Enhancing Lexical Relation Mining with Structured Sememe Knowledge

Hansi Wang[1,2], Qiliang Liang[1,2], Yue Wang[1,2], Yang Liu[1,2]*
[1]Key Laboratory of Computational Linguistics, Ministry of Education, Peking University
[2]School of Computer Science, Peking University
wanghansi2019@pku.edu.cn,lql.pkucs@gmail.com,{wyy209, liuyang}@pku.edu.cn

Abstract

Lexical Relation Mining (LRM) aims to identify and classify lexical relations between word pairs. In this paper, we focus on two subtypes of LRM: Lexical Relation Classification (LRC) and Lexical Entailment (LE). Existing top-performing methods for them rely heavily on Pre-trained Language Models (PLMs) yet fail to distinguish nuanced lexical relations. From a linguistic perspective, intralexical tree-structured sememe information can reflect interlexical relations. Inspired by this, we are motivated to explore leveraging such structured knowledge to enhance LRC and LE. We first propose an automated Sememe Tree Construction (STC) pipeline to predict sememe trees; Then, we present the SememeLRM method to fully leverage structured sememe knowledge; Experimental results show that it achieves a notable 1.6% improvement on average across benchmarks, even outperforming Large Language Model (LLM)-based methods that contain 20 times more parameters on most benchmarks. Further results also suggest that sememe trees predicted by our pipeline can rival the gold-standard in HowNet, extending their applicability to lexico-semantic computing. Overall, this paper presents a potentially generalizable framework for leveraging complete sememe trees and makes significant progress, helping to unlock the value of such intralexical knowledge in downstream tasks[1].

1 Introduction

Lexical Relation Mining (LRM) aims to identify and classify lexical relations between pairs of words (Sun et al., 2025) (e.g., antonymy between *fast* and *slow*). This task holds multi-faceted significance in NLP: It provides a reasonable way to evaluate the effectiveness and interpretability of word embeddings, as lexical relations can be probed more accurately through the differences between superior embeddings (Finley et al., 2017; Jadhav et al., 2020); It benefits many downstream tasks, such as Analogical Reasoning (Li et al., 2020), Sentiment Analysis (Xiang et al., 2021), and Knowledge Graph Completion (Wang et al., 2025); Besides, it focuses on relation representation learning, which potentially facilitates other NLP applications (Ushio et al., 2021), including Information Retrieval (Bordawekar and Shmueli, 2017) and Ontology Learning (Bouraoui and Schockaert, 2019).

There are various subtypes of LRM specialized in diverse relation mining objectives (Sun et al., 2025). In this work, we focus on Lexical Relation Classification (LRC) and Lexical Entailment (LE). Specifically, LRC aims to assign predefined relation labels (e.g., synonymy, hypernymy, antonymy) to word pairs, while LE is designed to quantify the entailment degree between two words.

Recent works on LRC and LE have mainly explored fine-tuning Pre-trained Language Models (PLMs) (Wang et al., 2021; Ushio et al., 2021; Pitarch et al., 2023). They leverage contextualized word embeddings and lexico-semantic knowledge acquired during pre-training to capture latent interlexical relations. Based on this, several works have attempted to integrate additional lexico-semantic information, such as textual definitions (Moskvoretskii et al., 2024) and lexical graph features (Sun et al., 2025) for performance gains. However, these methods still struggle to distinguish nuanced semantic relations. For example, they often confuse synonymy with hypernymy, as evidenced by the error analysis from Sun et al. (2025). Accordingly, it is necessary to incorporate fine-grained lexico-semantic knowledge, like sememes, into the model to deal with the challenges in such nuanced semantic understanding scenarios.

From a linguistic perspective, word senses can be decomposed into smaller units, and the smallest indivisible semantic units are called se-

*Corresponding author

[1]The resources and codes for this paper are available at https://github.com/COOLPKU/SememeLRM

Proceedings of the 64th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), pages 4930–4947

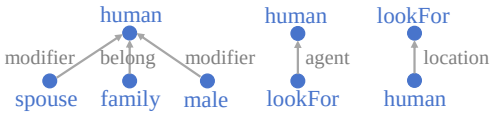

Figure 1: Sememe trees of the words *husband* (left), *searcher* (middle) and *frisk* (right) in OpenHowNet.

memes (Bloomfield, 1926; Goodenough, 1956; Lyons, 1968; Ullmann, 1973). For example, the most frequent sense of *husband* can be represented by a set of sememes: {human, spouse, family, male}. Some linguists believe that it is attainable to describe the word senses in any language through a finite set of sememes (Dong, 1988; Wierzbicka, 1996). To put this assumption into practice, Dong et al. (2010) developed a Sememe Knowledge Base (SKB), HowNet, which uses about 2,000 linguist-curated sememes to annotate over 100,000 English and Chinese words (Qi et al., 2022). In OpenHowNet (Qi et al., 2019), a publicly available version of HowNet, each word sense is structured hierarchically as a sememe tree (Ye et al., 2022), with the root node as the main sememe representing the core semantic category, as shown in Figure 1.

