VLSP 2025 Shared Task: A Seq2Seq Transformer Approach to Vietnamese Semantic Parsing

Giang Huynh Thi Ha[1,2] **and Thang Nguyen Kien Bao**[1,2] **and Tam Vo Minh**[1,2]
[1] University of Information Technology, Ho Chi Minh City, Vietnam
[2] Vietnam National University, Ho Chi Minh City, Vietnam
21522021@gm.uit.edu.vn 21521432@gm.uit.edu.vn tamvm@uit.edu.vn

Abstract

This paper presents our work on Semantic Parsing in the VLSP 2025 shared task. Given limited computational resources, we fine-tuned BARTpho and ViT5, two encoder–decoder transformer models. We also employed tokenization strategies at both the syllable and word levels and treated punctuation marks as distinct tokens to improve AMR linearization and graph connectivity. In our experiments, BARTpho with word level tokenization achieved the highest Smatch score of 0.37 on the private test set provided by the organizers. These encouraging results highlight the impact of tokenization strategies on transformer-based encoder–decoder models for Vietnamese semantic parsing and suggest promising directions for future research.

Index Terms - Semantic Parsing; Abstract Meaning Representation; BARTpho.

1 Introduction

Semantic parsing aims to transform natural language into structured meaning representations, thereby supporting applications such as question answering, dialogue systems, and text-to-SQL (Zhong et al., 2020; Chen et al., 2021). Among various approaches, Abstract Meaning Representation (AMR) has emerged as a widely used semantic formalism, representing sentence meaning as directed acyclic graphs that capture predicate–argument structures, semantic roles, and coreference relations (Banarescu et al., 2013). Recent advances in sequence-to-sequence models such as T5 (Raffel et al., 2020) have substantially improved AMR parsing for English (Bevilacqua et al., 2021), but their adaptation to Vietnamese, an analytical and tonal language with challenging word segmentation and ambiguity (Vu and Nguyen, 2020), remains relatively underexplored.

Many initiatives on semantic parsing have been organized as academic competition, such as Spider: Yale Semantic Parsing and Text-to-SQL Challenge (Yu et al., 2018) and SemEval 2019 Task 1: Cross-lingual Semantic Parsing with UCCA (Hershcovich et al., 2019). The upcoming SemEval-2026 has even issued an open call for proposals on semantic parsing tasks (Association for Computational Linguistics, 2025). Within the Vietnamese research community, prior work has explored meaning representation at different levels. Early studies addressed semantic role labeling (Le et al., 2017; Pham et al., 2015), while more recent efforts extended to semantic dependency parsing for Vietnamese (Do and Nguyen, 2021), providing finer-grained sentence-level structures. However, full-scale semantic parsing in the Abstract Meaning Representation (AMR) framework remains largely underexplored. At the same time, Vietnamese NLP has benefited from advances in deep learning methods applied to both language and document understanding (Nguyen et al., 2020; Linh et al., 2020), suggesting the potential to transfer modern modeling strategies to semantic parsing. The Vietnamese Language and Speech Processing (VLSP) community has also played a central role in advancing core NLP tasks-including part-of-speech tagging, dependency parsing, and named entity recognition-through shared tasks and benchmark datasets. In this trajectory, the VLSP 2025 Shared Task 9 on Semantic Parsing provides the first community-wide benchmark for AMR-style parsing in Vietnamese, expected to catalyze progress by establishing datasets, evaluation protocols, and computational baselines for semantic parsing.

This paper presents our work for the Shared Task 9 of VLSP 2025: Semantic Parsing, which focuses on generating Abstract Meaning Representation (AMR) graphs (Banarescu et al., 2013) for Vietnamese sentences. We applied two tokenization strategies at word and syllable levels, motivated by previous findings on Vietnamese word segmentation challenges (Vu and Nguyen, 2020), and fine-tuned encoder-decoder models, BARTpho (Nguyen et al., 2022) and ViT5 (Phan et al., 2022), to produce linearized AMR representations. Under resource constraints, we achieved a Smatch score (Cai and Knight, 2013) of 0.37. Our contributions include:

- A comparison of pre-trained encoder–decoder models (Raffel et al., 2020; Lewis et al., 2020),
- An analysis of tokenization strategies for AMR linearization, and
- Insights into low-resource AMR parsing for Vietnamese.

