Modular Neural Machine Translation with a Semantic Pivot – Pilot Study Using AMR

Wenyang Gao[†*], **Yaxuan Li**[*], **Yunxin Bao**[†], **Shulin Huang**[*], **Yue Zhang**[*]
[†]Zhejiang University, [*]Westlake University
{gaowenyang, liyaxuan, huangshulin, zhangyue}@westlake.edu.cn
baoyunxin.cathy@outlook.com

Abstract

Neural machine translation (NMT) has become the predominant approach for automated translation, yet conventional models trained on extensive bilingual datasets exhibit critical limitations, including quadratic scaling of training data, sensitivity to out-of-distribution inputs, and a lack of interpretability. Inspired by the classical "translation pyramid" concept, which advocates for translation via a semantic pivot (interlingua), this work explores the integration of Abstract Meaning Representation (AMR) as a structured semantic intermediary to decouple translation into comprehension (source-to-AMR) and generation (AMR-to-target) phases. Concretely, we adopt English AMR as the pivot representation: all source sentences — regardless of their language — are mapped to English AMR graphs, which are then used to generate the target language. We conduct a pilot study using a strong AMR parser to create a multilingual silver-standard AMR corpus from the United Nations Parallel Corpus, covering six languages: Arabic (ar), Chinese (zh), English (en), French (fr), Russian (ru), and Spanish (es), training modular semantic understanding and generation components for each language. Experimental results demonstrate that our approach achieves an average improvement of 3% in robustness and over 15% in generalization compared to traditional Seq2Seq baselines. Analysis suggests that enhancing semantic parsing and generation accuracy could bridge the gap to conventional NMT systems. To our knowledge, this is the first work to integrate AMR as a semantic pivot in NMT, offering enhanced transparency, scalability, and robustness. This study underscores the potential of semantic-driven translation frameworks and provides a foundation for future research in interpretable, resource-efficient multilingual systems.

Keywords: machine translation, abstract meaning representation, semantic pivot

1. Introduction

Neural machine translation (NMT) (Bahdanau et al., 2014) has become the dominant method for machine translation. Traditional NMT methods use a neural string transducer architecture, typically the Transformer (Vaswani, 2017), which consists of a neural encoder and a neural decoder. The former is used to find a representation of the source input, while the latter generates the target output sequence. Trained on large datasets (e.g., tens of millions of bilingual sentence pairs), NMT models yield strong results for resource-rich languages. Recently, with advances in large language models (LLMs) (Touvron et al., 2023; Xu et al., 2023; Ganesh et al., 2025), decoder-only NMT systems have attracted increasing attention. These methods typically make use of a base LLM that contains knowledge across many languages and fine-tune the model with a relatively smaller amount of bilingual sentence pairs (e.g., tens of thousands). By leveraging the rich multilingual knowledge in base models, LLM-based MT has shown promising performance.

Despite their strong performance, existing NMT systems still face several fundamental challenges. First, they rely on extensive multilingual sentence pairs for training, which scales with square complexity (Koehn and Knowles, 2017). For traditional NMT systems, this can necessitate training $O(n^2)$ separate models. While modern multilingual NMT systems can consolidate this into a single universal model, the complexity and language balance of the training data remain a significant challenge(Johnson et al., 2016; Aharoni et al., 2019). Second, existing NMT systems have been shown to be sensitive to out-of-distribution (OOD) data (Yin et al., 2023; Li et al., 2024), and give low performances on low-resource language pairs. Spurious features have been shown to be frequent in sequence-to-sequence (Seq2Seq) transducers, which deviate from the human rationale for translation (Yin et al., 2022). Third, NMT models are not directly interpretable in their translation process, and thus, it can be difficult for monolingual speakers to identify potential errors (Belinkov and Glass, 2019).

One key cause of the above shortcomings is the Seq2Seq mapping nature of NMT models. They learn to synthesize the target translation conditioned on the source sequence token by token, rather than first comprehending the source and then yielding the target output according to the semantic representation (Bender et al., 2021). Surface-level mappings are shallow, require large bilingual training data, and can be prone to spurious features. As shown in Figure 1, in traditional machine translation research, a "translation pyramid" model has been discussed, which shows an ideal scenario that

Proceedings of the Workshop on Structured Linguistic Data and Evaluation, pages 74–85

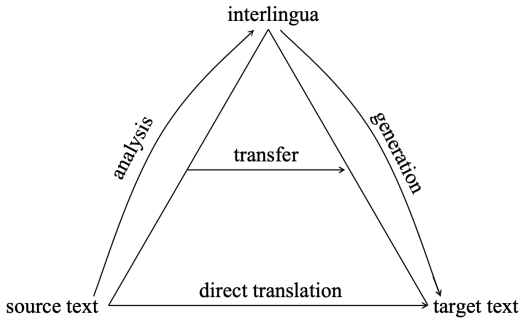

Figure 1: The machine translation pyramid categorizes approaches based on the extent of analysis and generation required. The interlingua approach involves comprehensive analysis and generation, while the direct translation approach minimizes both. The transfer approach falls between these two extremes.

makes use of a semantic pivot (or inter-lingua) for understanding-writing-style translation (Hutchins, 1995). A fundamental advantage of translating through a shared semantic representation lies in its inherent modularity and robustness.

The process involves: (i) Semantic Understanding: converting the source sentence into a language-independent meaning representation, and (ii) Surface Realization: generating the target sentence from this representation. This structure naturally avoids overfitting to superficial source-target correlations, leading to better generalization. Moreover, translation errors become interpretable—they stem either from inaccuracies in the semantic representation or from failures in the target language generation. Crucially, the modular design requires only O(n) components: each language necessitates one module for semantic encoding and one for surface generation, scaling efficiently to many languages.

The strong power of neural language models gives large potentials for building a translation system using a semantic pivot (Tenney et al., 2019). Unfortunately, little research has been done to this end in the literature (Matla et al., 2023). One reason may be that existing language models are trained over huge texts, yet it can be difficult to find semantic structures on the same scale (Raffel et al., 2019). In addition, there is no universally agreed semantic structure for cross-lingual representation. Despite the above challenges, some fundamental research in exploring the potential of semantic pivots can still provide valuable information. To this end, we take the Abstract Meaning Representation (AMR) (Banarescu et al., 2013) as the pivot and conduct a pilot study. Specifically, we adopt English AMR as the pivot throughout all translation directions. While AMR datasets for other languages do exist (e.g., Spanish, Chinese, and Czech AMR banks), their scale remains limited compared to English resources, and state-of-the-art AMR parsers are predominantly trained on English data. We therefore treat English AMR as a practical and well-supported choice for a first pivot study, while acknowledging that this introduces an inherent English-centric bias, a limitation we return to in Section 6.

