M(a, s)$. Transition-based parsers require the action sequence for each graph in the training data. This is determined by a rule-based oracle $a = O(g, s)$ which relies on external word-to-node alignments. In all the subsequent experiments we will use the oracle and action set from (Fernandez Astudillo et al., 2020).

3.1 Projection method

In order to train AMR parsers in a non-English language, we use the annotation projection method to leverage existing English AMR annotation and overcome resource shortage in the target language. First, the English text is aligned to corresponding AMR concepts using both rule-based JAMR aligner (Flanigan et al., 2014) and a IBM model type aligner (Pourdamghani et al., 2014). The latter will henceforth be referred to as the EM aligner. Given the English text-to-AMR concept alignments, we then project these to the target language using word alignment. In the following subsection we describe in the proposed word alignment method, called *contextual word alignment*, which is trained in a weakly supervised manner.

3.2 Contextual word alignments

Given two languages, we align word pairs within parallel sentences if their vector representations derived from the underlying multilingual pre-trained model are similar according to cosine distance. As vector representation we use the average of all 24 layers of the XLM-R large contextual embeddings. We will refer to this average as the word's contextual embedding henceforth for simplicity.

More precisely, suppose we have two parallel sentences - $\mathbf{E} = e_0, e_1, e_2, ..., e_M$ in English and $\mathbf{F} = f_0, f_1, f_2, ..., f_N$ in the target language. We will use r to represent the pre-trained multilingual model such that $r(\mathbf{S})_i$ is the contextual embedding for the i^{th} word in sentence $\mathbf{S}$. Then a word $e_i \in \mathbf{E}$ is contextually word aligned to f_j if and only if the cosine similarity score between their word embeddings is the highest. Thus we define the corresponding contextual alignment function $\chi(f_j|e_i)$ as,

$$\chi(f_j|e_i) = \text{argmax}_{0 \leq j \leq |\mathbf{F}|} cos(r(\mathbf{E})_i, r(\mathbf{F})_j). \quad (1)$$

Similarly, performing the same procedure in the reverse direction we have,

$$\chi(e_i|f_j) = \text{argmax}_{0 \leq i \leq |\mathbf{E}|} cos(r(\mathbf{F})_j, r(\mathbf{E})_i) \quad (2)$$

While these methods can be noisy, by only keeping word pairs in their intersection *i.e.* $\chi(\mathbf{E}|\mathbf{F}) \cap \chi(\mathbf{F}|\mathbf{E})$, one can derive the intersection cosine alignment approach which gives us a word-aligned dataset with low coverage but high accuracy.

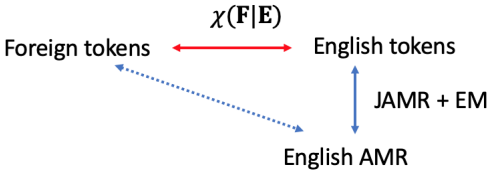

Figure 2: Annotation projection is achieved using JAMR and EM aligners for English text-to-AMR concept alignment and contextual word alignment between tokens of the source (English) and target languages.

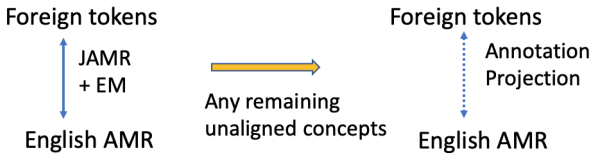

Figure 3: Illustration of the EM, JAMR + A.P. combination alignment: first align target tokens to AMR concepts using JAMR+EM aligners with any remaining concepts then aligned using the annotation projection method proposed in Figure 2.

As an example, the following are sentences from our German and English training datasets:

E: Establishing models in industrial Innovation
F: Etablierung von Modellen in der industriellen Innovation
Their contextual word alignments are,
$\chi(\mathbf{F}|\mathbf{E})$ = $[(e_0, f_0), (e_1, f_2), (e_2, f_3), (e_3, f_5), (e_4, f_6)]$
$\chi(\mathbf{E}|\mathbf{F})$ = $[(f_0, e_0), (f_1, e_1), (f_2, e_1), (f_3, e_2), (f_4, e_2), (f_5, e_1), (f_6, e_4)]$
$\chi(\mathbf{F}|\mathbf{E}) \cap \chi(\mathbf{E}|\mathbf{F})$
= $[(e_0, f_0), (e_1, f_2), (e_2, f_3), (e_4, f_6)]$

Figure 2 pictorially illustrates our complete annotation projection method using the contextual word alignment $\chi(\mathbf{F}|\mathbf{E})$. English tokens and AMR concepts are aligned using JAMR and EM aligners. The resulting AMR annotation augmented with English word-to-concept alignments is then projected onto the given target language using contextual word embeddings. Henceforth, for brevity we will at times refer to this approach as A.P.