2 Related Works

AMR parsing has seen significant progress internationally, driven by sequence-to-sequence models that linearize graphs for generation. Bevilacqua et al. (2021) introduced SPRING, a T5-based model that achieved state-of-the-art AMR parsing for English by encoding sentences into contextual embeddings and decoding linearized graphs token by token. In parallel, Xia et al. (2021) proposed LOME, which leverages transformer architectures for multilingual ontology extraction and demonstrates robustness across languages, although with limited coverage in low-resource settings. Cross-lingual AMR parsing has also been explored by Lyu et al. (2020), who applied transfer learning from English-trained models to other languages, but notably did not include Vietnamese.

In Vietnam, progress in NLP tasks, including parsing, has been driven by pre-trained models specifically designed for the language's morphological and tonal characteristics. PhoBERT (Nguyen et al., 2020), a BERT-based model with word- and subword-level tokenization, achieved state-of-the-art results in part-of-speech tagging, dependency parsing, and named entity recognition. Building on this, BARTpho (Nguyen et al., 2022) introduced a denoising autoencoder for sequence-to-sequence tasks, excelling in text generation with bidirectional encoding and autoregressive decoding. More recently, ViT5 (Phan et al., 2022), a Vietnamese adaptation of T5 (Raffel et al., 2020), was pretrained on large-scale corpora and obtained competitive results in summarization and machine translation. Collectively, these models provide a strong foundation for adapting modern encoder–decoder architectures to complex tasks such as AMR parsing, where challenges like word segmentation and tonal disambiguation remain.

Despite these advances, research on Vietnamese semantic parsing remains limited. Early efforts focused on building treebanks and developing methods for syntactic parsing (Viet et al., 2007) and dependency parsing (Thi et al., 2013; Nguyen et al., 2016; Do and Nguyen, 2021). Notably, the first large-scale Vietnamese Text-to-SQL dataset (Tuan Nguyen et al., 2020) was released, though derived from English translations rather than native annotation. However, comprehensive graph-based formalisms such as AMR, which enable structured semantic representation, have not yet been systematically explored.

The VLSP workshops have significantly advanced Vietnamese NLP by releasing benchmark datasets and hosting shared tasks on dependency and constituency parsing (Linh et al., 2020). These initiatives laid the groundwork for higher-level semantic analysis. Extending this trajectory, VLSP 2025 Shared Task 9 introduces the first community-wide benchmark on AMR-style semantic parsing for Vietnamese, offering a timely opportunity for evaluation in a low-resource setting. Our study contributes by systematically investigating AMR parsing with pre-trained encoder–decoder architectures and Vietnamese-specific tokenization strategies.

3 Dataset and Proposed Method

3.1 Task definition

In the VLSP 2025 Semantic Parsing shared task, the goal is to build a semantic parser for Vietnamese. Given a Vietnamese sentence, the system must generate a structured semantic representation in PENMAN format (e.g. AMR, or some specified semantic graph/logical form) that captures the syntactic and semantic relations within the sentence. The representation should include semantic roles, relationships between concepts and the logical structure underlying the sentence meaning.

3.2 Dataset and Preprocessing

The VLSP 2025 Shared Task organizers have provided a dataset of 1,841 sentences, of which 1,518 AMRs were successfully parsed and considered valid. Dataset statistics are presented in Table 1. On average, each sentence contains 11.99 tokens, with 7.07 nodes and 6.43 edges per AMR graph.

Description	Value
Valid AMRs	1,518
Avg. sentence length	11.99
Avg. nodes per AMR	7.07
Avg. edges per AMR	6.43
Reentrancy nodes	612

Table 1: Summary statistics of the dataset.