Without aiming to immediately achieve competitive results to the state-of-the-art, we endeavor to answer the following research questions:

- Can we build a decent NMT system using existing semantic resources?
- Can a semantic pivot give more robust translation results compared to Seq2Seq transducer in the NMT context?
- With the huge potential space for enhancing semantic structures, can increase in the semantic parsing and semantic generation accuracies lead to strong translators?

In our experiments, we use a state-of-the-art AMR parser to annotate over 11 million data points in the United Nations Parallel Corpus (UN datasets), creating a multilingual silver AMR dataset, where we train our modules for each language. We combine theses semantic understanding modules and language generation modules into a complete multilingual neural machine translation (M-NMT) system, establishing a transparent pipeline architecture.

Our method shows an average improvement of 3% for each language pair in the robustness test, compared to the baseline model. In the generalization test, our model outperforms the baseline by over 15%. We also conduct a performance analysis, revealing that data quality is the main bottleneck. As data quality improves, our translation approach has the potential to achieve results comparable to, or even better than, traditional Seq2Seq models. To our knowledge, we are the first to incorporate AMR as a pivot in the context of NMT. Our semantic-understanding-based translation approach surpasses traditional models in robustness, generalization, and interpretability.

2. Related Work

Pivot-based NMT. Due to the lack of data in low-resource languages, conventional pivot-based NMT approaches (Johnson et al., 2017) enhance translations between low-resource languages by incorporating a high-resource language as a pivot. Some researchers (Leng et al., 2019) refine the process by adopting a learning-to-route (LTR) method to optimize the choices of the pivot languages and path from source to target. Another pivot-based

NMT paradigm involves generating more multilingual training data through data augmentation (Park et al., 2017; Currey and Heafield, 2019) or knowledge distillation (Chen et al., 2017; Ahmed and Buys, 2024), further fine-tuning the model to reduce the impact of data sparsity on model performance. Unlike previous methods, we alleviate the explicit semantic representation as a pivot and formulate the end-to-end translation into a human-like process involving a semantic understanding module and a language generation module. This human-like translation improves performance, robustness, and interoperability.

Transfer Learning in NMT. Transfer Learning (TL) is a subfield of Machine Learning that reuses knowledge from solving one task (the parent) to address a different but related task (the child) (Pan and Yang, 2010; Zhuang et al., 2021). Researchers fine-tune bilingual (Maimaiti et al., 2019; Maimaiti and Sun, 2020) or multilingual (Stickland et al., 2021) PLMs on low-resource language data to stimulate their translation capabilities. However, parallel data for low-resource languages are difficult to acquire and usually come from domains such as religious text (Siddhant et al., 2022). To transfer the abilities of multilingual PLMs, we fine-tune them on the AMR parsing task and AMR-to-text task respectively, acquiring the semantic understanding module and language generation module. In this process, we use monolingual data to eliminate reliance on low-resource parallel corpora.

3. Method

Our study proposes a modular semantic machine translation approach that introduces AMR as an intermediate semantic representation, which decomposes the translation task into two independent stages: semantic understanding and language generation. As shown in Figure 2, the semantic understanding module transforms text in the source language into an AMR graph, which is subsequently processed by the language generation module to generate text in the target language.

3.1. AMR as Pivot

AMR is a heavily abstracted graph-based representation of surface text, compressing semantics information into nodes and edges. Importantly, AMR was originally designed as an English-centric formalism: concept strings are grounded in English PropBank frames and vocabulary. In this work, we use English AMR exclusively as the semantic pivot, meaning that for all source languages, the semantic understanding module produces an English AMR graph, which the language generation module then realizes in the target language. This choice is motivated by the maturity of English AMR parsers and the relative abundance of English AMR-annotated data compared to other languages. Many researchers use AMR as a pivot in various NLP tasks, such as text paraphrasing (Fan and Gardent, 2020; Wang et al., 2020), text style transfer (Shi et al., 2023; Jangra et al., 2022), and translationese reduction (Wein and Schneider, 2023). Their studies prove that introducing the AMR-as-a-Pivot paradigm enhances performance and improves robustness for NLP tasks.

Seq2Seq models, while effective for sequential data, often fall short when dealing with the structural complexity of natural language. To address this, we leverage the depth-first search (DFS) algorithm, which aligns well with the linearized syntactic structures of natural language (Bevilacqua et al., 2021). For instance, the AMR graph in Figure 2 is linearized as: `( <pointer:0> remain-01 :ARG1 ( <pointer:1> incident :quant 1 ) :ARG3 ( <pointer:2> verify-01 :ARG1 <pointer:1> ) )`, where the `<pointer:i>` values are node coreference indices. To process AMR symbols effectively, we expand the vocabulary to include all relevant relations and frames. Moreover, we introduce a novel language token, `<amr_XX>`, to the existing language code. This allows multilingual models to seamlessly recognize and generate AMR as a pivot language, facilitating efficient cross-lingual translation.

3.2. Semantic Understanding Module

The purpose of the semantic understanding module is to extract and represent semantic information from the source language text as a structured AMR graph. This module can: (1) map surface-level words to specific nodes based on contextual clues, effectively handling polysemy and synonymy; and (2) represent semantic relationships between concepts as edges connecting nodes within the AMR graph. These capabilities are grounded in the ability to process structured sequences, as demonstrated by Transformer models, enabling the module to effectively capture the underlying semantic structure of the text.

