Multilingual AMR-to-Text Generation

Angela Fan, Claire Gardent

▶ **To cite this version:**

Angela Fan, Claire Gardent. Multilingual AMR-to-Text Generation. 2020 Conference on Empirical Methods in Natural Language Processing, Nov 2020, Punta Cana, Dominican Republic. hal-02999676

HAL Id: hal-02999676
https://hal.science/hal-02999676

Submitted on 11 Nov 2020

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

Multilingual AMR-to-Text Generation

Angela Fan
FAIR/LORIA
Université de Lorraine, Nancy, France
angelafan@fb.com

Claire Gardent
CNRS/LORIA
Nancy, France
cgardent@loria.fr

Abstract

Generating text from structured data is challenging because it requires bridging the gap between (i) structure and natural language (NL) and (ii) semantically underspecified input and fully specified NL output. Multilingual generation brings in an additional challenge: that of generating into languages with varied word order and morphological properties. In this work, we focus on Abstract Meaning Representations (AMRs) as structured input, where previous research has overwhelmingly focused on generating only into English. We leverage advances in cross-lingual embeddings, pretraining, and multilingual models to create multilingual AMR-to-text models that generate in twenty one different languages. For eighteen languages, based on automatic metrics, our multilingual models surpass baselines that generate into a single language. We analyse the ability of our multilingual models to accurately capture morphology and word order using human evaluation, and find that native speakers judge our generations to be fluent.

1 Introduction

Generating text from structured data has a variety of applications in natural language processing. Tasks such as decoding from tables (Lebret et al., 2016; Sha et al., 2018), question answering from knowledge bases (Fan et al., 2019a), and generation from RDF Triples (Gardent et al., 2017), knowledge graphs (Marcheggiani and Perez-Beltrachini, 2018) and linguistic meaning representations (Konstas et al., 2017) face similar challenges: interpreting structured input and writing fluent output. We focus on generating from graph structures in the form of Abstract Meaning Representations (AMR) (Banarescu et al., 2013). Previous work has largely focused on generating from AMR into English, but we propose a multilingual approach that can decode into twenty one different languages.

Compared to multilingual translation, decoding from structured input has distinct challenges. Translation models take natural language input and must faithfully decode into natural language output. However, as shown in Zhao et al. (2020), bridging the gap between structured input and linear output is a difficult task. In addition, in structured input such as graphs, the input is usually semantically under-specified. For example, in AMRs, function words are missing and tense and number are not given. Thus, generation from structured input must bridge the gap between (i) structure and string and (ii) underspecified input and fully specified output. Multilinguality brings a third challenge — that of generating in languages that have varied morphological and word order properties.

Annotating natural language with AMR is a complex task and training datasets only exist for English[1], so previous work on AMR-to-text generation has overwhelmingly focused on English. We create training data for multilingual AMR-to-Text models, by taking the EUROPARL multilingual corpus and automatically annotating the English data with AMRs using the `jamr` semantic parser. We then use the English AMRs as the input for all generation tasks. To improve quality, we leverage recent advances in natural language processing such as cross-lingual embeddings, pretraining and multilingual learning. Cross-lingual embeddings have shown striking improvements on a range of cross-lingual natural language understanding tasks (Devlin et al., 2019; Conneau et al., 2019; Wu and Dredze, 2019; Pires et al., 2019). Other work has shown that the pre-training and fine-tuning approaches also help improve generation performance (Dong et al., 2019; Song et al., 2019; Lawrence et al., 2019; Rothe et al., 2019). Finally, multilingual models, where a single model

[1] AMR datasets from the LDC can be found at https://amr.isi.edu/download.html

is trained to translate from multiple source languages into multiple target languages, are achieving increasingly better results in machine translation (Johnson et al., 2017; Firat et al., 2017; Aharoni et al., 2019; Arivazhagan et al., 2019).

By combining these techniques, we demonstrate that fluent generation is possible for multilingual AMR-to-Text models. We use automatic and human evaluation to assess performance on (1) EUROPARL data with silver English-centric AMR as input and the 21 EUROPARL languages as target and (2) on LDC2015E86 data with gold English-centric AMR as input and English, Spanish, Italian, and German as target. Our results demonstrate, for the first time, that it is possible to generate from AMRs into multiple languages. We show that multilingual models have strong performance compared to single-language baselines and produce fluent output, based on the judgments of native speakers. We further investigate how factors, such as differences in the size of the training data, differences in language word order and morphological properties, and differences in the set of languages used for training many-to-one models, impact results. We will make code and models available, to aid research in multilingual AMR-to-Text Natural Language Generation.

2 Related Work

AMR-to-Text Generation. Initial work on AMR-to-text generation adapted methods from statistical machine translation (MT) (Pourdamghani et al., 2016), grammar-based generation (Mille et al., 2017), tree-to-string transducers (Flanigan et al., 2016), and inverted semantic parsing (Lampouras and Vlachos, 2017). Neural approaches explored sequence-to-sequence models where the AMR is linearized (Konstas et al., 2017) or modeled with a graph encoder (Marcheggiani and Perez-Beltrachini, 2018; Damonte and Cohen, 2019; Ribeiro et al., 2019; Song et al., 2018; Zhu et al., 2019). As professionally-annotated AMR datasets are in English, all this work focuses on English.

One exception is the work of Sobrevilla Cabezudo et al. (2019) which uses automatic translation to translate the English text of the LDC AMR data into Brazilian Portuguese and align English with the Portuguese translation to create Portuguese-centric AMRs. However, this work focuses only on one language. In contrast, we consider generation into twenty one languages. We use very different methods and generate from English-centric AMRs, not target-language AMRs.

Multilingual MR-to-Text Generation. While work on AMR-to-Text generation has mostly focused on generation into English, the Multilingual Surface Realization shared tasks (Mille et al., 2018, 2019) have made parallel MR/Text datasets available for 11 languages. Two tracks are proposed: a shallow track where the input is an unordered, lemmatized dependency tree and a deep track where the dependency tree edges are labelled with semantic rather than syntactic relations and where function words have been removed.