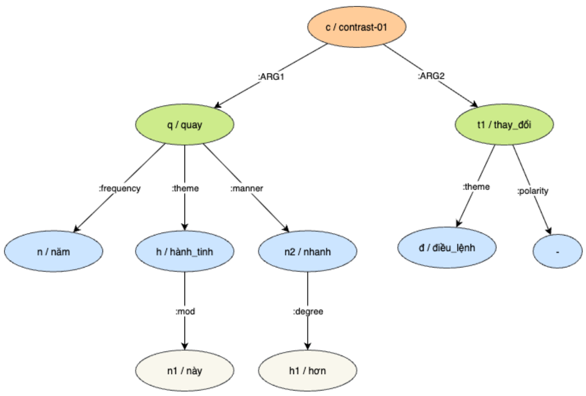

Figure 1: An example of AMR structure for a sentence.

The dataset also includes 612 reentrancy nodes, indicating moderate structural complexity.

Figure 1 illustrates an AMR-based semantic analysis of a Vietnamese sentence. AMR encodes sentence meaning as a semantic graph, where concepts are represented as nodes and semantic relations as labeled edges. In this example, the root node contrast-01 denotes the contrast between two events: (i) the event "hành tinh quay" (the planet rotates), with additional modifiers for time ("năm" – year, "này" – this) and manner ("nhanh hơn" – faster), and (ii) the event "điều lệnh thay đổi" (the command changes) with a negation attribute (:polarity -). This representation demonstrates how AMR abstracts the meaning of a sentence into a logical semantic structure, thereby supporting computational analysis beyond the surface word order.

To analyze relation frequency, we report the most common semantic roles: :agent (1,110), :mod (1,083), :theme (710), :pivot (554), and :compound (523). These distributions highlight the predominance of core argument and modifier relations.

For preprocessing, we apply the following steps:

1. Tokenization and normalization of input sentences.
2. Graph linearization using depth-first traversal.
3. Removal of AMRs with polarity inconsistencies.

These steps ensure that our models are trained on well-formed and representative AMRs while maintaining consistency with standard AMR parsing setups.

3.3 Pipeline and Model Strategy

Our method is based on the Transformer encoder–decoder paradigm, a widely adopted architecture for sequence-to-sequence generation tasks in natural language processing (Vaswani et al., 2017). As illustrated in Figure 2, the encoder maps an input sentence into contextualized representations, and the decoder autoregressively generates the target sequence. This architecture is particularly well-suited for semantic parsing, since structured outputs such as linearized AMR graphs can be represented as token sequences and subsequently transformed back into graph form.

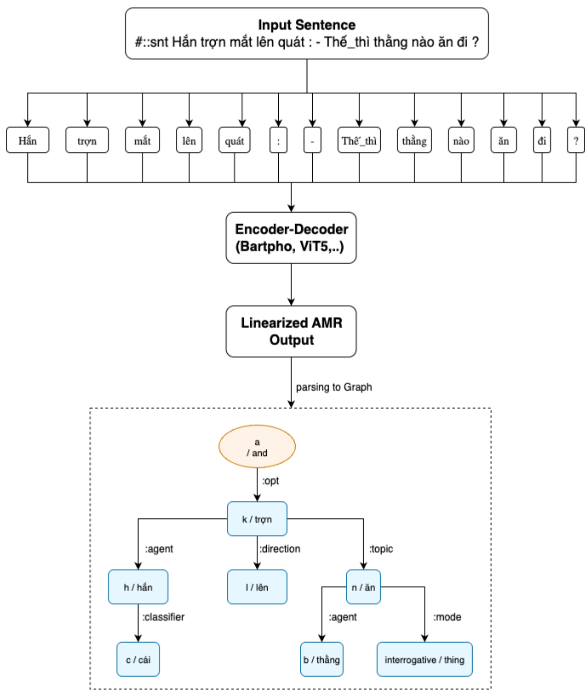

Figure 2: System pipeline for Vietnamese semantic parsing.