In our proposed method, the semantic understanding module consists of an encoder and a decoder. The encoder is primarily tasked with reading the input sentence in natural language and transforming it into a high-dimensional dense vector representation that encapsulates the sentence's semantic content. Typically implemented as a Transformer, the encoder processes input word embeddings through multiple neural network layers, employing self-attention mechanisms to capture dependencies among words. Moreover, the encoder benefits from pre-training on large multilingual cor-

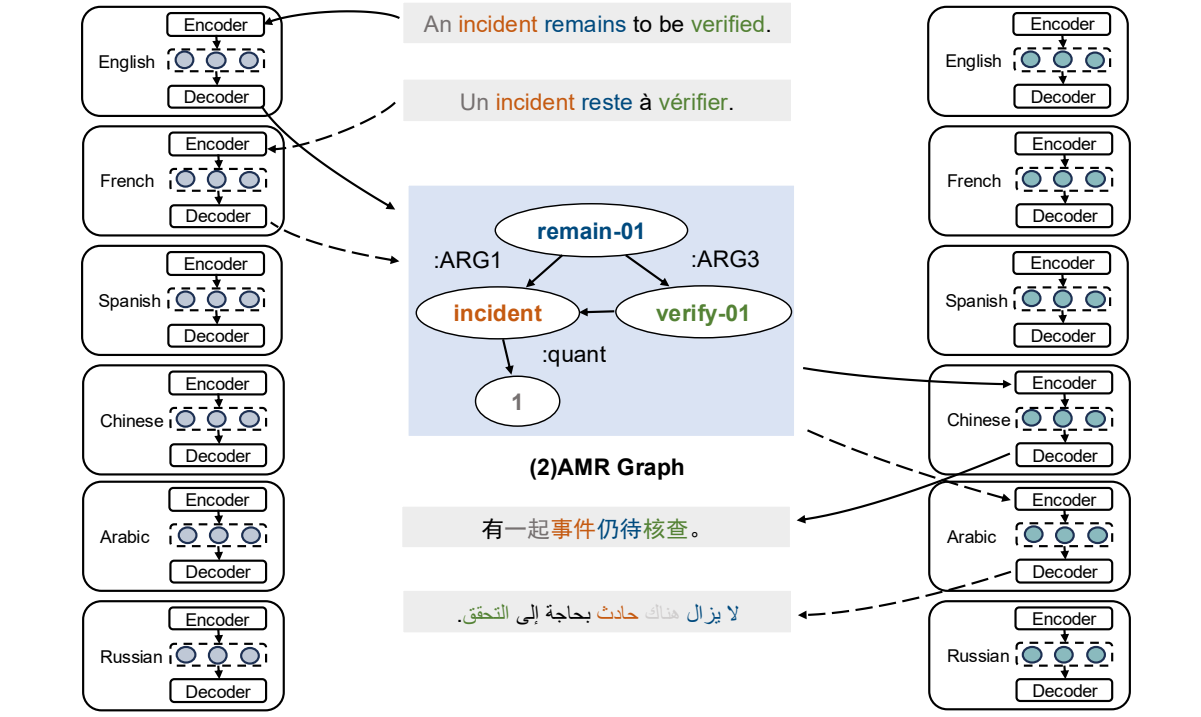

Figure 2: **Overview of the Proposed AMR-Pivot M-NMT Pipeline.** The pipeline illustrates the translation process for both English-to-Chinese (en-zh) and French-to-Arabic (fr-ar) pairs. Words in different languages that convey the same semantic meaning are highlighted with the same color, emphasizing alignment achieved through AMR pivot.

pora, which enables it to acquire cross-linguistic semantic features and enhances its generalizability across diverse source languages.

The decoder, in turn, converts the semantic vector representation generated by the encoder into a linearized AMR graph sequence. This sequence represents a directed graph where nodes correspond to concepts, and edges denote relationships between them. During training, the decoder's parameters are optimized by aligning its output with the ground truth AMR linear sequence, often through minimizing cross-entropy loss or similar loss functions suited to sequence generation tasks.

3.3. Language Generation Module

The language generation module aims to produce appropriate text in the target language based on the given AMR graph, ensuring that the generated translation is consistent with the semantic meaning of the source text.

The encoder's primary function is to transform the input AMR graph, which represents the underlying semantic structure through nodes (concepts) and edges (relationships between concepts), into a dense intermediate representation. This transformation is critical for capturing both the structure and meaning embedded in the AMR. In the language generation module, the encoder processes a linearized form of the AMR graph through self-attention mechanisms to capture the semantic relationships within the graph, thus producing a continuous vector representation that encapsulates the entire semantic meaning of the graph, which is subsequently processed by the decoder to generate text in the target language.

The decoder is tasked with converting the encoded semantic representation into a fluent sequence of text in the target language. It must ensure that the generated text accurately reflects the meaning represented by the AMR while adhering to the linguistic and syntactic rules of the target language. Typically integrated into a Transformer architecture, the decoder generates the target text autoregressively, token by token, by utilizing both the encoded semantic information and previously generated tokens. In multilingual settings, the decoder must accommodate cross-linguistic syntactic differences and word order variations to ensure natural and fluent output in the target language.

3.4. Modular Interaction

Here, we denote S as the source language, T as the target language, and AMR as the semantic representation.

- **Semantic Understanding Module**. The purpose of this module is to extract the semantic meaning from the source language text and convert it into a serialized AMR graph, which serves as the intermediate representation during the translation process $(S \rightarrow AMR)$.
- **Language Generation Module.** This module aims to produce appropriate text in the target language based on the given AMR graph, ensuring that the generated translation is consistent with the semantic meaning of the source text $(AMR \rightarrow T)$.

We denote $\mathcal{M}^{S}_{AMR}$ as the semantic understanding module for source language S, and $\mathcal{M}^{AMR}_{T}$ as the language generation module for the target language T. For each language, we fine-tune both modules on the corresponding data, resulting in a list of semantic understanding modules $\{\mathcal{M}^{S}_{AMR} \mid S \in \{ar, zh, en, fr, ru, es\}\}$ and a list of language generation modules $\{\mathcal{M}^{AMR}_{T} \mid T \in \{ar, zh, en, fr, ru, es\}\}$. Both modules are optimized using cross-entropy loss:

$$\mathcal{L}_{I,O} = -logP(\widehat{O}|I, O) \quad (1)$$

where (I, O) is the input and reference output for the module, and $(I, O) \in \{(S, AMR), (AMR, T)\}$. $\widehat{O}$ denotes the module-generated output.

As shown in Figure 2, this architecture allows for a flexible, modular approach to multilingual translation. A language-specific semantic understanding module and a language generation module can be combined to perform translation between any of the supported languages $(\mathcal{M}^{S}_{AMR} \oplus \mathcal{M}^{AMR}_{T})$. This process also yields an explicit semantic representation, which improves interpretability $(S \rightarrow AMR \rightarrow T)$.