Intralexical component organization can reflect interlexical relations, as word senses within lexical relations often show systematic sememe differences (Wang et al., 2025). For instance, synonymy can be formalized through identical sememe combination (e.g., *husband* and *hubby*: {human, spouse, family, male}); hypernymy through subset containment (e.g., *husband* → *man*: {human, male}); antonymy through single-component negation while preserving others (e.g., *husband* → *wife*: {human, spouse, family, female}). From these analyses, such means of semantic compositionality hold promising prospects for LRC and LE.

Recently, researchers have explored the application of sememe knowledge in LRM scenarios. Wang et al. (2025) leveraged such knowledge to enhance Link Prediction for lexico-semantic knowledge graphs. While yielding notable improvements, this exploration, like other sememe-based applications (Hou et al., 2020; Lyu et al., 2021), utilizes sememe labels and ignores tree-structured information due to the complexity of integration. However, in such scenarios, tree-structured sememe information is crucial for more precise semantic descriptions, and neglecting it may impede the model from making correct predictions of lexical relations. For example, *searcher* and *frisk* in Figure 1 share the same sememe labels but differ significantly in semantics and lexical relations as well.

Meanwhile, there are practical deployment issues in integrating sememe information to benefit lexical relation mining. Not all words in the LRC and LE datasets have sememe annotations in HowNet, and it would be prohibitively labor-intensive and time-consuming to manually annotate sememes for out-of-vocabulary senses. While the introduction of Sememe Prediction (SP) (Xie et al., 2017) and Structured Sememe Prediction (SSP) (Ye et al., 2022) offers a promising solution to this issue, the effectiveness of predicted information in downstream tasks lacks sufficient validation.

Inspired by the potential value of structured sememe knowledge and the practical challenges in its integration into downstream LRM tasks, we are motivated to explore leveraging such sememe knowledge to enhance LRC and LE. We first propose an automated Sememe Tree Construction (STC) pipeline. It fully predicts sememe trees, covering the node, edge, and edge type information; Then, we design SememeLRM, a method that incorporates structured sememe knowledge into PLM-based models. Experimental results show that such sememe knowledge helps PLMs better understand nuanced lexical relations, consistently and significantly improving the performance across benchmarks for both tasks, even outperforming Large Language Model (LLM)-based baselines with 20 times more parameters on most benchmarks. Further results suggest that sememe trees automatically constructed by our pipeline can rival the gold-standard in HowNet, extending their applicability to lexico-semantic computing. We summarize the contributions as follows:

(1) We propose an automated STC pipeline, aiming to tackle the challenges of adopting structured sememe knowledge in annotation-scarce scenarios;

(2) We propose the SememeLRM method to fully leverage structured sememe knowledge for enhancing LRC and LE, achieving a notable 1.6% improvement on average across benchmarks;

(3) We present a potentially generalizable framework to leverage complete sememe trees in downstream tasks, helping to unlock the value of such intralexical knowledge in more NLP applications.

2 Related Works

2.1 Lexical Relation Mining

LRC methods have undergone three main developmental stages: **Pattern-based methods** (Hearst,

1992; Shwartz and Dagan, 2016; Roller et al., 2018) extract lexical relations in texts based on a set of syntactic patterns that could indicate such relations. Due to the ambiguity of natural language expressions, these methods fail to cover implicit relations via a closed set of patterns; **Embedding-based methods** (Shwartz and Dagan, 2016; Shwartz et al., 2016; Glavaš and Vulić, 2018; Wang et al., 2019) use static word embeddings as classification features, but these methods rely solely on the representation capacity of the embeddings and thus fail to capture and utilize dynamic contextual information to distinguish nuanced lexical relations; **PLM-based methods** (Jadhav et al., 2020; Wang et al., 2021; Pitarch et al., 2023) leverage contextualized word embeddings and lexico-semantic knowledge learned during pre-training to capture latent lexical relations. To further improve the performance, some of these methods (Ushio et al., 2021; Sun et al., 2025) adopt parameter-efficient fine-tuning (PEFT) to incorporate additional trainable parameters, or integrate supplementary lexico-semantic information, like lexical graph features.