4 Combination approaches

We apply three types of combination techniques to the multilingual AMR parsers, trained by projecting English annotations using contextual word alignments derived from the multilingual contextual word embeddings, each of which improves the parser performance significantly.

4.1 Alignment combination

One such technique is to combine the contextual word alignment based A.P. with the baseline word-to-concept alignment which aligns the target tokens directly to AMR concepts using JAMR and EM aligners. Since the EM aligner is an unsupervised method, it can be directly applied to the target language tokens and English AMR concepts. However, we note that this baseline alignment approach gives incomplete coverage (87% concepts aligned to German, 88% to Italian and 91% to Spanish tokens). Thus, we supplement this by aligning the remaining concepts using the A.P. of Figure 2.

For example, suppose we have as before two parallel sentences - $\mathbf{E} = e_0, ..., e_M$ in English and $\mathbf{F} = f_0, ..., f_N$ in the target language, as well as AMR concepts $\mathbf{N} = n_0, ..., n_L$. Then one of our proposed foreign text-to-AMR concept combination alignment procedures $EA(f_i|n_j)$ (see Figure 3) is defined as,

$$EA(f_i|n_j) = AP(BA(f_i|n_j)) \quad (3)$$

where $BA(f_i|n_j)$ represents that the j^{th} concept is aligned to the i^{th} token in $\mathbf{F}$ using the baseline aligner BA. If for any concept $n_j \in \mathbf{N}$, $BA(f_i|n_j) =$ None, we use annotation projection to align it where $AP(f_i|n_j)$ is given by,

$$\chi(f_i|e_k) \wedge BA(e_k|n_j) \Rightarrow AP(f_i|n_j) \quad (4)$$

We also experiment with other such alignments, in particular by using the intersection of cosine alignment ($\chi(F|E) \cap \chi(E|F)$) as the contextual word alignment. In this case,

$$EA(f_i|n_j) = \max AP(BA(\mathrm{i}AP(f_i|n_j))) \quad (5)$$

wherein,

$$(\chi(f_i|e_k) \cap \chi(e_k|f_i)) \wedge BA(e_k|n_j) \Rightarrow \mathrm{i}AP(f_i|n_j) \quad (6)$$

As before, $\forall n_j \in \mathbf{N}$ where $\mathrm{i}AP(f_i|n_j) =$ None we align it using the baseline aligner $BA(f_i|n_j)$. For any further remaining unaligned concepts, we employ $\max AP(f_i|n_j)$ which can be described as:

$$\max(\chi(f_i|e_k), \chi(e_k|f_i)) \wedge BA(e_k|n_j) \Rightarrow \max AP(f_i|n_j) \quad (7)$$

That is, we pick the uni-directional contextual word alignment with the higher score and project the AMR annotation accordingly.

4.2 Multilingual treebank combination

In addition to training the parser on the treebank of each language - derived from English treebank via annotation projection - we also experiment with combining all the target language treebanks to create a single multilingual treebank. We notice that pre-training an AMR parser on this multilingual treebank with subsequent finetuning on the treebank of each language, improves performance over the parser trained only on each individual treebank.

4.3 Human and synthetic treebank combination

We create a synthetic AMR corpus by parsing 85k unlabeled sentences from the context portion of SQuAD-2.0. The resulting synthetic AMR graphs are filtered as per the procedure in (Lee et al., 2020) and combined with the AMR-2.0 training set (LDC2017T10), to produce an expanded *AMR-2.0 + SQuAD* training dataset of 94k sentences. We then project annotations of this expanded English treebank onto each of the target languages, and train the corresponding target language parser. We observe that despite the lower quality of the synthetic AMRs as compared to their human-annotated counterparts, their inclusion in the training set significantly improves parser performance.

5 Experimental Results

5.1 AMR Parser and Data

For our experiments, we use the stack-Transformer model (Fernandez Astudillo et al., 2020)[1] as our AMR parser. The stack-Transformer is a transition based parser with a modified Transformer architecture to encode the parser state. It uses a cross entropy loss function and has hyper-parameters similar to those of machine translation described in (Vaswani et al., 2017). We use a beam size of 3 to decode our models and evaluate them using Smatch scores (Cai and Knight, 2013). Model performance values in this manuscript are an average over the best performing models across 3 random seeds. Lastly, the input to the parser - the vector representation of each word - is obtained by averaging over not only all 24 layers of the pre-trained XLM-R large contextual embeddings but also over constituent wordpieces within each word.