4. Experiments

4.1. Datasets and Data Construction

We construct an extensive multilingual parallel dataset (UN-AMR) consisting of six official languages of the United Nations: Arabic, Chinese, English, French, Russian, and Spanish. We use the United Nations Parallel Corpus (Ziemski et al., 2016) and follow its original data split. We generate silver AMR annotations by parsing the corresponding English text using AMRBART (Bai et al., 2022). These AMR graphs serve as explicit semantic representations for texts in each language, paired with sentences in respective languages to form a parallel corpus with six subsets:

$$\mathcal{D} = \{(ar, AMR), (zh, AMR), (en, AMR), (fr, AMR), (ru, AMR), (es, AMR)\}$$

To assess the performance of our proposed modular semantic machine translation framework across varying data scales, we conducted comparative experiments using both the complete dataset and a subset from the UN-AMR corpus training partition. We denote the complete dataset as UN-AMR and the subset as UN-AMR-subset. The UN-AMR-subset consists of the first 640K instances of the complete dataset. The UN-AMR dataset is used to train models from scratch, while the UN-AMR-subset is employed for fine-tuning pre-trained models. The dataset statistics are detailed in Table 1. For each experimental condition, we executed parallel training procedures for both our model and baseline architectures, followed by comprehensive evaluations of their generalization capability and robustness across the corresponding dataset.

	#Sentences	
Split	**UN-AMR**	**UN-AMR-subset**
Train	11,365,709 (11.4M)	640,000 (640K)
Test	4,000	
Validation	4,000	

Table 1: Statistics of UN-AMR dataset.

4.2. Metrics

We evaluate translation quality using BLEU (Papineni et al., 2002) and COMET (Rei et al., 2020) on the standard UN-AMR test set and paraphrased versions to measure robustness. Additionally, we employ Smatch (Cai and Knight, 2013a) to evaluate the semantic accuracy of our AMR graph generation, comparing generated graphs against silver annotations.

BLEU. This metric is used to evaluate the quality of machine-translated text by comparing it to human-translated references. It calculates the n-gram precision between the machine-translated text and the reference translations. We evaluate the robustness of the translation system by measuring the rate of decline in BLEU scores.

COMET. A neural network-based metric for machine translation evaluation predicts human judgments by leveraging pre-trained models fine-tuned on human evaluation data. The model-based metric, COMET, offers a semantic-level assessment of machine translation quality.

Smatch. An evaluation metric specifically designed for semantic feature structures. It calculates a similarity score between two AMR graphs based on their structural and lexical similarities (Cai and Knight, 2013b). To evaluate our understanding modules' ability to capture semantic features, we

	AMRpivot on testset						AMRpivot on paraphrased testset					
$S \backslash T$	ar	en	es	fr	ru	zh	ar	en	es	fr	ru	zh
ar	-	37.0	35.8	27.3	25.0	38.6	-	29.2 $_{\downarrow 21\%}$	30.9 $_{\downarrow 14\%}$	23.6 $_{\downarrow 14\%}$	21.0 $_{\downarrow 16\%}$	33.9 $_{\downarrow 12\%}$
en	22.4	-	42.8	32.0	28.7	43.1	15.4 $_{\downarrow 31\%}$	-	32.1 $_{\downarrow 25\%}$	25.2 $_{\downarrow 21\%}$	22.2 $_{\downarrow 23\%}$	35.9 $_{\downarrow 17\%}$
es	19.9	41.7	-	30.6	26.3	39.3	16.5 $_{\downarrow 17\%}$	33.2 $_{\downarrow 20\%}$	-	27.1 $_{\downarrow 11\%}$	23.2 $_{\downarrow 12\%}$	35.8 $_{\downarrow 9\%}$
fr	17.4	35.7	35.9	-	24.3	36.2	14.6 $_{\downarrow 16\%}$	29.3 $_{\downarrow 18\%}$	31.6 $_{\downarrow 11\%}$	-	21.2 $_{\downarrow 12\%}$	33.3 $_{\downarrow 8\%}$
ru	17.9	35.3	34.1	27.2	-	36.3	14.7 $_{\downarrow 18\%}$	28.2 $_{\downarrow 20\%}$	29.8 $_{\downarrow 13\%}$	23.9 $_{\downarrow 12\%}$	-	32.7 $_{\downarrow 10\%}$
zh	17.1	34.4	33.1	25.3	23.4	-	14.4 $_{\downarrow 16\%}$	28.6 $_{\downarrow 17\%}$	29.5 $_{\downarrow 11\%}$	23.0 $_{\downarrow 9\%}$	20.7 $_{\downarrow 12\%}$	-
Avg↓	5.89 (16%)											

Table 2: Robustness results of our proposed model.

	Transformer on testset						Transformer on paraphrased testset					
$S \backslash T$	ar	en	es	fr	ru	zh	ar	en	es	fr	ru	zh
ar	-	42.95	40.48	28.26	28.35	39.0	-	31.7 $_{\downarrow 26\%}$	32.9 $_{\downarrow 19\%}$	25.4 $_{\downarrow 10\%}$	22.8 $_{\downarrow 20\%}$	35.1 $_{\downarrow 10\%}$
en	28.52	-	55.10	41.01	36.60	47.7	18.0 $_{\downarrow 37\%}$	-	37.6 $_{\downarrow 32\%}$	30.1 $_{\downarrow 27\%}$	25.3 $_{\downarrow 31\%}$	38.8 $_{\downarrow 19\%}$
es	21.96	45.41	-	39.02	31.61	42.3	18.1 $_{\downarrow 18\%}$	34.9 $_{\downarrow 23\%}$	-	31.6 $_{\downarrow 19\%}$	26.5 $_{\downarrow 16\%}$	38.8 $_{\downarrow 8\%}$
fr	19.31	43.49	45.52	-	29.87	24.6	16.1 $_{\downarrow 17\%}$	34.1 $_{\downarrow 22\%}$	36.7 $_{\downarrow 19\%}$	-	25.0 $_{\downarrow 16\%}$	22.3 $_{\downarrow 9\%}$
ru	16.87	41.96	40.06	30.93	-	40.3	14.5 $_{\downarrow 14\%}$	32.6 $_{\downarrow 22\%}$	32.5 $_{\downarrow 19\%}$	25.3 $_{\downarrow 18\%}$	-	36.4 $_{\downarrow 10\%}$
zh	18.49	41.02	38.15	30.56	27.49	-	15.2 $_{\downarrow 18\%}$	32.3 $_{\downarrow 21\%}$	31.3 $_{\downarrow 18\%}$	26.0 $_{\downarrow 15\%}$	23.0 $_{\downarrow 16\%}$	-
Avg↓	8.15 (19%)											

Table 3: Robustness results of baseline model.

calculate Smatch scores by comparing their generated AMR graphs against silver AMR annotations.