The participants approaches to this multilingual generation task use gold training data and mostly focus on the shallow track where the input is an unordered lemmatized dependency tree and the generation task reduces to linearization and morphological realization. The models proposed are pipelines that model each of these subtasks and separate models are trained for each target language (Kovács et al., 2019; Yu et al., 2019; Shimorina and Gardent, 2019a,b; Castro Ferreira and Krahmer, 2019). In this work, we focus instead on more abstract, deeper, input (AMRs) and propose end-to-end, multilingual models for all target languages.

3 Method

To generate from AMRs, we use neural sequence to sequence models that model the input AMR with a Transformer Encoder and generate natural language with a Transformer Decoder. For all languages, the input is an English-centric AMR that was derived automatically using the `jamr` semantic parser from English text. We pre-train both the AMR encoder and the multilingual decoder and we leverage crosslingual embeddings.

3.1 Encoding English AMR

Abstract Meaning Representations are semantic representations that take the form of a rooted, directed acyclic graph. AMR abstracts away syntax such that sentences with similar meanings have similar AMR graphs. Full detail is not kept by the AMR — for example, elements such as verb tense are lost. While we focus on decoding from AMR input, the structured form is reflective of other structured inputs used in tasks such as generating from semantic role labels (Fan et al., 2019c) or RDF triples (Gardent et al., 2017).

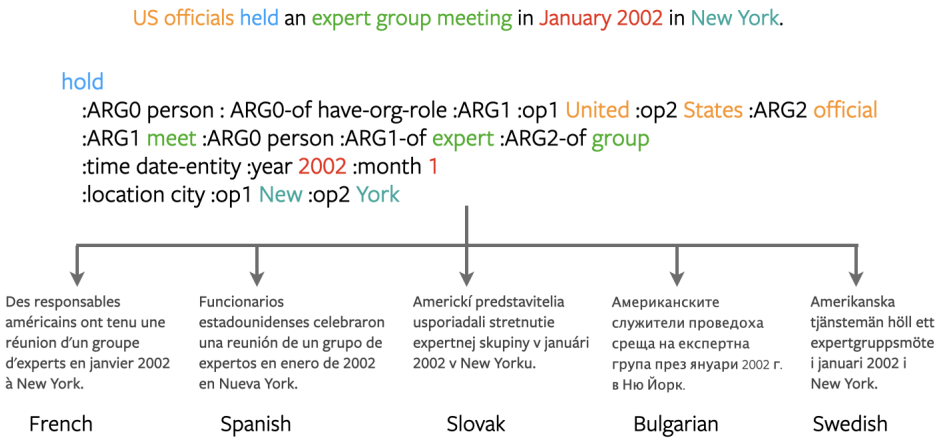

Figure 1: **Generating into Multiple Languages from English AMR.**

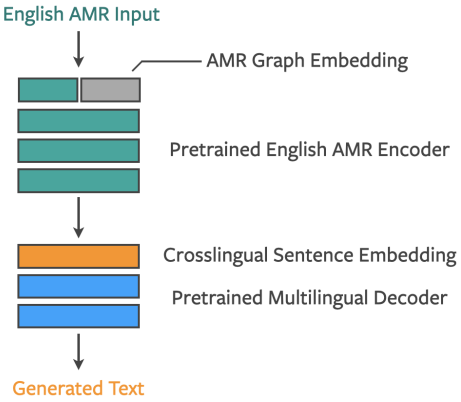

Figure 2: **One-to-Many Architecture for Multilingual AMR-to-Text Generation.** The English-centric AMR input is linearized and modeled with graph embeddings with a pre-trained Transformer Encoder. Text is generated with a pre-trained Transformer Decoder initialized with cross-lingual embeddings.

The AMR graph is first linearized into a sequence of tokens as shown in Figure 1 after preprocessing following (Konstas et al., 2017) (see Section 4.1 for a detailed description). Rather than model the graph structure directly, following Fan et al. (2019a), we model the graph using a *graph embedding*. The graph embedding provides additional information to the Transformer Encoder by encoding the depth of each node in the rooted graph and the subgraph each node belongs to. Concretely, each token has a word and position embedding, and additionally an indicator of depth calculated from the root and an indicator of which subtree the node belongs to (with all subtrees stemming from the root). These additional embeddings are concatenated to the word and position embeddings. Such information allows the Transformer Encoder to capture some graph structure information, while still modeling a sequence. This is depicted in Figure 2.

To create a one-to-many multilingual model, we model a *language embedding* on the encoder side to allow the decoder to distinguish which language to generate into. This technique has been previously used in multilingual translation (Arivazhagan et al., 2019). The English AMR begins with a token that indicates the decoder side language.

To improve the quality of the encoder, we incorporate large-scale pretraining on millions of sequences of AMR by adopting the generative pretraining approach proposed in Lewis et al. (2019a). This pretraining incorporates various noise operations, such as masking (Devlin et al., 2019), span masking (Fan et al., 2019a), and shuffling. Previous work has shown that pretraining is effective for providing neural models with additional information about the structure of natural language and improving model quality (Dong et al., 2019; Song et al., 2019; Lawrence et al., 2019). As models increase in size, smaller training datasets (such as human-annotated AMR) are often not large enough to fully train these models. The entire encoder is pretrained on silver AMRs, as shown in Figure 2.

3.2 Multilingual Decoding from AMR

The Transformer Decoder attends to the encoded English AMR, a graph of concepts and relations, and generates text into many different languages with varied word order and morphology.

As displayed in Figure 2, we use both language model pretraining and crosslingual embeddings to improve decoder quality. Monolingual data from various languages is used to pretrain each language model. Further, we incorporate crosslingual em-

beddings. These embeddings aim to learn universal representations that encode sentences into shared embedding spaces. Various recent work in crosslingual embeddings (Conneau and Lample, 2019) show strong performance on other multilingual tasks, such as XNLI (Conneau et al., 2018), XTREME (Hu et al., 2020), and MLQA (Lewis et al., 2019b). We use the embeddings from XLM (Conneau and Lample, 2019) to initialize the multilingual embeddings of our decoder.

3.3 Model Training

To train our one-to-many multilingual AMR-to-text generation model, we use pairs of English AMR and text in multiple different languages. The English AMR does not need to be aligned to sentences in multiple languages. Instead, we create one AMR-to-text corpus for each language and concatenate all of them for training a multilingual model. During the training process, the pretrained AMR encoder and pretrained crosslingual decoder are finetuned on our multilingual AMR-to-text training corpus.