Similar to LRC, LE methods can also be broadly categorized into embedding-based (Nguyen et al., 2017; Nickel and Kiela, 2017; Vulić and Mrkšić, 2018; Yang et al., 2022; Sato et al., 2022) and PLM-based (Pitarch et al., 2023; Moskvoretskii et al., 2024; Sun et al., 2025). Among embedding-based methods, some embed hierarchical structures in hyperbolic space, as it is well-suited for modeling hypernymy-hyponymy relations (Nickel and Kiela, 2017; Sato et al., 2022). For PLM-based methods, many of them (Pitarch et al., 2023; Sun et al., 2025) can also be adapted to tackle LRC by adjusting the training objective.

Currently, PLM-based methods serve as the SOTA for LRC and LE. However, they still struggle to distinguish nuanced semantic relations, such as synonymy versus antonymy (Pitarch et al., 2023) and synonymy versus hypernymy (Sun et al., 2025), making the integration of fine-grained semantic knowledge a worthwhile direction for exploration.

2.2 Sememe Knowledge Base: Application and Expansion

As a specialized lexical knowledge base, SKB describes word senses using fine-grained semantic units, sememes. HowNet (Dong et al., 2010), the most comprehensive SKB, has displayed its effectiveness in various NLP tasks, such as Language Modeling (Gu et al., 2018), Reverse Dictionary (Qi et al., 2020b), Text Matching (Lyu et al., 2021), and Word Sense Disambiguation (Hou et al., 2020; Zhang et al., 2022). However, almost all of these applications flatten sememe trees as semantic labels for word senses, ignoring the valuable tree-structured information (Qi et al., 2021), while it is crucial for more precise semantic descriptions.

Currently, the applications rely heavily on gold-standard sememe annotations in HowNet. However, it is impractical for HowNet to cover all word senses. To enable the automated construction and expansion of SKBs, SP (Xie et al., 2017) and SSP (Ye et al., 2022) have been proposed. The former aims to assign appropriate sememes to word senses from a predefined set, while the latter predicts sememes along with their hierarchical structures. For SP, MSGI (Qi et al., 2022) achieves SOTA performance by integrating diverse information into the model, including word sense definitions, multilingual synonyms, and visual information. Compared with the multilingual and multimodal information, textual definitions are more suitable for SP as they often align with sememe annotations and are more accessible in most application scenarios (Wang et al., 2025). Consequently, some definition-only methods (Li et al., 2018; Du et al., 2020; Wang et al., 2025) tend to be more applicable. For SSP, TaSTG (Ye et al., 2022) is a rare and valuable exploration. It incorporates sense definitions into a tree-attention Transformer (Vaswani et al., 2017) to generate sememe tree sequences. Evaluation results (Ye et al., 2022) show that joint learning of node and edge prediction increases models' burden, as evidenced by lower node accuracy than SP methods. In particular, improved node accuracy correlates with substantial gains in edge and overall tree prediction. This suggests cascading the two tasks, i.e., using more accurate SP-derived node prediction as the foundation of SSP, may enhance the overall performance of tree generation.

3 Automated Sememe Tree Construction

To adopt structured sememe knowledge in scenarios with limited annotations, we propose a Sememe Tree Construction (STC) pipeline that aims to build the sememe tree for a given word sense automatically. This task builds upon two prior tasks, SP (Xie et al., 2017) and SSP (Ye et al., 2022), while further requiring the relation prediction between sememes in the tree.

Given that existing SP methods outperform SSP

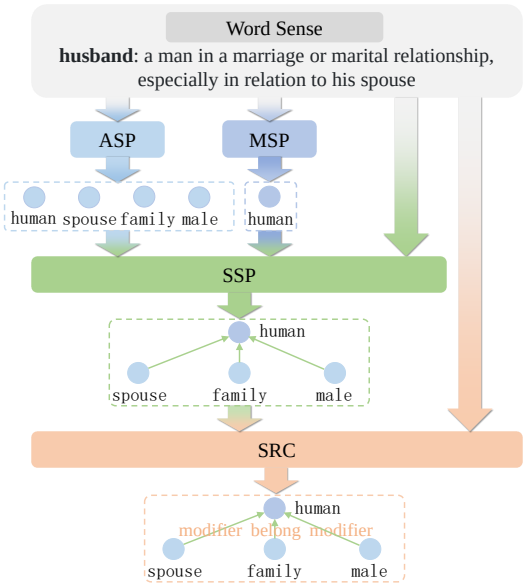

Figure 2: The proposed stage-wise pipeline for STC, from word senses to sememe trees with parent-child relations labeled.

methods in node prediction accuracy (Ye et al., 2022), we formulate STC as a composite task consisting of four stage-wise subtasks to maximize the quality of constructed sememe trees. The overall pipeline is depicted in Figure 2. Specifically, **All Sememe Prediction (ASP)** identifies all sememes of the word sense; **Main Sememe Prediction (MSP)** pinpoints its main sememe; **Structured Sememe Prediction (SSP)** generates a sememe tree for the word sense based on the predicted sememe set from ASP and the main sememe from MSP[2]; **Sememe Relation Classification (SRC)** identifies the relation type between parent-child sememe pairs in the tree generated by SSP.