Human Evaluation. To supplement automatic metrics, we conduct human evaluation for en→zh and zh→ar. Bilingual annotators rate each translation on adequacy and fluency. Inter-annotator agreement is reported via Cohen's κ.

4.3. Baseline Models

We adopt mBART-cc25-large (hereafter referred to as mBART-25) (Liu et al., 2020) as the foundational architecture for our experiments. We fine-tune mBART-25 on the UN-AMR-subset. Both the semantic understanding module and the language generation module are based on this architecture in our proposed framework. Our AMR-pivot translation system is denoted as `AMRpivot`.

For a rigorous comparison, we also employ mBART-25 as the foundational model for the baseline system. We develop an M-NMT system comprising 30 models, each specifically tailored to a distinct language pair. These models are trained on datasets identical to those used in our method for the respective language pairs with source text as input and target text as output. Similar to our proposed method, the baseline model is denoted as `Transformer`.

4.4. Robustness Test

We evaluate the robustness of each translation framework by measuring how sensitive its output quality is to surface-level variation in the input. Specifically, we use GPT-4o[1] to paraphrase each source sentence in the test set while preserving its semantics[2], then translate both the original and paraphrased sentences using the fine-tuned models. BLEU scores are computed for both sets of translations against the original reference translations, and robustness is quantified as the **percentage decline in BLEU score** between the original and paraphrased inputs.

Why percentage decline, not absolute BLEU score. A critical consideration when interpreting these results is the asymmetry between `AMRpivot` and `Transformer` in terms of pre-trained knowledge. Both systems are built on mBART-25, which has been pre-trained on large multilingual corpora and already encodes substantial translation knowledge for the six UN languages. When fine-tuned directly on source–target sentence pairs, the baseline `Transformer` can leverage this pre-existing translation competence, leading to higher absolute BLEU scores on the standard test set. By contrast, our system `AMRpivot` fine-tunes mBART-25 on the AMR parsing and AMR-to-text tasks, neither of which directly exercises the model's translation knowledge. As a result, the two systems begin from meaningfully different effective starting points: `Transformer` benefits from a head start in bilingual signal, while `AMRpivot` must construct its translation capability almost entirely through the AMR intermediate representation.

In this context, comparing absolute BLEU scores conflates two different things: the quality of the un-

[1]The version employed is gpt-4o-2024-08-06.

[2]Prompt: "Please rewrite this sentence while keeping the semantics as unchanged as possible."

	BLEU		COMET	
	Random-test	CG-test	Random-test	CG-test
mBART-25	61.4	58.0 $_{\downarrow 6\%}$	72.7	63.0 $_{\downarrow 13\%}$
Transformer	51.2	43.2 $_{\downarrow 16\%}$	62.0	49.9 $_{\downarrow 20\%}$
AMRpivot	38.6	37.9 $_{\downarrow 2\%}$	47.4	47.5

Table 4: Generalizability on CoGnitnion.

derlying translation framework and the degree to which the model can exploit pre-trained bilingual knowledge. What we are specifically interested in here is the former — how much a framework's output degrades when the surface form of the input changes while meaning is preserved. A system that genuinely translates via semantic understanding should, in principle, be invariant to such surface variation, whereas a system that relies on surface co-occurrence patterns should degrade more. The percentage decline in BLEU score isolates exactly this property, making it the appropriate primary metric for robustness comparison. We nonetheless report absolute scores in full in Table 2 and 3 for completeness, and note that even on this metric, AMRpivot outperforms Transformer on certain language pairs (e.g., ru–ar and fr–zh) despite the inherent disadvantage described above.

Overall, our method achieves an average BLEU decline of 16% across language pairs on paraphrased inputs, compared to 19% for the Transformer baseline — an average improvement of 3 percentage points per language pair. As shown in Tables 2 and 3, the advantage is particularly pronounced for language pairs involving Arabic and Russian, which are relatively lower-resourced within mBART-25's pre-training data, suggesting that the semantic pivot provides the most benefit precisely where surface-level pre-trained knowledge is weakest.

4.5. Generalizability Test

Further, to better validate our conclusion that the translation framework we propose relies more on semantic representation than surface sentence structures compared to the traditional translation framework, we design a two-stage generalization test. In the first stage, based on the experimental results from Section 4.4, we selected the English-to-Chinese language direction (which achieved the highest BLEU score in both translation models) for verification. We train two models, namely Transformer and AMRpivot, from scratch using the entire UN-AMR dataset, consuming 80*A100 GPU hours of computational resources. It is important to note that, according to the scaling law, when the training data scale reaches the computational optimal boundary, the model performance improvement enters a plateau phase. The experimental configuration is nearly at this critical point, which ensures that the results effectively reflect the inherent characteristics of the model framework.

We adopt the CoGnition (Li et al., 2021) dataset for evaluation of generalizability. The dataset contains 216,246 training sentence pairs, and the test set is composed of two subsets: (1) the conventional test set (10,000 sentence pairs), which contains typical combinations of training vocabulary; (2) the combination generalization test set (CG-testset, 10,800 sentence pairs), which specifically constructs 2,160 novel compounds, with up to 5 atoms and 7 words. We fine-tuned the two models obtained in the previous step on the training dataset and tested them on both two subsets to distinguish the model's memory of surface structures from its generalization ability to deep semantics.

The comparative experimental results on both test subsets are shown in Table 4. As shown, our model performs almost identically on both the CG-test and the conventional test set, with even a slight increase in BLEU score, while other benchmark models show significant declines. This demonstrates that our translation framework enables the model to learn deeply from semantics, so the combination generalization of the data does not significantly affect model performance. It is worth mentioning that since the AMR used for training was derived from AMR parser-generated silver data, which has lower precision, the large dataset (11M instances) leads to significant error accumulation during the semantic understanding process. Therefore, directly comparing the absolute results would not be fair.

Beyond absolute performance, the contrast between AMRpivot and direct transduction models on the CG-test set reflects differences in inductive bias. The combination generalization split introduces novel lexical compounds unseen during training. For Seq2Seq models, translation quality partly relies on memorized surface co-occurrence patterns, and novel combinations therefore cause performance degradation.