4 Experimental Setting

We describe the various sources of data used to create multilingual AMR-to-text generation models and describe the implementation and evaluation.

4.1 Data

Pretraining For encoder pretraining on silver AMR, we take thirty million sentences from the English portion of CCNET[2] (Wenzek et al., 2019), a cleaned version of Common Crawl (an open source version of the web). We use `jamr`[3] to parse English sentences into AMR. For multilingual decoder pretraining, we take thirty million sentences from each language split of CCNET.

Multilingual Data We use EUROPARL, an aligned corpus of European Union parliamentary debates. Each language in EUROPARL is aligned to English. We study the twenty one languages available in EUROPARL: Bulgarian, Czech, Danish, Dutch, English, German, Greek, Spanish, Estonian, Finnish, French, Hungarian, Italian, Lithuanian, Latvian, Polish, Portuguese, Romanian, Slovak, Slovenian, and Swedish. The earliest releases in EUROPARL were prepared with a fixed common testing set across all languages, but later releases in ten new languages do not have a validation or test set. Thus, for the languages where the standard split is applicable, we report results on the common testing set, splitting it in half for validation and testing. For languages where there is no evaluation set, we take a part of the training set and reserve it for validation and another portion for testing. We use `jamr` to parse the English text of the Europarl corpus into AMRs. This creates a corpus of automatically created silver English AMRs aligned with sentences in twenty one European languages.

Gold AMR We also evaluate our models (trained on silver AMRs) on gold AMR where available. For this, we use the CROSSLINGUAL AMR dataset from Damonte and Cohen (2018)[4]. The corpus was constructed by having professional translators translate the English text of the LDC2015E86 test set into Spanish, Italian, German, and Chinese. We only evaluate on languages where we have training data from EUROPARL (i.e. we do not include Chinese as it is not in EUROPARL).

Preprocessing All data remains untokenized and cased. For AMR, we follow Konstas et al. (2017) in processing the `jamr` output into a simpler form. We remove variable names and instance-of relation (/) before every concept. However, we do not anonymize entities or dates, as improvements in modeling have allowed for better representations of rare words such as entities. We learn a sentencepiece model with 32K operations to split the English AMR into subword units. On the decoder side, we apply the sentencepiece model and vocabulary of XLM (Conneau and Lample, 2019). We choose to use the existing XLM sentencepiece and vocabulary so that the XLM cross-lingual embeddings can be used to initialize our models. For the encoder, we do not use existing vocabularies, as they do not capture the AMR graph structure.

4.2 Models

We implement our models in `fairseq-py` (Ott et al., 2019). We use large Transformer (Vaswani et al., 2017) sequence-to-sequence models and train all models for 50 epochs with LayerDrop (Fan et al., 2019b), which takes around 2 days. We initialize all weights with the pretrained models. When combining crosslingual word embeddings and encoder and decoder pretraining, we initialize all weights

[2] https://github.com/facebookresearch/cc_net

[3] https://github.com/jflanigan/jamr

[4] To evaluate on this data, please contact Damonte and Cohen (2018)

Model Amount of Data	**en** 8.2M	**da** 1.9M	**de** 1.9M	**el** 1.2M	**es** 1.9M	**fi** 1.9M	**fr** 2M	**it** 1.9M	**nl** 1.9M	**pt** 1.9M	**sv** 1.9M
Machine Translation	—	—	**17.8**	—	24.9	—	**20.7**	18.6	19.4	21.0	19.2
English AMR-XX	**34.2**	21.3	16.9	14.2	24.3	12.9	20.5	**19.1**	18.8	20.4	18.6
Multilingual	32.5	21.2	17.0	13.8	24.2	12.4	19.7	17.8	18.5	20.5	18.7
+ Graph Embedding	32.9	21.4	17.0	14.0	24.3	12.5	19.9	18.0	18.6	20.7	18.9
+ Crosslingual Embedding	33.0	21.7	17.3	14.4	24.7	12.9	19.9	18.5	19.0	21.0	19.0
+ Encoder Pretraining	33.4	21.7	17.3	14.5	24.9	13.0	20.2	18.7	19.1	21.0	19.1
+ Decoder Pretraining	33.8	**21.9**	17.5	**14.6**	**25.1**	**13.4**	20.3	18.9	**19.4**	**21.2**	**19.5**

Model Amount Data	**bg** 400K	**cs** 650K	**et** 650K	**hu** 620K	**lt** 630K	**lv** 640K	**pl** 630K	**ro** 400K	**sl** 640K	**sk** 620K
English AMR-XX	33.8	27.5	18.9	23.1	23.9	25.4	23.4	30.6	30.1	28.7
Multilingual	34.6	28.4	19.1	23.8	24.4	26.9	23.4	31.5	30.6	29.7
+ Graph Embedding	34.7	28.5	19.3	23.9	24.5	27.0	23.6	31.5	20.7	29.9
+ Crosslingual Embedding	35.0	28.9	19.7	24.3	24.8	27.4	24.0	31.7	30.8	30.1
+ Encoder Pretraining	35.2	29.0	19.8	24.5	25.0	27.5	24.1	31.9	31.0	30.2
+ Decoder Pretraining	**35.7**	**29.5**	**21.2**	**24.7**	**25.5**	**27.9**	**24.4**	**32.1**	**31.4**	**30.6**

Table 1: **Results on 21 Languages in Europarl**. The English-XX baseline (generation into a single language) combines all modeling improvements. When training on multiple seeds, the standard deviation is around 0.1 to 0.3 BLEU, making the difference between the multilingual baseline and the addition of our modeling improvements statistically significant.

with pretraining, then use crosslingual word embeddings. We do not perform extensive hyperparameter search, but experimented with various learning rate values to maintain stable training with pretrained initialization. To generate, we decode with beam search with beam size 5. Our pretrained models are available for download.[5]

4.3 Monolingual and Translation Baselines

We compare our multilingual models both to monolingual models (one model trained for each language) and to a hybrid NLG/MT baseline. For the latter, we first generate with the AMR-to-English model and then translate the generation output to the target language using MT. Our translation models are Transformer Big models trained with LayerDrop (Fan et al., 2019b) for 100k updates on public benchmark data from WMT where available and supplemented with mined data from the `ccMatrix` project (Schwenk et al., 2019). We trained translation models for languages where large quantities of aligned bitext data are readily available, and cover a variety of languages.