To instantiate this paradigm for Vietnamese AMR parsing, we adopt two pre-trained encoder–decoder models specifically developed for Vietnamese: **BARTpho** (Nguyen et al., 2022) and **ViT5** (Phan et al., 2022). Unlike multilingual models (e.g., mBART, mT5) that distribute their capacity across many languages, these models are pre-trained exclusively on large-scale Vietnamese corpora, making them especially suitable for low-resource settings where annotated AMR data is scarce. Leveraging Vietnamese-focused pretraining not only provides stronger semantic representations than multilingual or cross-lingual models but also improves tokenization granularity — for instance, BARTpho supports both syllable- and word-level tokenization. Overall, BARTpho and ViT5 offer a practical and effective foundation for Vietnamese text-to-AMR parsing in low-resource scenarios.

- **BARTpho-syllable** and **BARTpho-word** (Nguyen et al., 2022) adapt the BART architecture (Lewis et al., 2020) to Vietnamese, using syllable and word level tokenization schemes, respectively. The syllable level variant mitigates errors from Vietnamese word segmentation, whereas the word level variant explicitly leverages lexical boundaries. Both achieve competitive results in generation tasks such as summarization and translation, making them strong candidates for semantic parsing.

- **ViT5** (Phan et al., 2022), a Vietnamese adaptation of T5 (Raffel et al., 2020), is pretrained on large-scale Vietnamese corpora within a text-to-text framework. It employs a SentencePiece tokenizer that handles both sub-words and syllables, with the encoder processing the input sentence and the decoder generating the linearized AMR sequence token by token until an end-of-sequence marker is reached. In our setting, the input sentence is provided to the encoder, and the decoder generates a token-by-token linearized AMR sequence until an end-of-sequence marker is produced.

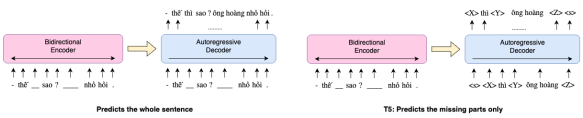

Figure 3: Pre-training objectives of BART vs. T5.

Figure 3 illustrates the difference in pre-training objectives between BART and T5. BART is trained as a denoising autoencoder that reconstructs entire corrupted sentences, while T5 reformulates all tasks in a text-to-text fashion, predicting only missing spans marked by sentinel tokens (adapted from Lewis et al., 2020; Raffel et al., 2020). This distinction motivates the Vietnamese adaptations BARTpho-syllable/word and ViT5, which are particularly suitable for AMR parsing. All models are fine-tuned on the VLSP 2025 semantic parsing dataset in the official linearized AMR format, enabling us to exploit their pre-trained generation capabilities to produce graph-structured semantic representations of Vietnamese sentences.

3.4 AMR Graph Reconstruction

As previously illustrated in Figure 2, our system processes each input instance by first applying model-specific tokenization, treating punctuation marks as independent tokens. The model then generates the corresponding AMR representation for the input sentence. Following this workflow, the final step involves a post-processing procedure that reconstructs well-formed AMR graphs from the linearized outputs. This procedure, detailed in Algorithm 1, draws inspiration from encoder-decoder approaches to AMR parsing (Konstas et al., 2017).

Algorithm 1 AMR Graph Reconstruction from Linearized Outputs

Require: Vietnamese sentence S, trained encoder-decoder model M
Ensure: Linearized AMR sequence L and reconstructed graph G
$T \leftarrow$ Tokenize(S) ▷ Word / Syllable / Subword
$L \leftarrow [\,]$ ▷ Initialize output sequence
while not end-of-sequence **do**
    $next\_token \leftarrow M.predict(T, L)$
    Append $next\_token$ to L
end while
$L \leftarrow$ PostProcess(L) ▷ Normalize variables, fix brackets, handle punctuation
$G \leftarrow$ LinearizedToGraph(L)
return L, G

Specifically, Algorithm 1 proceeds as follows. Here, S denotes the input Vietnamese sentence, and M is the pretrained encoder–decoder model. The sentence is tokenized into T using word-, syllable-, or subword-level segmentation. The model autoregressively predicts the next token until an end-of-sequence marker is reached, producing the linearized AMR sequence L. The function $PostProcess$ normalizes variables, corrects bracket mismatches, and handles punctuation. Finally, $LinearizedToGraph$ transforms L into the target AMR graph G.