In contrast, the AMR-based framework separates lexical realization from predicate-argument structure. The semantic understanding module maps surface expressions into structured graphs capturing events and relations, allowing many unseen lexical combinations to correspond to previously observed semantic substructures composed in new ways. Let $\mathcal{X}$ denote the surface string space and $\mathcal{Z}$ the semantic graph space. While $\mathcal{X}$ grows combinatorially with vocabulary size, $\mathcal{Z}$ consists of structured recombinations of a finite predicate and role inventory. Modeling translation via $\mathcal{Z}$ thus compresses the hypothesis space and promotes systematic recombination rather than memorization of surface patterns.

The near-identical performance of AMRpivot on

ID	Source	AMR	Translation	Reference
1	送往Shami医院的**孩子名单**	(<pointer:0> thing :ARG2-of (<pointer:1> list-01 :ARG1 (<pointer:2> person :ARG0-of (<pointer:3> have-rel-role-91 :ARG2 (**<pointer:4> client**)) :ARG1-of (<pointer:5> send-01 :ARG2 (<pointer:6> hospital :wiki - :name (<pointer:7> name :op1 <lit> Shami </lit> :op2 <lit> Hospital </lit>))))))	**Lists of clients** sent to Shami Sham Hospital.	**Lists of children** taken to Shami Hospital.
2	它是玻璃**做的**。	(<pointer:0> **make-01** :ARG1 (<pointer:1> it) :ARG2 (<pointer:2> glass))	It was **making** glasses of it.	It was **made of glass**.
3	在取消了对欧元汇率**下限**的同时，还将活期存款账户余额利率降为负数，为-0.75%		In addition to the removal of the **ceiling** on the exchange rate for the Euro, the interest rate on the balance of deposit accounts would be reduced to a negative level of -0.75 percent.	The removal of the **floor** with respect to the euro was accompanied by a further move into negative territory of the interest rate on sight deposit account balances to -0.75 percent.

Figure 3: Mistranslation instances for the outputs of two translation systems.

both test sets supports this interpretation, suggesting that explicit semantic abstraction provides an effective inductive bias for compositional generalization.

4.6. Human Evaluation

To complement the automatic metrics, we conduct a small-scale human evaluation assessing adequacy (meaning preservation) and fluency (grammaticality and naturalness), each rated on a 1–5 scale.

We focus on en→zh and zh→ar. For each language pair, we randomly sample 25 sentences from the UN-AMR test set, each evaluated in both its original and paraphrased form (using the same GPT-4o procedure as Section 4.4), yielding 50 translation instances per system. The two translations per sentence are presented to two bilingual annotators in randomized order without system labels, producing 200 judgments per dimension per language pair (25 × 2 conditions × 2 systems × 2 annotators). Disagreements are resolved by averaging, and inter-annotator agreement is reported via Cohen's κ.

As shown in Table 5, `AMRpivot` achieves higher adequacy on both language pairs under both conditions, with the advantage widening on paraphrased inputs, from $+0.14$ to $+0.31$ on en→zh and from $+0.13$ to $+0.27$ on zh→ar, which directly confirming the robustness findings of Section 4.4. `Transformer` scores higher on fluency throughout, reflecting its direct exposure to source–target pairs during fine-tuning. The fluency gap is larger on zh→ar (-0.54) than en→zh (-0.37), consistent with greater error accumulation through the AMR parsing stage on a more distant language pair. This adequacy–fluency tradeoff points to improving the language generation module — through higher-quality AMR annotations or stronger generative models — as the primary direction for future work.

Condition	System	en→zh		zh→ar	
		Adequacy	Fluency	Adequacy	Fluency
Original	`Transformer`	3.74	**3.98**	3.31	**3.52**
	`AMRpivot`	**3.88**	3.61	**3.44**	2.98
Paraphrased	`Transformer`	3.49	**3.74**	3.05	**3.16**
	`AMRpivot`	**3.80**	3.21	**3.32**	2.60
Cohen's κ		0.66	0.72	0.63	0.69

Table 5: Human evaluation results (mean scores on a 1–5 scale) for en→zh and zh→ar.

5. Interpretability Analysis

Figure 3 displays three instances of mistranslation. The first two examples are produced by our proposed model, while the third is generated by the Transformer baseline model. As shown, the first two cases indicate the effectiveness of our method in localizing error sources, distinguishing whether issues stem from the semantic understanding module or the language generation module, with AMR serving as a pivot. In contrast, the final example, generated by the traditional NMT system, fails to explicitly identify whether the errors originate from the encoder or decoder components.

For the first instance, according to AMR annotation conventions, `ARG1` typically represents the direct object of an action, while `ARG2` is often used for instrumental or comitative relations. In this case, Chinese characters meaning "Lists of children" are mistranslated as "Lists of clients". With the help of AMR pivot, the error can be easily identified. This

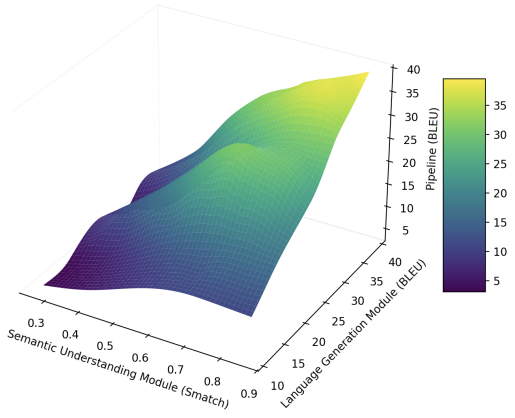

Figure 4: Overall translation performance of AMR-pivot with sacling abilities of semantic understanding modules and language generation modules..

mistake suggests that the model may have misinterpreted the Chinese characters meaning “children” during the semantic understanding process. By correcting the AMR parse structure, the model successfully generates the correct translation. In the second instance, the AMR graph is correctly extracted but it erroneously realizes the concept “make-01” as “making” during the language generation process and finally generates an incorrect target sequence.

In the third instance, Chinese characters meaning “floor” was erroneously translated as “ceiling” rather than “floor”. However, it is unclear where the error originates. Specifically, it is uncertain whether the mistake arises from a misinterpretation of the source language or an error in generating the target language. This demonstrates that the transformer model is unable to effectively identify such errors. This comparative analysis underscores an important distinction: While Transformer models rely on implicit vector representations that obscure error localization, our approach allows for explicit identification of error sources within specific processing modules. The opaque nature of intermediate representations in Transformer architectures complicates the diagnosis of translation errors, making it difficult to ascertain whether the issue arises from semantic misinterpretation or generation failures, thereby hindering targeted error correction. Moreover, this interpretability also lays the foundation for us to selectively improve the overall performance.