For all four languages - German, Spanish, Italian and Chinese - we experiment on AMR1.0 (LDC2015E86). For the first three we also experiment on AMR-2.0 (LDC2017T10). Results from the former are compared to Damonte and Cohen (2018) and from the latter to Blloshmi et al. (2020). Details of our training, dev and test sets are given in Table1.[2] To train each target language parser, we first translate the input sentences of AMR-2.0 and AMR-1.0 with Watson Language Translator.[3] This creates the supervised parallel corpus which we then use for our unsupervised annotation projection via contextual word alignment. We also align target language tokens directly to AMR concepts using JAMR and EM aligners for baseline system evaluation and for combination alignments. We select the best performing models using the devset. Finally, for our best models, we report results using the machine as well as human translations (LDC2020T07) of the test sets.

5.2 Baselines

Our first baseline is zero-shot learning, where we train on the English dataset but test on a foreign language dev-set (Baseline I). The reason behind this experiment is to test the ability of the XLM-R contextual word embeddings to capture the meaning of the given token irrespective of the underlying language. Note that it is only for this experiment that languages for the train and dev sets differ. In another set of experiments we align the target language tokens directly to the AMR concepts only using the JAMR and EM aligners (Baseline II). Lastly, we also test the annotation projection procedure of Damonte and Cohen (2018). Note that while the previous authors use fast align (Dyer et al., 2013) for word alignment between the parallel data and only JAMR aligner for the English text-to-AMR alignment, in Baseline III we have utilized fast align in conjunction with both JAMR and EM aligners (for English text-to-AMR alignment) for improved performance.

5.3 Results

Table 2 compares our different proposed approaches to the three baseline methods using the AMR2.0 and AMR1.0 datasets. We see that our proposed approach - annotation projection with contextual word alignment, in this case using $\chi(\mathbf{F}|\mathbf{E})$ - shows fairly competitive results with

[1] https://github.com/IBM/transition-amr-parser

[2] Word segmentation is applied to the Chinese raw texts for model training and testing.

[3] https://www.ibm.com/watson/services/language-translator/

Data set	Experiment	Number of sentences	Number of tokens			
			DE	ES	IT	ZH
Train set	AMR2.0 LDC	36k	677k	694k	654k	
	AMR2.0 LDC + synAMR	94k	2.1m	2.2m	2.1m	
	AMR1.0 LDC	10k	222k	240k	227k	195k
Development set	All experiments	1368	30k	32k	31k	26k
Test set	All experiments	1371	31k	33k	32k	27k

Table 1: Details of our dataset

Model	AMR2.0			AMR1.0			
	DE	ES	IT	DE	ES	IT	ZH
Baseline I (zero-shot)	39.0	39.6	41.0	37.4	38.8	39.3	33.4
Baseline II	61.4	66.2	68.3	57.2	60.3	60.7	55.4
Baseline III	63.8	68.7	68.6	56.3	60.8	61.0	54.7
Annotation Projection (A.P)	61.9	67.7	66.8	55.7	60.7	60.5	46.5
EM,JAMR+A.P	63.9	68.7	69.8	57.7	62.3	62.5	55.8
Intersect A.P+EM,JAMR+max(A.P)	64.2	69.1	68.7				
EM,JAMR+A.P (Multilingual)	64.6	69.2	70.4	**58.6**	**62.7**	**62.9**	**58.1**
EM,JAMR+A.P (synAMR)	**67.8**	**71.3**	**72.2**				

Table 2: Dev set Smatch for AMR2.0 and AMR1.0.

Model	Machine translation				Human translation			
	DE	ES	IT	ZH	DE	ES	IT	ZH
Damonte and Cohen (2018)					39	42	43	35
Baseline I (zero-shot)	37.1	37.99	38.5	31.8	36.3	37.6	37.4	30.2
Baseline II	56.1	58.94	59.7	53.3	53.6	57.8	56.8	48.3
Baseline III	55.1	59.24	59.0	53.1	52.7	57.9	57.3	48.1
Annotation Projection (A.P)	54.9	58.9	59.4	44.6	52.7	57.7	57.0	41.4
EM,JAMR + A.P	56.4	60.6	61.3	54.0	53.6	59.2	58.6	48.3
EM,JAMR + A.P (Multilingual)	**57.4**	**61.4**	**61.6**	**55.7**	**54.5**	**60.1**	**59.0**	**50.3**

Table 3: Test set Smatch for AMR1.0.