4 Experiments

4.1 Configurations

We fine-tune both BARTpho and ViT5 models for the text-to-AMR parsing task, as detailed in Table 2. Due to hardware constraints, our setup accommodated a maximum batch size of 8. Fine-tuning was carried out on a local machine with a 15GB GPU, supplemented by complimentary GPU resources from Google Colab. Under these conditions, each run required approximately one hour to complete.

Table 2: Fine-tuning configurations for BARTpho and ViT5 models.

Configuration	**Value**
Pretrained Models	BARTpho, ViT5-base
Optimizer	AdamW
Learning Rate	5e-5
Batch Size	8
Max Input Length	256 tokens
Max Output Length	512 tokens
Training Epochs	15
Dropout	0.1
Beam Search	Beam size = 4
Training Time	~1 hours/run
Hardware	15GB GPU, Google Colab

For inference, we employed beam search with a beam size of 4 and a maximum sequence length of 256 tokens. We then applied post-processing scripts to normalize variable names, correct bracket mismatches, and ensure compatibility with the official AMR evaluation toolkit.

4.2 Results

Table 3 presents the performance of the three trained models on both the public and private test sets in terms of Smatch, Precision, and Recall. On the public test set, the BARTpho model with word-level tokenization achieved the best results across all three metrics, with scores of 0.33 (Smatch), 0.46 (Precision), and 0.26 (Recall). The large gap between Precision and Recall suggests that the model exhibits a conservative decoding behavior: predicting fewer but more accurate triples when confronted with structurally diverse or unseen data. Both BARTpho-syllable and ViT5 showed similar tendencies, yielding slightly lower scores than BARTpho-word, though the differences were not substantial.

On the private test set, BARTpho-word again achieved the highest overall performance, with a Smatch score of 0.37, Precision of 0.42, and Recall of 0.33, slightly outperforming ViT5 and BARTpho-syllable. The improvement is mainly

Table 3: Smatch, Precision, and Recall scores of the evaluated models on the public and private test sets.

Model	**Public Test**			**Private Test**		
	Smatch	**Prec.**	**Rec.**	**Smatch**	**Prec.**	**Rec.**
ViT5	0.31	0.39	0.30	0.35	0.42	0.30
BARTpho-syllable	0.31	0.39	0.29	0.34	0.42	0.28
BARTpho-word	**0.33**	**0.46**	0.26	**0.37**	0.42	**0.33**

attributed to its higher Recall, as Precision remained identical (0.42) across all models. Since the Smatch score is essentially the F1-score computed over matched AMR triples, this increase in Recall directly contributes to a higher overall Smatch. The uniform Precision across private configurations indicates comparable local accuracy in predicting AMR fragments.

When compared to the public test results, all models tend to achieve higher Recall and slightly lower Precision on the private test, leading to an overall improvement in the harmonic mean. Overall, BARTpho-word consistently outperforms the other models, confirming the advantages of the BART architecture and word-level tokenization strategy in producing more semantically coherent alignments between lexical units and AMR concepts.

4.3 Error Analysis on the Public Test Set

To better understand the model's behavior on unseen data, we conducted a manual and quantitative error analysis on the public test set. The qualitative inspection revealed three main categories of errors:

- **Concept Alignment Errors** ($\approx 40\%$)**:** The model often misaligns or omits compound verbs and idiomatic expressions. For example, in "*tôi nhớ lời anh chủ tịch xã Bùi Văn Luyến nhắc đi nhắc lại...*", the model merged "nhớ" and "nhắc" into a single concept, losing part of the predicate structure. This highlights its difficulty in modeling multiword predicates and nested verbal relations.

- **Relation Misclassification** ($\approx 35\%$)**:** Relations such as :pivot, :agent, and :topic are frequently confused. In "*chủ trương tốt nhưng dân không hiểu...*", the model replaced the intended :concession relation with a surface-level contrast-01, indicating sensitivity to discourse connectives like "nhưng", "thì", and "cũng".