4.4 Evaluation

We evaluate with detokenized BLEU using `sacrebleu` (Post, 2018). We conduct human evaluation by asking native speakers to evaluate word order, morphology, semantic faithfulness (with respect to the reference) and paraphrasing (how much the generation differs from the reference) on a 3 point scale. The evaluation was done online. For each language, evaluators annotated 25 test set sentences with high BLEU score and 25 sentences with low BLEU score. We removed sentences that were shorter than 5 words. As it is difficult to ensure high quality annotations for 21 languages using crowdsourcing, we relied on colleagues by reaching out on NLP and Linguistics mailing lists. As a result, the number of evaluators per language varies (cf. Table 3).

[5] https://github.com/facebookresearch/m-amr2text

Model	**en**	**es**	**it**	**de**
Konstas et al. (2017)	22.0	—	—	—
Song et al. (2018)	23.3	—	—	—
Cao et al. (2019)	23.5	—	—	—
Damonte et al. (2019)	24.4	—	—	—
Guo et al. (2019)	25.7	—	—	—
Ribeiro et al. (2019)	24.3	—	—	—
Zhu et al. (2019)	**29.7**	—	—	—
Machine Translation	—	21.6	19.6	**15.7**
English-XX Seq2Seq	25.2	21.1	**19.8**	14.9
Multilingual Seq2Seq	24.2	21.0	19.0	14.7
+ Graph Attribute	24.5	21.0	19.2	14.8
+ Crosslingual Embed	24.6	21.3	19.4	15.1
+ AMR Enc Pretrain	24.7	21.5	19.6	15.1
+ Multiling Dec Pretrain	24.9	**21.7**	**19.8**	15.3
+ Finetune on Gold AMR	26.3	—	—	—

Table 2: **Results on Gold AMR from LDC2015E86**.

Language	Number of Evaluators	Morphology	Word Order	Semantic Accuracy	Good Paraphrases	Std Dev Morphology	Std Dev Word Order
English	7	2.9	2.9	2.4	84%	0.06	0.04
Danish	2	2.9	2.9	2.3	88%	0.09	0.04
German	4	3.0	2.9	2.2	75%	0.02	0.06
Greek	5	2.9	2.9	2.2	75%	0.06	0.04
Spanish	10	2.9	2.9	2.2	81%	0.09	0.07
Finnish	2	2.9	3.0	2.1	69%	0.01	0.00
French	7	3.0	3.0	2.3	81%	0.02	0.03
Italian	5	3.0	3.0	2.3	82%	0.04	0.05
Dutch	7	2.9	2.9	1.9	60%	0.06	0.06
Portuguese	7	2.9	2.9	2.4	83%	0.08	0.06
Swedish	5	2.9	2.9	2.3	84%	0.04	0.08
Bulgarian	6	2.8	2.8	2.0	67%	0.07	0.11
Czech	3	2.9	2.8	2.3	79%	0.05	0.11
Estonian	1	2.9	2.9	2.2	78%	—	—
Hungarian	5	2.6	2.5	2.1	70%	0.14	0.23
Latvian	3	2.8	2.7	2.1	74%	0.07	0.16
Polish	2	2.8	2.9	1.6	54%	0.10	0.04
Romanian	10	2.7	2.7	1.9	68%	0.22	0.23

Table 3: **Human Evaluation**. Native speakers assess fifty sentences on a scale of 1 to 3, with 3 the highest score. Good Paraphrases are sentences with high scores (2 or 3) for both Semantic Accuracy and Paraphrasing.

We evaluate multilingual AMR-to-Text generation models in 21 languages. We conduct an ablation study which demonstrates the improvements in modeling performance induced by incorporating graph embeddings, cross lingual embeddings, and pretraining. Finally, we analyze model performance with respect to several linguistic attributes (word order, morphology, paraphrasing, semantic faithfulness) using both automatic metrics and human evaluation.

4.5 Multilingual AMR-to-Text Generation

Monolingual vs. Multilingual Models. We compare English-XX baselines trained to generate from AMR into a single language with multilingual models. We note that as the English-XX models specializes for each language, they have less to model with the same parameter capacity. Results are shown in Table 1. Overall, multilingual models perform well — on 18 of the 21 languages, the performance measured by BLEU is stronger than the monolingual baseline.

One advantage of multilingual AMR-to-Text generation is increased quantities of AMR on the encoder side. This is particularly helpful when the size of the training data is low. For instance, Estonian (*et*) sees a 2.3 BLEU point improvement from multilingual modeling. Conversely, languages such as English, Swedish and French benefit less from multilingual modeling, most likely because there is sufficient data for those languages already. More generally, there is a marked difference between languages for which the training data is large and those for which the training data is smaller. When the training data is large (1.9 to 2M training instances, top part of Table 1), the average improvment is +0.36 BLEU (Min:-0.2, Max:+0.9) whereas for languages with smaller training data (400 to 620K training instances, bottom part of Table 1[6]), the average improvement is +1.75 (Min:+1, Max:+2.3). These trends are similar to observations on other tasks — namely that pretraining is most helpful when there is not sufficient training data in the task itself to train strong representations.

5 Results

Performance on Gold English AMR We evaluate our models trained on silver AMR on the CROSSLINGUAL AMR dataset from Damonte and Cohen (2018) where the input is a gold English-centric AMR and the output is available in three European languages: Spanish, French, and Italian. The results are shown in Table 2. Similar to the trends seen when generating from silver AMR, we find that multilingual models have strong performance. BLEU scores are lower than on EUROPARL as the models are tested out of domain (training on parliamentary debates but testing on newswire and forum data domains).

On English LDC data, we compare to existing work. Even though it is trained on silver AMRs

[6]For many languages, such as Slavic languages, it is because the EU expanded to include these countries later on. Thus there is less European Parliamentary proceeding data.

and out of domain, non-LDC data, the multilingual model compares well with previous work (see Table 2). When finetuned on the LDC2015E86 train set, our model improves on English by over 1 BLEU point, outperforming all previous work except Zhu et al. (2019). This work directly models the graph structure of AMR with structure aware attention to improve Transformer architectures — this is orthogonal to our main aim of multilingual generation and can be incorporated in future work.