Model	Machine translation			Human translation			
	DE	ES	IT	DE	ES	IT	ZH
Blloshmi et al. (2020)				53	58	58.1	43.1
EM, JAMR + A.P (Multilingual)	63.8	67.7	69.0	59.9	66.0	65.7	
EM, JAMR + A.P (synAMR)	**66.9**	**69.6**	**71.0**	**62.7**	**67.9**	**67.4**	

Table 4: Test set Smatch for AMR2.0.

those of Baseline III for the target languages of German, Italian and Spanish, especially when applied to the smaller corpus of AMR1.0. This is remarkable considering our method requires no additional training and can be easily generalized for zero-shot learning on all different languages that XLM-R was pretrained on. We then train several parsers using our suggested combination approaches. The first such method comprises of both the EM, JAMR + A.P aligners (see Eq. 3). In a different approach, we use the intersection cosine word alignment based annotation projection (*i.e* $\chi(\mathbf{F}|\mathbf{E}) \cap \chi(\mathbf{E}|\mathbf{F})$). Since this leaves many AMR concepts unaligned, we follow it by aligning concepts using the baseline JAMR and EM aligners. Any leftover unaligned concepts are then aligned using $\max(\chi(\mathbf{E}|\mathbf{F}), \chi(\mathbf{F}|\mathbf{E}))$ (Eq. 5). In another set of experiments, we pre-train a parser on a multilingual treebank, where the train set is a combination of the LDC treebank in all target languages. The parser is then finetuned on each individual language. We surmise that such an experiment will give us a truly multilingual parser capable of successfully decoding all the target languages. Its strength is evident in its performance, it outperforms all our baseline approaches - in the case of AMR1.0 dev set by at least 1.4 points. Finally, in the last two experiments on AMR2.0 we train on the language-specific LDC + SQuaD train set. We see that this gives us our best performing parsers, where the training data is aligned using a combination (EM, JAMR + A.P) alignment.

We test a subset of the AMR2.0 and all of the AMR1.0 models on corresponding test sets. The results are shown in Tables 3 and 4. For AMR1.0, while all of our models including the baselines outperform previously published results, the best performing model is the parser which was trained on multilingual data and whose training input text was aligned to its AMR concepts using the combination of EM, JAMR and A.P aligners. For AMR2.0, models trained on the LDC + SQuAD dataset outperform those trained on multilingual data. Both of these outperform the recently published work of Blloshmi et al. (2020). [4]

We note that the parser performs better on the machine translated test data than on the human translated data. This should be attributed to the training and testing condition mismatch of the human translated test data since all models are trained on machine translated training data. For instance, the out-of-vocabulary (oov) ratio of the human translated test data is consistently higher than that of the machine translated test data. For example, for AMR1.0 the oov ratio of human translated test data vs. machine translated test data is 10.2% vs. 9% for German, 7.3% vs. 6.8% for Spanish, 8.1% vs. 7.6% for Italian and 7.6% vs. 5.5% for Chinese.

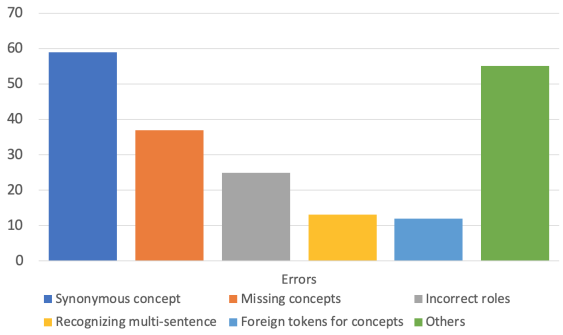

Figure 4: Histogram of different kinds of errors

6 Error analysis

We carried out an error analysis of 56 German sentences parsed by the best performing model trained on the combination of AMR2.0 and SQuAD training data. Statistics of the various errors are depicted in Figure 4. Top 5 most frequent errors include (i) introduction of synonymous concepts, (ii) missing concepts, (iii) incorrect roles, (iv) target tokens in AMR concepts, (v) incorrect parsing of multi-sentence as an instance of conjunction.