- **Graph Structural Errors** ($\approx 25\%$)**:** The model occasionally produces malformed

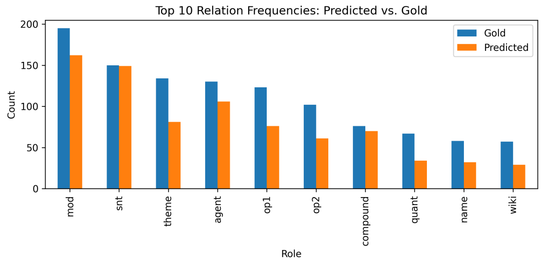

Figure 4: Comparison of the top 10 most frequent AMR relations in predicted and gold graphs on the public test set.

graphs with duplicated or missing edges. For instance, in "*nên trước nhất người đảng viên phải làm gương*", the root (*obligate_01*) was correctly identified, but modifiers were misattached, leading to structural inconsistencies.

These findings suggest that while the system effectively captures core predicate–argument structures, it still struggles with long-distance dependencies, nested coordination, and discourse-level reasoning. Future work may incorporate structural decoding constraints or syntactic–semantic alignment modules to alleviate such issues.

Quantitative Analysis. To complement the qualitative observations, we further analyzed the role distribution between the predicted and gold graphs. Figure 4 compares the relative frequencies of the most common AMR relations on the public test set. The model tends to overproduce surface-level modifiers (e.g., :mod, :topic) while underestimating core semantic roles such as :agent and :ARG0. This imbalance is consistent with the manual findings, indicating that while the model recognizes predicate structures, it often misrepresents argument hierarchy and role specificity. Such insights provide valuable guidance for refining relation-level modeling in future work.

5 Discussion: Lessons Learned

5.1 Effective Strategies

Our experiments highlight the critical role of tokenization granularity in AMR parsing performance. Word level tokenization (BARTpho-word) consistently outperformed syllable level and subword approaches, aligning with prior findings in Vietnamese NLP where preserving semantic units improved sequence-to-sequence generation (Pham et al., 2021; Vu and Nguyen, 2020). In addition, linearized AMR sequences benefited from punctuation-aware tokenization, which helped preserve graph structure integrity during autoregressive decoding.

5.2 Challenges and Limitations

Despite these promising results, several limitations were observed. First, low-resource training restricted the use of larger batch sizes and longer fine-tuning, which are known to improve generalization in sequence-to-sequence models (Guo et al., 2020). Second, domain-specific vocabulary including idiomatic expressions and unique Vietnamese named entities remained challenging across all models, echoing the cross-lingual transfer difficulties reported by (Lyu et al., 2020).

5.3 Insights from Vietnamese Data

The analytic and tonal properties of Vietnamese further shaped model behavior. While syllable level tokenization offered linguistic granularity, it often fragmented meaningful semantic units and led to lower Smatch scores. In contrast, word level tokenization more effectively captured predicate-argument relationships, underscoring the importance of aligning tokenization strategies with semantic roles (Phan et al., 2022). These findings reinforce the need to tailor preprocessing and tokenization to language-specific characteristics in semantic parsing tasks.

6 Conclusion

This study presents an adaptation of encoder–decoder transformer-based architectures for Vietnamese AMR parsing in low-resource settings. Fine-tuning BARTpho and ViT5 with targeted tokenization strategies enabled the effective generation of linearized AMRs, yielding encouraging results in the VLSP 2025 private test set. In particular, our experiments revealed that BARTpho with word level tokenization delivered higher performance than both its syllable level counterpart and the ViT5 models. These findings emphasize the crucial role of tokenization choices combined with punctuation-aware preprocessing in Vietnamese semantic parsing and open promising directions for improvement.

In future research, we aim to investigate the following improvements:

- **Enhanced Graph Linearization:** Exploring graph-to-sequence models that preserve hierarchical and reentrant structures more robustly (Bevilacqua et al., 2021).