6. Challenges and Opportunities

We also train semantic understanding modules and language generation modules of varying capacities using different sizes of training datasets, then combine them pairwise into a pipeline to evaluate translation performance.

As shown in Figure 4, the performance of the AMR-based model is currently limited by the effectiveness of the semantic understanding module and the language generation module. With improvements in these two key components, we anticipate a gradual performance enhancement of AMRpivot. Specifically, as these modules advance, the translation performance will continue to improve.

Additionally, since we use English AMR as the pivot, the system carries an inherent English-centric bias. For non-English source languages, the semantic understanding module must bridge not only a linguistic gap but also a conceptual one, as AMR frames and concept strings are grounded in English lexical and syntactic conventions. This may disproportionately disadvantage morphologically rich or syntactically distant languages such as Arabic and Russian. While language-specific AMR banks exist for a small number of languages, their scale and parser maturity currently preclude their use at the data volumes required here. Future work should explore whether a more language-neutral semantic representation or a cross-lingual AMR formalism that decouples concepts from English vocabulary — could alleviate this bias and further boost translation quality across typologically diverse language pairs. Such a representation should focus on compressing redundant information, making it adaptable to various languages. This shift would not only enhance the interpretability of the model but also significantly boost its translation performance across multiple languages. Moreover, if language models are capable of computing rich, high-level structures rather than merely fragmented token-level representations, and if these structures can be aligned across languages, then the translation paradigm that operates without bilingual training data is likely to achieve significant progress.

7. Conclusion

We conducted a pilot study on using a semantic pivot to achieve neural machine translation without sequence-to-sequence transduction learning, finding that it has the potential to reduce the influence of spurious features, leading to more robust, interpretable, and less bilingual-data-hungry neural models. Additionally, although the existing method does not yet compete with state-of-the-art NMT systems, there is a salient trend: improving semantic parsing and generation accuracies leads to stronger translation quality. Such findings can shed light on further research on the new translation paradigm. Future research directions include finding a strong universal semantic representation and discovering new foundational models that can provide reliable text-level semantic information, rather than relying on token-level vectors.

8. Acknowledgements

This work has been financially supported by the National Key R&D Program of China (Grant No. 2022YFE0204900) and its Macao counterpart funded by FDCT, Macao SAR (Grant No. FDCT/0070/2022/AMJ).

9. Bibliographical References

Roee Aharoni, Melvin Johnson, and Orhan Firat. 2019. Massively multilingual neural machine translation. *ArXiv*, abs/1903.00089.

Khalid Ahmed and Jan Buys. 2024. Neural machine translation between low-resource languages with synthetic pivoting. In *Proceedings of the 2024 Joint International Conference on Computational Linguistics, Language Resources and Evaluation, LREC/COLING 2024, 20-25 May, 2024, Torino, Italy*, pages 12144–12158. ELRA and ICCL.

Dzmitry Bahdanau, Kyunghyun Cho, and Yoshua Bengio. 2014. Neural machine translation by jointly learning to align and translate. *CoRR*, abs/1409.0473.

Xuefeng Bai, Yulong Chen, and Yue Zhang. 2022. Graph pre-training for AMR parsing and generation. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), ACL 2022, Dublin, Ireland, May 22-27, 2022*, pages 6001–6015. 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 *LAW@ACL*.

Yonatan Belinkov and James Glass. 2019. Analysis methods in neural language processing: A survey. *Transactions of the Association for Computational Linguistics*, 7:49–72.

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

Michele Bevilacqua, Rexhina Blloshmi, and Roberto Navigli. 2021. One SPRING to rule them both: Symmetric AMR semantic parsing and generation without a complex pipeline. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 35, pages 12564–12573.

Shu Cai and Kevin Knight. 2013a. Smatch: an evaluation metric for semantic feature structures. In *Proceedings of the 51st Annual Meeting of the Association for Computational Linguistics, ACL 2013, 4-9 August 2013, Sofia, Bulgaria, Volume 2: Short Papers*, pages 748–752. The Association for Computer Linguistics.

Shu Cai and Kevin Knight. 2013b. 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.

Yun Chen, Yang Liu, Yong Cheng, and Victor O. K. Li. 2017. A teacher-student framework for zero-resource neural machine translation. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics, ACL 2017, Vancouver, Canada, July 30 - August 4, Volume 1: Long Papers*, pages 1925–1935. Association for Computational Linguistics.

Anna Currey and Kenneth Heafield. 2019. Zero-resource neural machine translation with monolingual pivot data. In *Proceedings of the 3rd Workshop on Neural Generation and Translation@EMNLP-IJCNLP 2019, Hong Kong, November 4, 2019*, pages 99–107. Association for Computational Linguistics.

Angela Fan and Claire Gardent. 2020. Multilingual amr-to-text generation. *arXiv preprint arXiv:2011.05443*.

Devalla Bhaskar Ganesh, Paruchuri Eesha Chowdary, Dokuparthi Nilesh, Jonnala HarshaVardhan Reddy, Chittela Venkata Sai Tarun Reddy, and Suryakanth V. Gangashetty. 2025. Advances in machine translation: A comprehensive survey of large language models. *2025 3rd International Conference on Intelligent Data Communication Technologies and Internet of Things (IDCIoT)*, pages 1671–1675.

W. John Hutchins. 1995. Machine translation: A brief history.

Anubhav Jangra, Preksha Nema, and Aravindan Raghuveer. 2022. T-star: Truthful style transfer using amr graph as intermediate representation. *arXiv preprint arXiv:2212.01667*.

Melvin Johnson, Mike Schuster, Quoc V. Le, Maxim Krikun, Yonghui Wu, Z. Chen, Nikhil Thorat, Fernanda B. Viégas, Martin Wattenberg, Gregory S. Corrado, Macduff Hughes, and Jeffrey Dean.

2016. Google's multilingual neural machine translation system: Enabling zero-shot translation. *Transactions of the Association for Computational Linguistics*, 5:339–351.

Melvin Johnson, Mike Schuster, Quoc V. Le, Maxim Krikun, Yonghui Wu, Zhifeng Chen, Nikhil Thorat, Fernanda B. Viégas, Martin Wattenberg, Greg Corrado, Macduff Hughes, and Jeffrey Dean. 2017. Google's multilingual neural machine translation system: Enabling zero-shot translation. *Trans. Assoc. Comput. Linguistics*, 5:339–351.

Philipp Koehn and Rebecca Knowles. 2017. Six challenges for neural machine translation. In *Proceedings of the First Workshop on Neural Machine Translation*, pages 28–39, Vancouver. Association for Computational Linguistics.