6.1 Synonymous concepts

The most common error we encounter is synonymous AMR concepts, as shown in Figure 5. Comparing the expected graph (top) to the parsed version (bottom), we note that concept *previous* is synonymized to *past*. While this error is mainly caused by the fact that the multilingual word embeddings bridge non-English input tokens to English concepts, it also highlights the highly lexical nature of Smatch scoring (Cai and Knight, 2013) which does not take synonymous concepts into consideration. Given that AMR is supposed to represent the core meaning of a sentence regardless of its syntactic and morphological variations, Smatch scoring should be able to capture lexical variations such as synonymous concepts.

[4] We did not run experiments with LDC + SQuAD dataset on AMR1.0 since our primary reason for running experiments on AMR1.0 was to more directly be able to compare our results to (Damonte and Cohen, 2018)

```
In this environment, what's wrong if they
criticize the previous stupefying propaganda
a bit?

(w / wrong-02
      :ARG1 (a2 / amr-unknown)
      :ARG2 (c / criticize-01
            :ARG0 (t / they)
            :ARG1 (p / propaganda
                  :time (p2 / previous)
                  :ARG1-of (s / stupefy-01))
            :degree (b / bit))
      :location (e / environment
            :mod (t2 / this)))

Was ist in dieser Umgebung falsch, wenn sie
die bisherige stupeftende Propaganda ein
bisschen kritisieren?

(w / wrong-02
      :ARG1 (c / criticize-01
            :ARG0 (t2 / they)
            :ARG1 (p2 / propaganda
                  :time (p / past))
                  :degree (b / bit))
      :ARG2 (a / amr-unknown)
      :location (e / environment
            :mod (t / this)))
```

Figure 5: The gold AMR (top) and the parsed AMR (bottom) for a German sentence exemplifying errors: synonymous concept (*previous* vs. *past*), missing concept (concept *stupefy-01* is missing in the parsed AMR), incorrect roles (the two arguments, *:ARG1* and *:ARG2*, of *wrong-02* are swapped in the parsed AMR).

```
In critical moments, we are all descendants of
Yan emperor and Huang emperor.

(d / descend-01
      :ARG0 (w / we
            :mod (a / all))
      :source (a2 / and
            :op1 (p / person
                 :name (n / name
                       :op1 "Yan")
                 :ARG0-of (h / have-org-role-91
                          :ARG2 (e / emperor)))
            :op2 (p2 / person
                 :name (n2 / name :op1 "Huang")
                 :ARG0-of h))
      :time (m / moment
            :ARG1-of (c / critical-02)))

In kritischen Momenten sind wir alle Nachfahren
des Yan Kaisers und Huang Kaisers.

(d / descend-01
      :ARG0 (w / we
            :mod (a / all))
      :ARG1 (a2 / and
            :op1 (p / person
                  :name (n / name
                        :op1 "Yan"
                        :op2 "Kaisers"))
            :op2 (p2 / person
                  :name (n2 / name
                        :op1 "Huang"
                        :op2 "Kaisers")))
      :time (m / moment
            :ARG1-of (c / critical-02)))
```

Figure 6: The gold AMR (top) and the parsed AMR (bottom) for a German sentence illustrating incorrect roles (*:source* is replaced by *:ARG1* in the parsed AMR) and incorrect identification of the target token *Kaisers* as a named entity.

6.2 Missing concepts and incorrect roles

Some concepts are missing in the parsed AMR, such as *stupefy-01* in Figure 5. The parser also incorrectly identifies relations between concepts. In Figure 5, arguments *ARG1* and *ARG2* for concept *wrong-02* are swapped. In Figure 6, the relation *:source* is replaced by frame argument *ARG1*.

6.3 Incorrect parsing of Multi-sentence

Another frequent error includes incorrect parsing of multi-sentence as an instance of conjunction, especially when sentences are demarcated by commas. Note that the multi-sentence errors are not specific to multilingual parsing and occur frequently when parsing English input sentences as well. This multi-sentence error is mostly caused by the ambiguity of commas, which can subsume various semantics depending on the contexts across languages.

6.4 Misrecognition of foreign token as a named entity

Some target tokens may legitimately be realized in the gold AMR, especially when the target tokens are named entities, e.g. *Frankfurt, Anna, Noah, etc*. This often leads to errors in the parsed AMR when a target token is incorrectly recognized as a named entity. In Figure 6, German token *Kaisers* is incorrectly parsed as part of named entities *Yan Kaisers* and *Huang Kaisers*. The failure to capture the correct concept *emperor* for the German token *Kaisers* leads to a subsequent error of not reifying the role to *have-org-role-91*[5], evident in the comparison of the parsed AMR with the gold AMRs.