- **Data Augmentation:** Leveraging cross-lingual transfer from English AMR corpora or synthetic Vietnamese data to improve low-resource performance (Xu and Zhang, 2021).
- **Incorporating Linguistic Features:** Integrating POS tags, dependency parses, and named entity recognition into the encoder to enrich semantic representations (Nguyen et al., 2020; Pham et al., 2021).
- **Scalable Training:** Employing larger GPU clusters or mixed precision training to fine-tune larger models without memory bottlenecks (Phan et al., 2022).
- **Evaluation on Downstream Tasks:** Assessing AMR utility in Vietnamese question answering, summarization, and reasoning tasks to measure real-world applicability (Shapira and Zhang, 2023; Bevilacqua et al., 2021).

References

Association for Computational Linguistics. 2025. First call for task proposals: Semeval 2026. https://www.aclweb.org/portal/content/first-call-task-proposals-semeval-2026. Accessed: 2025-09-10.

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. Association for Computational Linguistics.

Michele Bevilacqua, Rexhina Blloshmi, and Roberto Navigli. 2021. One spring to rule them all: Amr parsing with sequence-to-sequence models. In *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 1319–1334. Association for Computational Linguistics.

Deng Cai and Wai Lam. 2020. Amr parsing via graph-sequence iterative inference. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics (ACL)*, pages 1290–1301. ACL.

Shu Cai and Kevin Knight. 2013. Smatch: an evaluation metric for semantic feature structures. In *Proceedings of the 51st Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, pages 748–752. Association for Computational Linguistics.

Xinyun Chen, Kevin Lin, and Zhou Yu. 2021. Seq2seq generation for semantic parsing: Current state and challenges. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pages 1284–1295. Association for Computational Linguistics.

T.T.T. Do and D.T. Nguyen. 2021. Sentential semantic dependency parsing for vietnamese. *SN Computer Science*, 2(323).

Zhijiang Guo, Yue Zhang, and Wei Lu. 2020. Low-resource neural amr parsing with cross-lingual transfer. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics (ACL)*, pages 3711–3722. ACL.

Daniel Hershcovich, Zohar Aizenbud, Leshem Choshen, Elior Sulem, Ari Rappoport, and Omri Abend. 2019. SemEval-2019 task 1: Cross-lingual semantic parsing with UCCA. In *Proceedings of the 13th International Workshop on Semantic Evaluation*, pages 1–10, Minneapolis, Minnesota, USA. Association for Computational Linguistics.

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. Association for Computational Linguistics.

Taku Kudo and John Richardson. 2018. Sentencepiece: A simple and language independent subword tokenizer and detokenizer for neural text processing. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing: System Demonstrations*, pages 66–71. Association for Computational Linguistics.

Hong Phuong Le, Thai Hoang Pham, Xuan Khoai Pham, Thi Minh Huyen Nguyen, Thi Luong Nguyen, and Minh Hiep Nguyen. 2017. Vietnamese semantic role labelling. *VNU Journal of Science: Computer Science & Communication Engineering*, 33(2):39–58.

Mike Lewis, Yinhan Liu, Naman Goyal, Marjan Ghazvininejad, Abdelrahman Mohamed, Omer Levy, Veselin Stoyanov, and Luke Zettlemoyer. 2020. BART: Denoising sequence-to-sequence pre-training for natural language generation, translation, and comprehension. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 7871–7880, Online. Association for Computational Linguistics.

HA My Linh, NGUYEN Thi Minh Huyen, VU Xuan Luong, NGUYEN Thi Luong, PHAN Thi Hue, and

LE Van Cuong. 2020. Vlsp 2020 shared task: Universal dependency parsing for vietnamese. In *Proceedings of the 7th International Workshop on Vietnamese Language and Speech Processing*, pages 77–83.

Chenchen Lyu, Shu Cai, and Laura Banarescu. 2020. Cross-lingual amr parsing with transfer learning. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing*, pages 1369–1378. Association for Computational Linguistics.

Dat Quoc Nguyen, Mark Dras, and Mark Johnson. 2016. An empirical study for vietnamese dependency parsing. In *Australasian Language Technology Association Workshop 2016*, pages 143–149. Australasian Language Technology Association. Australasian Language Technology Association Workshop (14th : 2016) ; Conference date: 05-12-2016 Through 07-12-2016.