Impact of Modeling Improvements. For the multilingual model, we display the effect of incrementally adding additional modeling improvements (cf. Table 1). Each improvement is essentially universally helpful across all considered languages, though some have a greater improvement on performance than others.

Comparison to the Hybrid NLG/MT Baseline. Compared to the NLG/MT baseline, our multilingual models provide comparable results while providing an arguably simpler approach (end-to-end rather than pipeline) and training on much lower quantities of parallel data — on German and French (very high resource languages with millions of examples of training data), there is slightly stronger performance. On other languages we compare to, the translation models perform a bit worse.

We further conduct a human evaluation study on Spanish, Italian, and German. We ask evaluators to assess the morphology, word order, and semantic accuracy of our Multilingual AMR to Text system compared to this hybrid English AMR to Text + Machine Translation baseline. We show in Table 4 that the two models score very similarly in human evaluation, indicating the strength of this fully multilingual system in producing fluent output.

5.1 Analysis of Multilingual Generation

A core challenge for multilinguality is that languages differ with respect to word order and morphology, so models must learn this per language. We use automatic and human evaluation to investigate how these differences affect performance.

Morphology Instead of operating on words, our models use sentencepiece (Wu et al., 2016), a data-driven approach to break words into subwords. As shown in Wu and Dredze (2019), in transfer-based approaches to natural language understanding tasks, the proportion of subwords shared between the source and the transfer language impacts

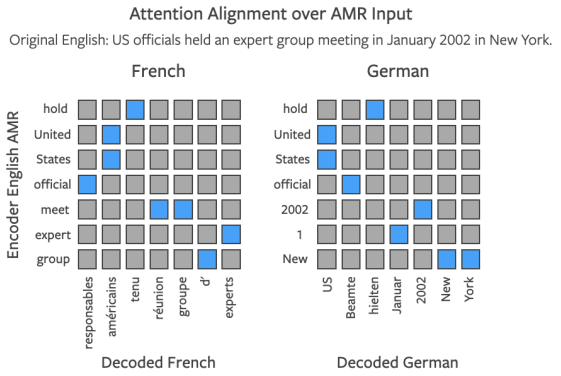

Figure 3: **Attention alignment** when decoding in French and German from the same input AMR.

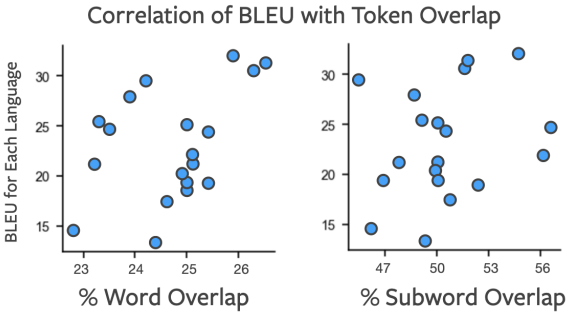

Figure 4: **Relationship between BLEU Score and Token Overlap** for all 21 languages. Correlation coefficient between word overlap and BLEU is 0.42, and coefficient between subword overlap and BLEU is 0.26.

performance. We therefore explore the relation between the proportion of subwords and words shared between the AMR and the output vocabulary. Figure 4 displays this relationship, with weak positive correlation for both word and subword overlap.

We further assess morphology by asking human evaluators to grade the morphology of sentences (*Is the morphology correct? Are agreement constraints e.g., verb/subject, noun/adjective respected?*) on a scale from 1 to 3 with 3 being the highest score. As Table 3 shows, there is not much difference in performance between languages even though there is a marked difference in terms of agreement constraints between e.g., Finnish and English. Between annotators, agreement was high — the standard deviation across was low, with the exception of Romanian, Hungarian, and Spanish (as shown in Table 3). This demonstrates the surprisingly high ability of multilingual models to generalize across languages.

Word Order To assess the impact of varied word orders by language, we ask human evaluators to

Evaluation	Morphology	Word Order	Semantic Accuracy
Spanish			
Machine Translation	2.9	2.7	2.0
Multilingual AMR to Text	2.8	2.9	2.1
Italian			
Machine Translation	3.0	2.9	2.2
Multilingual AMR to Text	2.9	3.0	2.1
German			
Machine Translation	2.8	2.9	2.0
Multilingual AMR to Text	3.0	3.0	2.2

Table 4: **Human Evaluation of our approach compared to the Hybrid English AMR to Text + Machine Translation baseline using Gold AMR from LDC2015E86**. Two native speakers per language assess fifty sentences each on a scale of 1 to 3, with 3 being the highest score.

English	**Generation**	This point will certainly be the subject of subsequent further debates in the council.
	Reference	This is a point that will undoubtedly be discussed later in the Council.
French	**Generation**	Je ne suis pas favorable à des exceptions à cette règle.
	Reference	A mon avis, il n'est pas bon de faire des exceptions à cette règle.
Swedish	**Generation**	Därför röstade vi inte för detta betänkande.
	Reference	Vi har därför inte röstat för detta betänkande.

Table 5: **Example Paraphrases** generated by our multilingual model.

Model	**es**	**fr**	**it**	**pt**	**ro**
One Language	25.2	20.3	18.9	22.2	32.1
Romance Family	25.5	20.5	19.3	22.5	32.5
All Languages	25.3	20.5	19.3	22.4	32.2
		da	**de**	**nl**	**sv**
One Language		21.3	17.0	18.5	18.7
Germanic Family		21.8	21.9	19.6	19.3
All Languages		21.9	17.5	19.4	19.5

Table 6: **Performance when training with increasingly more languages.** Training one multilingual AMR-to-Text model with languages in the related language family improves performance.

judge if the word order is natural. As shown in Table 3, for all languages except Latvian and Romanian, the score is very high (close to 3) indicating that the model learns to decode into multiple languages even though word order differs. The agreement between annotators was high, with low standard deviation (see Table 3). Further, the attention pattern between the encoder English AMR and the decoder clearly reflects the word order of the various languages. This is illustrated in Figure 3, where the activation pattern mirrors the word order difference between French (1) and German (2).