6.5 Others

Other errors include lack of stemming in the target language, such as *Kaisers* in Figure 6. Stemming errors are mostly caused by the fact that we have not incorporated target language stemmers whereas we have incorporated spacy[6] for English. Some errors are caused by machine translation. English fragmentary input *taking a look* is translated to *Sehen Sie sich*, which is then incorrectly parsed as *imperative* sentence. Nominal target language tokens often fail to invoke predicates. Given the input in English "cultural tyranny in the cloak of nationalism", *tyranny* invokes the predicate *tyrannize-01*. Its German counterpart *Tyrannei*, however, fails to

[5]Refer to https://www.isi.edu/ ulf/amr/lib/roles.html and https://www.isi.edu/ ulf/amr/lib/amr-dict.html/have-org-role-91 for details.

[6]https://spacy.io/

	Contextual Alignment	Fast Align
German	23.47	20.52
Italian	29.40	29.30
Spanish	28.81	26.69

Table 5: AMR1.0 parser performance on negations in terms of Smatch. Fast align is compared with the proposed contextual alignment for different languages.

invoke the predicate in "kulturellen Tyrannei im Mantel des Nationalismus".

7 Word alignment error analysis

We compared the annotation projection for AMR1.0 between fast align and the contextual alignment. As noted in Table 3 they perform comparably for German, Italian and Spanish. However, on detailed analysis we notice that annotation projection using contextualized alignments has a greater coverage in terms of foreign text-to-AMR alignments compared to fast align (eg. for German, contextual alignment A.P. gives 99.95% coverage in comparison to 97.47%.). This is likely due to the fact that fast align is based on an IBM alignment model, which relies on expected counts of alignment pairs and uses additional alignment constraints. Contextualized alignment relies on the unrestricted pairing by cosine distance of the XLM-R contextual word embeddings of the input tokens. Given an English token, the contextualized alignment necessarily aligns it to a foreign language word. Furthermore, since embeddings are contextual and pre-trained with large amounts of data, they are robust to non frequent alignment pairs.

The difference between contextualized alignment and fast align for their coverage is most noticeable for compounds. A German counterpart of English non – tariff is *nichttarifäre*. While contextualized alignment aligns *nichttarifäre* to non, which is subsequently aligned to the concept "–" for polarity, fast align leaves *nichttarifäre* unaligned. Such difference is evidenced in the parser performance on negations realized in diverse morphologies. Comparing the AMR1.0 parser performance on negations between fast align (Baseline III in Table 3) and the contextualized alignment (A.P in Table 3), we find that contextualized alignment consistently outperforms fast align across the three European target languages, as shown in Table 5.

8 Conclusion and future directions

In this paper we propose to use transformer-based multilingual word embeddings for *annotation projection* of AMR annotations. We show that our proposed procedure achieves competitive results as some of the classical methods for text-to-AMR alignment. We apply combination techniques to concept alignments and AMR parser training, which significantly improve performance over the base models. We also provide a detailed error analysis of the multilingual AMR parsing.

Given pre-trained transformer-based multilingual word embeddings, contextual word alignment proves to be a useful avenue for overcoming differences amongst languages and addressing the multilingual AMR problem with weak supervision. Moreover, our annotation projection procedure not only achieves a highly competitive performance for German, Spanish, Italian and Chinese but also permits zero-shot learning to other languages included in the training set of the underlying XLM-R multilingual transformer.

Future work may include diversifying input texts using AMR2text (Mager et al., 2020) generation which can address the difference in results between machine translated and human translated test data. The potential of the AMR parser to overcome translation divergence also points to its utility in an end-to-end multilingual translation system, bypassing the need for supervised parallel corpora for machine translation system training.

Acknowledgement

We thank the anonymous reviewers for helpful suggestions. We also thank Revanth Reddy and Jason Furmanek for their varied inputs.

References

Roee Aharoni, Melvin Johnson, and Orhan Firat. 2019. Massively multilingual neural machine translation. 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 3874–3884, Minneapolis, Minnesota. Association for Computational Linguistics.

Rafael Anchiêta and Thiago Pardo. 2018. Towards AMR-BR: A SemBank for Brazilian Portuguese language. In *Proceedings of the Eleventh International Conference on Language Resources and Evaluation (LREC 2018)*, Miyazaki, Japan. European Language Resources Association (ELRA).