Dat Quoc Nguyen, Duong Minh Le, and Nguyen Luong Tran. 2022. BARTpho: Pre-trained sequence-to-sequence models for Vietnamese. *arXiv preprint arXiv:2109.09701*.

Dat Quoc Nguyen, Anh Tuan Nguyen, Dinh Pham, Xuan-Phi Nguyen, Son Nguyen, and Thanh Vu. 2020. Phobert: Pre-trained language models for vietnamese. In *Findings of the Association for Computational Linguistics: EMNLP 2020*, pages 1037–1042. Association for Computational Linguistics.

Thai Hoang Pham, Xuan Khoai Pham, and Hong Phuong Le. 2015. Building a semantic role labelling system for vietnamese. In *Proceedings of the 10th International Conference on Digital Information Management (ICDIM)*, pages 77–84.

V. Pham, T. Nguyen, and D. Nguyen. 2021. State-of-the-art vietnamese nlp: A survey. *Journal of Vietnamese Language and Linguistics*, 5(2):45–67.

Long Phan, Hieu Tran, Hieu Nguyen, and Trieu H. Trinh. 2022. Vit5: Pretrained text-to-text transformer for vietnamese language generation. In *Proceedings of the NAACL 2022 Student Research Workshop*, pages 136–142. Association for Computational Linguistics.

Colin Raffel, Noam Shazeer, Adam Roberts, Katherine Lee, Sharan Narang, Michael Matena, Yanqi Zhou, Wei Li, and Peter J. Liu. 2020. Exploring the limits of transfer learning with a unified text-to-text transformer. In *Journal of Machine Learning Research*, volume 21, pages 1–67.

M. Shapira and L. Zhang. 2023. Multilingual amr parsing: A survey. *Journal of Natural Language Processing*, 28(1):1–23.

Luong Nguyen Thi, Linh Ha My, Hung Nguyen Viet, Huyen Nguyen Thi Minh, and Phuong Le Hong. 2013. Building a treebank for vietnamese dependency parsing. In *The 2013 RIVF International Conference on Computing & Communication Technologies-Research, Innovation, and Vision for Future (RIVF)*, pages 147–151. IEEE.

Anh Tuan Nguyen, Mai Hoang Dao, and Dat Quoc Nguyen. 2020. A pilot study of text-to-SQL semantic parsing for Vietnamese. In *Findings of the Association for Computational Linguistics: EMNLP 2020*, pages 4079–4085, Online. 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 *Proceedings of the 31st International Conference on Neural Information Processing Systems*, pages 6000–6010.

Hoang Anh Viet, Dinh Thi Phuong Thu, and Huynh Quyet Thang. 2007. Vietnamese parsing applying the pcfg model. In *Proceedings of the Second Asia Pacific International Conference on Information Science and Technology, Vietnam*.

Xuan-Son Vu and Dat Quoc Nguyen. 2020. Vietnamese nlp resources: Word segmentation, pos tagging, and dependency parsing. In *Proceedings of the 12th Language Resources and Evaluation Conference*, pages 2145–2151. European Language Resources Association.

Qingrong Xia, Linfeng Song, Yixiao Li, and Jinsong Wang. 2021. Lome: Large ontology multilingual extraction. In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics*, pages 3214–3225. Association for Computational Linguistics.

X. Xu and Y. Zhang. 2021. Cross-lingual transfer learning for amr parsing. *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics*, pages 123–145.

Tao Yu, Rui Zhang, Kai Yang, Michihiro Yasunaga, Dongxu Wang, Zifan Li, James Ma, Irene Li, Qingning Yao, Shanelle Roman, and 1 others. 2018. Spider: A large-scale human-labeled dataset for complex and cross-domain semantic parsing and text-to-sql task. *arXiv preprint arXiv:1809.08887*.

Victor Zhong, Caiming Xiong, and Richard Socher. 2020. Semantic parsing with pre-trained language models. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 4312–4327. Association for Computational Linguistics.