(1) *ont tenu (une réunion de groupe)*$_{\text{OBJ}}$ *(en Janvier 2020)*$_{\text{TIME}}$ *(à New York)*$_{\text{LOC}}$

(2) *hielten (im Januar 2020)*$_{\text{TIME}}$ *(in New York)*$_{\text{LOC}}$ *(eine Gruppestreffen)*$_{\text{OBJ}}$

Training on Related Languages Multilingual models have the potential to benefit from similarities between languages. Languages of the same family often have shared morphological characteristics and vocabulary. First, we analyze the performance of training on languages within a family. Table 6 displays that a model trained on languages within a family has the strongest performance.

Second, we analyze languages within the same family. For four families: Romance, Germanic, Uralic, and Slavic, we create multilingual models trained on pairs. One pair is for the most related languages within that family (e.g. Spanish and Portuguese) and another pair is for the farthest languages within that family (e.g. Spanish and Romanian). We determine which pairs are close and far from Ahmad et al. (2019). Results in Figure 5 display that training on pairs of closely related languages has better performance than pairs of less closely related languages, even within a family. Multilingual models could pick up on similarities between languages to improve performance.

Semantic Accuracy and Paraphrasing. We ask human evaluators to grade the faithfulness of the hypothesis compared to the reference on a scale of 1 to 3. As shown in Table 3, the overall semantic accuracy is very high (note a score of 2 indicates *minor differences*). We also asked annotators to

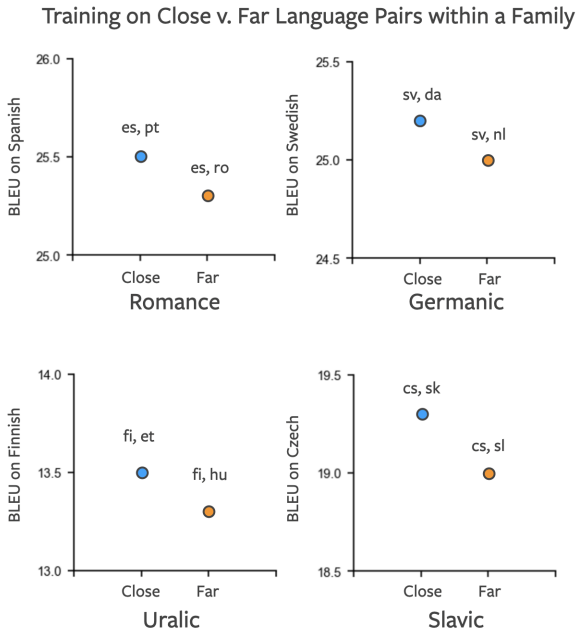

Figure 5: **BLEU difference training on Close v. Far Languages within One Family**. Training on a close pair consistently improves performance compared to training on a far pair, even within a language family.

evaluate how different the generated sentence was from the reference. When coupled with the semantic accuracy score, this allows us to evaluate generation of true paraphrases i.e., sentences with the same meaning as the reference but different surface form. In Table 3, *Good Paraphrases* indicates the percentage of cases that scored highly (2 or 3) with respect to both semantic adequacy and paraphrasing. A large majority of generated sentences are labeled as valid paraphrases by native speakers, indicating (i) that despite underspecified input, the written sentence retains the meaning of the reference and (ii) that this underspecification allows for the generation of paraphrases. This also suggests that BLEU scores only partially reflect model performance as good paraphrases typically differ from the reference and are likely to get lower BLEU score even though they may be semantically accurate. Table 5 shows some examples illustrating the paraphrasing potential of the approach.

6 Conclusion

Abstract Meaning Representations were designed to describe the meaning of English sentences. As such they are heavily biased towards English. AMR concepts are either English words, PropBank framesets ("want-01") or special, English-based keywords (e.g., "date-entity"). The structure of AMRs is also influenced by English syntax. For instance, the main relation of *"I like to eat"* is the concept associated with its main verb ("like") whereas given the corresponding German sentence *"Ich esse gern" (Lit. "I eat willingly")*, the main predicate might have been chosen to be "eat" ("essen"). In other words, AMRs should not necessarily be viewed as an interlingua (Banarescu et al., 2013). Nonetheless, our work suggests that it can be used as one: given an English-centric AMR it is possible to generate the corresponding sentence in multiple languages. This is in line with previous work by (Damonte and Cohen, 2019) which shows that despite translation divergences, AMR parsers can be learned for Italian, Chinese, German and Spanish which all map into an English-centric AMR.

Acknowledgments

We thank the anonymous reviewers for their feedback. We gratefully acknowledge the support of the French National Research Agency (Gardent; award ANR-20-CHIA-0003, XNLG "Multilingual, Multi-Source Text Generation").

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.

Wasi Ahmad, Zhisong Zhang, Xuezhe Ma, Eduard Hovy, Kai-Wei Chang, and Nanyun Peng. 2019. On difficulties of cross-lingual transfer with order differences: A case study on 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 2440–2452.

Naveen Arivazhagan, Ankur Bapna, Orhan Firat, Dmitry Lepikhin, Melvin Johnson, Maxim Krikun, Mia Xu Chen, Yuan Cao, George Foster, Colin Cherry, et al. 2019. Massively multilingual neural machine translation in the wild: Findings and challenges. *arXiv preprint arXiv:1907.05019*.

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.

Thiago Castro Ferreira and Emiel Krahmer. 2019. Surface realization shared task 2019 (MSR19): The team 6 approach. In *Proceedings of the 2nd Workshop on Multilingual Surface Realisation (MSR 2019)*, pages 59–62, Hong Kong, China. 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 and Guillaume Lample. 2019. Cross-lingual language model pretraining. In *Advances in Neural Information Processing Systems*, pages 7057–7067.

Alexis Conneau, Ruty Rinott, Guillaume Lample, Adina Williams, Samuel Bowman, Holger Schwenk, and Veselin Stoyanov. 2018. Xnli: Evaluating cross-lingual sentence representations. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 2475–2485.

Marco Damonte and Shay B Cohen. 2018. Cross-lingual abstract meaning representation parsing. In *Proceedings of NAACL-HLT*, pages 1146–1155.