Yichong Leng, Xu Tan, Tao Qin, Xiang-Yang Li, and Tie-Yan Liu. 2019. Unsupervised pivot translation for distant languages. In *Proceedings of the 57th Conference of the Association for Computational Linguistics, ACL 2019, Florence, Italy, July 28- August 2, 2019, Volume 1: Long Papers*, pages 175–183. Association for Computational Linguistics.

Yafu Li, Yongjing Yin, Yulong Chen, and Yue Zhang. 2021. On compositional generalization of neural machine translation. *arXiv preprint arXiv:2105.14802*.

Yafu Li, Huajian Zhang, Jianhao Yan, Yongjing Yin, and Yue Zhang. 2024. What have we achieved on non-autoregressive translation? *ArXiv*, abs/2405.12788.

Yinhan Liu, Jiatao Gu, Naman Goyal, Xian Li, Sergey Edunov, Marjan Ghazvininejad, Mike Lewis, and Luke Zettlemoyer. 2020. Multilingual denoising pre-training for neural machine translation. *Transactions of the Association for Computational Linguistics*, 8:726–742.

Mieradilijiang Maimaiti, Yang Liu, Huanbo Luan, and Maosong Sun. 2019. Multi-round transfer learning for low-resource NMT using multiple high-resource languages. *ACM Trans. Asian Low Resour. Lang. Inf. Process.*, 18(4):38:1–38:26.

Yang Liu Huanbo Luan Mieradilijiang Maimaiti and Maosong Sun. 2020. Enriching the transfer learning with pre-trained lexicon embedding for low-resource neural machine translation. *Tsinghua Science & Technology*, page 1.

Syed Matla, Ul Qumar, Muzaffar Azim, and S. M. K. Quadri. 2023. Neural machine translation: A survey of methods used for low resource languages. *2023 10th International Conference on Computing for Sustainable Global Development (INDIACom)*, pages 1640–1647.

Sinno Jialin Pan and Qiang Yang. 2010. A survey on transfer learning. *IEEE Trans. Knowl. Data Eng.*, 22(10):1345–1359.

Kishore Papineni, Salim Roukos, Todd Ward, and Wei-Jing Zhu. 2002. Bleu: a method for automatic evaluation of machine translation. In *Proceedings of the 40th Annual Meeting of the Association for Computational Linguistics, July 6-12, 2002, Philadelphia, PA, USA*, pages 311–318. ACL.

Jaehong Park, Jongyoon Song, and Sungroh Yoon. 2017. Building a neural machine translation system using only synthetic parallel data. *CoRR*, abs/1704.00253.

Colin Raffel, Noam M. Shazeer, Adam Roberts, Katherine Lee, Sharan Narang, Michael Matena, Yanqi Zhou, Wei Li, and Peter J. Liu. 2019. Exploring the limits of transfer learning with a unified text-to-text transformer. *J. Mach. Learn. Res.*, 21:140:1–140:67.

Ricardo Rei, Craig Stewart, Ana C Farinha, and Alon Lavie. 2020. Comet: A neural framework for mt evaluation. *arXiv preprint arXiv:2009.09025*.

Kaize Shi, Xueyao Sun, Li He, Dingxian Wang, Qing Li, and Guandong Xu. 2023. Amr-tst: Abstract meaning representation-based text style transfer. In *Findings of the Association for Computational Linguistics: ACL 2023*, pages 4231–4243.

Aditya Siddhant, Ankur Bapna, Orhan Firat, Yuan Cao, Mia Xu Chen, Isaac Caswell, and Xavier Garcia. 2022. Towards the next 1000 languages in multilingual machine translation: Exploring the synergy between supervised and self-supervised learning. *CoRR*, abs/2201.03110.

Asa Cooper Stickland, Xian Li, and Marjan Ghazvininejad. 2021. Recipes for adapting pre-trained monolingual and multilingual models to machine translation. In *Proceedings of the 16th Conference of the European Chapter of the Association for Computational Linguistics: Main Volume, EACL 2021, Online, April 19 - 23, 2021*, pages 3440–3453. Association for Computational Linguistics.

Ian Tenney, Dipanjan Das, and Ellie Pavlick. 2019. Bert rediscovers the classical nlp pipeline. In *Annual Meeting of the Association for Computational Linguistics*.

Hugo Touvron, Thibaut Lavril, Gautier Izacard, Xavier Martinet, Marie-Anne Lachaux, Timothée

Lacroix, Baptiste Rozière, Naman Goyal, Eric Hambro, Faisal Azhar, Aurélien Rodriguez, Armand Joulin, Edouard Grave, and Guillaume Lample. 2023. Llama: Open and efficient foundation language models. *ArXiv*, abs/2302.13971.

A Vaswani. 2017. Attention is all you need. *Advances in Neural Information Processing Systems*.

Tianming Wang, Xiaojun Wan, and Hanqi Jin. 2020. Amr-to-text generation with graph transformer. *Transactions of the Association for Computational Linguistics*, 8:19–33.

Shira Wein and Nathan Schneider. 2023. Lost in translationese? reducing translation effect using abstract meaning representation. *arXiv preprint arXiv:2304.11501*.

Haoran Xu, Young Jin Kim, Amr Sharaf, and Hany Hassan Awadalla. 2023. A paradigm shift in machine translation: Boosting translation performance of large language models. *ArXiv*, abs/2309.11674.

Yongjing Yin, Yafu Li, Fandong Meng, Jie Zhou, and Yue Zhang. 2022. Categorizing semantic representations for neural machine translation. In *International Conference on Computational Linguistics*.

Yongjing Yin, Jiali Zeng, Yafu Li, Fandong Meng, Jie Zhou, and Yue Zhang. 2023. Consistency regularization training for compositional generalization. In *Annual Meeting of the Association for Computational Linguistics*.

Fuzhen Zhuang, Zhiyuan Qi, Keyu Duan, Dongbo Xi, Yongchun Zhu, Hengshu Zhu, Hui Xiong, and Qing He. 2021. A comprehensive survey on transfer learning. *Proc. IEEE*, 109(1):43–76.

Michal Ziemski, Marcin Junczys-Dowmunt, and Bruno Pouliquen. 2016. The united nations parallel corpus v1.0. In *Proceedings of the Tenth International Conference on Language Resources and Evaluation LREC 2016, Portorož, Slovenia, May 23-28, 2016*. European Language Resources Association (ELRA).