Mikel Artetxe, Gorka Labaka, and Eneko Agirre. 2017. Learning bilingual word embeddings with (almost) no bilingual data. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 451–462, Vancouver, Canada. Association for Computational Linguistics.

Mikel Artetxe, Gorka Labaka, and Eneko Agirre. 2018. A robust self-learning method for fully unsupervised cross-lingual mappings of word embeddings. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 789–798, Melbourne, Australia. Association for Computational Linguistics.

Miguel Ballesteros and Yaser Al-Onaizan. 2017. AMR parsing using stack-LSTMs. In *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing*, pages 1269–1275, Copenhagen, Denmark. 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.

Rexhina Blloshmi, Rocco Tripodi, and Roberto Navigli. 2020. XL-AMR: Enabling cross-lingual AMR parsing with transfer learning techniques. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 2487–2500, Online. Association for Computational Linguistics.

Peter Brown, John Cocke, Stephen Della Pietra, Vicent Della Pietra, Fredrick Jelinek, John Lafferty, Robert Mercer, and Paul Roossin. 1990. A statistical approach to machine translation. *Computational Linguistics, Volume 16, Number 2, June 1990*.

Peter Brown, Stephen Della Pietra, Vicent Della Pietra, and Robert Mercer. 1993. The mathematics of statistical machine translation: Parameter estimation. *Computational Linguistics, Volume 19, Number 2*.

Shu Cai and Kevin 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. Association for Computational Linguistics.

Steven Cao, Nikita Kitaev, and Dan Klein. 2020. Multilingual alignment of contextual word representations. In *Proceedings of the 8th International conference on learning representations*.

Wei-Te Chen and Martha Palmer. 2017. Unsupervised AMR-dependency parse alignment. In *Proceedings of the 15th Conference of the European Chapter of the Association for Computational Linguistics: Volume 1, Long Papers*, pages 558–567, Valencia, Spain. Association for Computational Linguistics.

Alexis Conneau, Kartikay Khandelwal, Naman Goyal, Vishrav Chaudhary, Guillaume Wenzek, Francisco Guzmán, Edouard Grave, Myle Ott, Luke Zettlemoyer, and Veselin Stoyanov. 2019. Unsupervised cross-lingual representation learning at scale. *arXiv preprint arXiv:1911.02116*.

Alexis Conneau, Guillaume Lample, Marc'Aurelio Ranzato, Ludovic Denoyer, and Herve Jegou. 2018. Word translation without parallel data. In *Proceedings of the 6th International conference on learning representations*.

Marco Damonte and Shay B. Cohen. 2018. Cross-lingual Abstract Meaning Representation parsing. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long Papers)*, pages 1146–1155, New Orleans, Louisiana. Association for Computational Linguistics.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2018. Bert: Pre-training of deep bidirectional transformers for language understanding. *arXiv preprint arXiv:1810.04805*.

Chris Dyer, Victor Chahuneau, and Noah A. Smith. 2013. A simple, fast, and effective reparameterization of IBM model 2. In *Proceedings of the 2013 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 644–648, Atlanta, Georgia. Association for Computational Linguistics.

Ramon Fernandez Astudillo, Miguel Ballesteros, Tahira Naseem, Austin Blodgett, and Radu Florian. 2020. Transition-based parsing with stack-transformers. In *Findings of the EMNLP2020 (to appear)*.

Jeffrey Flanigan, Sam Thomson, Jaime Carbonell, Chris Dyer, and Noah A. Smith. 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: Long Papers)*, pages 1426–1436, Baltimore, Maryland. Association for Computational Linguistics.

Jan Hajič, Ondřej Bojar, and Zdeňka Urešová. 2014. Comparing Czech and English AMRs. In *Proceedings of Workshop on Lexical and Grammatical Resources for Language Processing*, pages 55–64, Dublin, Ireland. Association for Computational Linguistics and Dublin City University.

Paul R Kingsbury and Martha Palmer. 2002. From treebank to propbank. In *LREC*, pages 1989–1993. Citeseer.

Ioannis Konstas, Srinivasan Iyer, Mark Yatskar, Yejin Choi, and Luke Zettlemoyer. 2017. Neural AMR: Sequence-to-sequence models for parsing and generation. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 146–157, Vancouver, Canada. Association for Computational Linguistics.