Marco Damonte and Shay B. Cohen. 2019. Structural neural encoders for AMR-to-text generation. 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 3649–3658, Minneapolis, Minnesota. Association for Computational Linguistics.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. BERT: Pre-training of deep bidirectional transformers for language understanding. 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 4171–4186, Minneapolis, Minnesota. Association for Computational Linguistics.

Li Dong, Nan Yang, Wenhui Wang, Furu Wei, Xiaodong Liu, Yu Wang, Jianfeng Gao, Ming Zhou, and Hsiao-Wuen Hon. 2019. Unified Language Model Pre-training for Natural Language Understanding and Generation. *CoRR*, abs/1905.03197.

Angela Fan, Claire Gardent, Chloé Braud, and Antoine Bordes. 2019a. Using local knowledge graph construction to scale seq2seq models to multi-document inputs. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 4177–4187.

Angela Fan, Edouard Grave, and Armand Joulin. 2019b. Reducing transformer depth on demand with structured dropout. In *International Conference on Learning Representations*.

Angela Fan, Mike Lewis, and Yann Dauphin. 2019c. Strategies for structuring story generation. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 2650–2660.

Orhan Firat, Kyunghyun Cho, Baskaran Sankaran, Fatos T Yarman Vural, and Yoshua Bengio. 2017. Multi-way, multilingual neural machine translation. *Computer Speech & Language*, 45:236–252.

Jeffrey Flanigan, Chris Dyer, Noah A. Smith, and Jaime Carbonell. 2016. Generation from abstract meaning representation using tree transducers. In *Proceedings of the 2016 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 731–739, San Diego, California. Association for Computational Linguistics.

Claire Gardent, Anastasia Shimorina, Shashi Narayan, and Laura Perez-Beltrachini. 2017. The WebNLG challenge: Generating text from RDF data. In *Proceedings of the 10th International Conference on Natural Language Generation*, pages 124–133, Santiago de Compostela, Spain. Association for Computational Linguistics.

Junjie Hu, Sebastian Ruder, Aditya Siddhant, Graham Neubig, Orhan Firat, and Melvin Johnson. 2020. Xtreme: A massively multilingual multi-task benchmark for evaluating cross-lingual generalization. *arXiv preprint arXiv:2003.11080*.

Melvin Johnson, Mike Schuster, Quoc V Le, Maxim Krikun, Yonghui Wu, Zhifeng Chen, Nikhil Thorat, Fernanda Viégas, Martin Wattenberg, Greg Corrado, et al. 2017. Google's multilingual neural machine translation system: Enabling zero-shot translation. *Transactions of the Association for Computational Linguistics*, 5:339–351.

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.

Ádám Kovács, Evelin Ács, Judit Ács, Andras Kornai, and Gábor Recski. 2019. BME-UW at SRST-2019: Surface realization with interpreted regular tree grammars. In *Proceedings of the 2nd Workshop on Multilingual Surface Realisation (MSR 2019)*, pages 35–40, Hong Kong, China. Association for Computational Linguistics.

Gerasimos Lampouras and Andreas Vlachos. 2017. Sheffield at SemEval-2017 task 9: Transition-based

language generation from AMR. In *Proceedings of the 11th International Workshop on Semantic Evaluation (SemEval-2017)*, pages 586–591, Vancouver, Canada. Association for Computational Linguistics.

Carolin Lawrence, Bhushan Kotnis, and Mathias Niepert. 2019. Attending to Future Tokens for Bidirectional Sequence Generation. In *EMNLP-IJCNLP*, pages 1–10, Hong Kong, China.

Rémi Lebret, David Grangier, and Michael Auli. 2016. Neural text generation from structured data with application to the biography domain.

Mike Lewis, Yinhan Liu, Naman Goyal, Marjan Ghazvininejad, Abdelrahman Mohamed, Omer Levy, Ves Stoyanov, and Luke Zettlemoyer. 2019a. Bart: Denoising sequence-to-sequence pre-training for natural language generation, translation, and comprehension. *arXiv preprint arXiv:1910.13461*.

Patrick Lewis, Barlas Oğuz, Ruty Rinott, Sebastian Riedel, and Holger Schwenk. 2019b. Mlqa: Evaluating cross-lingual extractive question answering. *arXiv preprint arXiv:1910.07475*.

Diego Marcheggiani and Laura Perez-Beltrachini. 2018. Deep graph convolutional encoders for structured data to text generation. In *Proceedings of the 11th International Conference on Natural Language Generation*, pages 1–9.

Simon Mille, Anja Belz, Bernd Bohnet, Yvette Graham, Emily Pitler, and Leo Wanner. 2018. The first multilingual surface realisation shared task (sr'18): Overview and evaluation results. In *Proceedings of the First Workshop on Multilingual Surface Realisation*, pages 1–12. Association for Computational Linguistics.

Simon Mille, Anja Belz, Bernd Bohnet, Yvette Graham, and Leo Wanner. 2019. The Second Multilingual Surface Realisation Shared Task (SR'19): Overview and Evaluation Results. In *Proceedings of the 2nd Workshop on Multilingual Surface Realisation (MSR), 2019 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, Hong Kong, China.

Simon Mille, Roberto Carlini, Alicia Burga, and Leo Wanner. 2017. FORGe at SemEval-2017 task 9: Deep sentence generation based on a sequence of graph transducers. In *Proceedings of the 11th International Workshop on Semantic Evaluation (SemEval-2017)*, pages 920–923, Vancouver, Canada. Association for Computational Linguistics.

Myle Ott, Sergey Edunov, Alexei Baevski, Angela Fan, Sam Gross, Nathan Ng, David Grangier, and Michael Auli. 2019. fairseq: A fast, extensible toolkit for sequence modeling. In *Proceedings of NAACL-HLT 2019: Demonstrations*.

Telmo Pires, Eva Schlinger, and Dan Garrette. 2019. How multilingual is multilingual BERT? In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 4996–5001, Florence, Italy. Association for Computational Linguistics.

Matt Post. 2018. A call for clarity in reporting BLEU scores. In *Proceedings of the Third Conference on Machine Translation: Research Papers*, pages 186–191, Belgium, Brussels. Association for Computational Linguistics.

Nima Pourdamghani, Kevin Knight, and Ulf Hermjakob. 2016. Generating English from abstract meaning representations. In *Proceedings of the 9th International Natural Language Generation conference*, pages 21–25, Edinburgh, UK. Association for Computational Linguistics.

Leonardo F. R. Ribeiro, Claire Gardent, and Iryna Gurevych. 2019. Enhancing AMR-to-text generation with dual graph representations. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 3181–3192, Hong Kong, China. Association for Computational Linguistics.

Sascha Rothe, Shashi Narayan, and Aliaksei Severyn. 2019. Leveraging pre-trained checkpoints for sequence generation tasks. *ArXiv*, abs/1907.12461.

Holger Schwenk, Guillaume Wenzek, Sergey Edunov, Edouard Grave, and Armand Joulin. 2019. Ccmatrix: Mining billions of high-quality parallel sentences on the web. *arXiv preprint arXiv:1911.04944*.

Lei Sha, Lili Mou, Tianyu Liu, Pascal Poupart, Sujian Li, Baobao Chang, and Zhifang Sui. 2018. Order-planning neural text generation from structured data. In *Thirty-Second AAAI Conference on Artificial Intelligence*.

Anastasia Shimorina and Claire Gardent. 2019a. LORIA / lorraine university at multilingual surface realisation 2019. In *Proceedings of the 2nd Workshop on Multilingual Surface Realisation (MSR 2019)*, pages 88–93, Hong Kong, China. Association for Computational Linguistics.

Anastasia Shimorina and Claire Gardent. 2019b. Surface realisation using full delexicalisation. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 3086–3096, Hong Kong, China. Association for Computational Linguistics.

Marco Antonio Sobrevilla Cabezudo, Simon Mille, and Thiago Pardo. 2019. Back-translation as strategy to

tackle the lack of corpus in natural language generation from semantic representations. In *Proceedings of the 2nd Workshop on Multilingual Surface Realisation (MSR 2019)*, pages 94–103, Hong Kong, China. Association for Computational Linguistics.

Kaitao Song, Xu Tan, Tao Qin, Jianfeng Lu, and Tie-Yan Liu. 2019. MASS: Masked Sequence to Sequence Pre-training for Language Generation. In *ICML*, volume 97, pages 5926–5936.

Linfeng Song, Yue Zhang, Zhiguo Wang, and Daniel Gildea. 2018. A graph-to-sequence model for AMR-to-text generation. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1616–1626, Melbourne, Australia. Association for Computational Linguistics.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. 2017. Attention is all you need. In *Advances in neural information processing systems*, pages 5998–6008.

Guillaume Wenzek, Marie-Anne Lachaux, Alexis Conneau, Vishrav Chaudhary, Francisco Guzman, Armand Joulin, and Edouard Grave. 2019. Ccnet: Extracting high quality monolingual datasets from web crawl data. *arXiv preprint arXiv:1911.00359*.

Shijie Wu and Mark Dredze. 2019. Beto, bentz, becas: The surprising cross-lingual effectiveness of BERT. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 833–844, Hong Kong, China. Association for Computational Linguistics.

Yonghui Wu, Mike Schuster, Zhifeng Chen, Quoc V Le, Mohammad Norouzi, Wolfgang Macherey, Maxim Krikun, Yuan Cao, Qin Gao, Klaus Macherey, et al. 2016. Google's neural machine translation system: Bridging the gap between human and machine translation. *arXiv preprint arXiv:1609.08144*.

Xiang Yu, Agnieszka Falenska, Marina Haid, Ngoc Thang Vu, and Jonas Kuhn. 2019. IMSurReal: IMS at the surface realization shared task 2019. In *Proceedings of the 2nd Workshop on Multilingual Surface Realisation (MSR 2019)*, pages 50–58, Hong Kong, China. Association for Computational Linguistics.

Chao Zhao, Marilyn Walker, and Snigdha Chaturvedi. 2020. Bridging the structural gap between encoding and decoding for data-to-text generation. ACL.

Jie Zhu, Junhui Li, Muhua Zhu, Longhua Qian, Min Zhang, and Guodong Zhou. 2019. Modeling graph structure in transformer for better AMR-to-text generation. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 5459–5468, Hong Kong, China. Association for Computational Linguistics.

A Appendix

A.1 Model Training Details

For our baseline models, we train Transformer Big architectures with 240M parameters. We set the learning rate to 0.001 with the inverse square root learning rate schedule from Vaswani et al. (2017), warming up for 4000 updates. We train with the Adam optimizer with no weight decay and label smoothing 0.1. We set the encoder and decoder layerdrop to 0.1, and set the standard dropout to 0.3 as well. We experiment with dropout values between $0.1, 0.2, 0.3$. We set the number of maximum tokens per batch to 3584. We tune based on the validation loss at training time. We train for a fixed number of updates $(100,000)$ and take the best checkpoint by validation loss. We train using 8 GPUs. The overall training time varies depending on the amount of training data available. Overall, we train for about a day and a half to two days, though good performance can be achieved within a day. The remaining training only marginally improves the quality as measured by BLEU.

For the multilingual models, we train with the same parameters as above, except the parameter size is slightly larger: around 250M parameters. The reason for this is the increased size of the XLM vocabulary that we use for initializing our cross-lingual embeddings. We again tune the dropout values between $0.1, 0.2, 0.3$. As we use pretraining and pretrained cross-lingual embeddings, we lower the learning rate to 0.0001. A smaller learning rate can be used because the model parameters are initialized to a much better starting point. We warm up for 8000 updates to ease the learning rate schedule at the beginning of training. We experimented with a variety of learning rates between $0.001 - 0.00001$, and tried five different values in this range. We chose the best performing value based on validation loss. The convergence speed is faster for multilingual models due to the pretraining initialization. Good performance can be achieved within half a day, though for experimental consistency we continue to train for the full $100,000$ updates to compare to the baseline.

To generate from our models, we decode with beam search with beam size 5. We experiment with beam size values between $4, 5$ and length penalty values between $0.4, 0.6, 1, 1.2$. We tune these values based on validation BLEU and use the best performing values to decode on the test set. Our decoding process is as follows: generate with the model on the validation set, remove the sentencepiece markers, then use the `sacrebleu` library for evaluation. As we use `sacrebleu`, we provide detokenized text.