Young-Suk Lee, Ramon Fernandez Astudillo, Tahira Naseem, Revanth Gangi Reddy, Radu Florian, and Salim Roukos. 2020. Pushing the limits of amr parsing with self-learning. In *Findings of the EMNLP2020 (to appear)*.

Bin Li, Yuan Wen, Weiguang Qu, Lijun Bu, and Nianwen Xue. 2016. Annotating the little prince with Chinese AMRs. In *Proceedings of the 10th Linguistic Annotation Workshop held in conjunction with ACL 2016 (LAW-X 2016)*, pages 7–15, Berlin, Germany. Association for Computational Linguistics.

Chunchuan Lyu and Ivan Titov. 2018. AMR parsing as graph prediction with latent alignment. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 397–407, Melbourne, Australia. Association for Computational Linguistics.

Manuel Mager, Ramón Fernandez Astudillo, Tahira Naseem, Md Arafat Sultan, Young-Suk Lee, Radu Florian, and Salim Roukos. 2020. Gpt-too: A language-model-first approach for amr-to-text generation. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, Seattle, USA. Association for Computational Linguistics.

Noelia Migueles-Abraira, Rodrigo Agerri, and Arantza Diaz de Ilarraza. 2018. Annotating Abstract Meaning Representations for Spanish. In *Proceedings of the Eleventh International Conference on Language Resources and Evaluation (LREC 2018)*, Miyazaki, Japan. European Language Resources Association (ELRA).

Tomas Mikolov, Quoc V Le, and Ilya Sutskever. 2013. Exploiting similarities among languages for machine translation. *arXiv preprint arXiv:1309.4168*.

Tahira Naseem, Abhishek Shah, Hui Wan, Radu Florian, Salim Roukos, and Miguel Ballesteros. 2019. Rewarding Smatch: Transition-based AMR parsing with reinforcement learning. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 4586–4592, Florence, Italy. Association for Computational Linguistics.

Rik van Noord and Johan Bos. 2017. Neural semantic parsing by character-based translation: Experiments with abstract meaning representations. *arXiv preprint arXiv:1705.09980*.

Franz Josef Och and Hermann Ney. 2003. A systematic comparison of various statistical alignment models. *Computational linguistics*, 29(1):19–51.

Telmo Pires, Eva Schlinger, and Dan Garrette. 2019. How multilingua is multilingual bert? *arXiv preprint arXiv:1906.01502*.

Nima Pourdamghani, Yang Gao, Ulf Hermjakob, and Kevin Knight. 2014. Aligning English strings with Abstract Meaning Representation graphs. In *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 425–429, Doha, Qatar. Association for Computational Linguistics.

Sebastian Ruder, Iva Vulić, and Søgaar Andersd. 2019. A survey of cross-lingual word embedding models. *J. Artif. Int. Res.*, page 569–630.

Tal Schuster, Ori Ram, Regina Barzilay, and Amir Globerson. 2019. Cross-lingual alignment of contextual word embeddings, with applications to zero-shot dependency parsing. 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 1599–1613, Minneapolis, Minnesota. Association for Computational Linguistics.

Samuel Smith, David Turban, Steven Hamblin, and Nils Hamerla. 2017. Offline bilingual word vectors, orthogonal transformations and the inverted softmax. In *Proceedings of the 5th International conference on learning representations*.

Marco Antonio Sobrevilla Cabezudo and Thiago Pardo. 2019. Towards a general Abstract Meaning Representation corpus for Brazilian Portuguese. In *Proceedings of the 13th Linguistic Annotation Workshop*, pages 236–244, Florence, Italy. Association for Computational Linguistics.

Lucy Vanderwende, Arul Menezes, and Chris Quirk. 2015. An AMR parser for English, French, German, Spanish and Japanese and a new AMR-annotated corpus. In *Proceedings of the 2015 Conference of the North American Chapter of the Association for Computational Linguistics: Demonstrations*, pages 26–30, Denver, Colorado. Association for Computational Linguistics.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan Gomez, Lukasz Kaiser, and Illia Polosukhin. 2017. Attention is all you need. *arXiv preprint arXiv:1706.03762*.

Nianwen Xue, Ondřej Bojar, Jan Hajič, Martha Palmer, Zdeňka Urešová, and Xiuhong Zhang. 2014. Not an interlingua, but close: Comparison of English AMRs to Chinese and Czech. In *Proceedings of the Ninth International Conference on Language Resources and Evaluation (LREC'14)*, pages 1765–1772, Reykjavik, Iceland. European Language Resources Association (ELRA).