Across all these subtasks, we take easily accessible textual definitions for word senses as input, which extends the method's applicability in real-world annotation-scarce scenarios. Moreover, dedicated methods are designed for each task:

For the first two upstream tasks, ASP and MSP, we follow Wang et al. (2025) and feed soft prompts (Hambardzumyan et al., 2021; Qin and Eisner, 2021) and sense definitions into a PLM-based encoder with corresponding classifiers;

For the subsequent SSP task, as LLMs have shown decent ability in understanding sememe tree structures (Shen et al., 2025) and achieved promising performance on various tree generation tasks, such as Syntactic Parsing (Tian et al., 2024) and Semantic Parsing (Liu et al., 2025b), we attempt to employ Qwen3-30B-A3B-Instruct (Yang et al., 2025) and DeepSeek-V3.2 (Liu et al., 2025a), two top-performing, cost-efficient open-source LLMs, for sememe tree sequence generation. We compare this LLM-based method with the SOTA, TaSTG (Ye et al., 2022), on the BabelSememe (Qi et al., 2020a) dataset. The evaluation is under the restricted setting where the ground-truth sememe set is provided. Evaluation details, including the prompt design, are provided in Appendix A. As shown in Table 1, LLMs with 3-shot learning (Dong et al., 2024) show superior structural organization capability and generate more accurate sememe trees. Based on the overall performance, we thus utilize DeepSeek-V3.2 to tackle SSP;

Finally, for the SRC task, we input soft prompts containing both a word sense definition and a parent-child sememe pair into a PLM-based encoder, which we then fine-tune for classification.

We train task-specific models for each subtask on the BabelSememe (Qi et al., 2020a) and SememeDef (Wang et al., 2025) datasets, with evaluation on the test set of BabelSememe. Our pipeline attains F1-scores of 60.3% and 68.9% on ASP and MSP, respectively, a Strict score of 52.3% on SSP, and an F1-score of 98.2% on SRC. More evaluation details are in Appendix A. Given the inherent challenge of this task, the results are sufficient to ensure constructing relatively reasonable sememe trees (Ye et al., 2022). Based on this pipeline, we conduct automated construction of sememe trees for WordNet (Miller, 1995), and evaluate their effectiveness in the subsequent LRM task.

Method	Strict	Edge
TaSTG	70.3	72.5
Qwen3-30B-A3B-Instruct	84.3	86.1
Deepseek-V3.2	88.0	89.7

Table 1: Restricted evaluation for the SSP task, where Strict assesses overall tree generation, and Edge evaluates the edge accuracy. Please refer to Ye et al. (2022) for detailed descriptions of the evaluation metrics.

4 Enhance LRM by Structured Sememe Knowledge

4.1 Task Formulation

LRC: We frame LRC as a multi-class classification task. Given a vocabulary V and a mutually exclusive, exhaustive set (Pitarch et al., 2023) of

[2]The main sememe predicted by MSP is added into the sememe set from ASP to ensure input consistency for SSP.

lexical relations $R = \{r_1, r_2, ..., r_k\}$, the task requires a function f_{LRC} that maps a pair of words $(w_1, w_2) \in V \times V$ to a k-dimensional probability distribution $\mathbf{p}$, where each element in the distribution denotes the probability of the respective relation. At prediction time, the most suitable relation for (w_1, w_2) is determined as the one with the highest probability in the output of f_{LRC}.

LE: We frame LE as a regression task, which requires a function f_{LE} that maps (w_1, w_2) to a continuous entailment score es in $[0, 1]$, denoting the degree to which w_1 is a type of w_2.

In this work, structured sememe information is utilized to tackle these tasks. Given a word $w \in \mathcal{V}$ with a predefined sense set $D_w = \{d_1, d_2, ..., d_n\}$, each sense $d_i \in D_w$ is defined by a sememe tree $T_{d_i} = (S_{d_i}, m_{d_i}, E_{d_i})$, where $S_{d_i} \subseteq S$ represents the set of sememes contained in T_{d_i}, with S denoting the global sememe inventory; $m_{d_i} \in S_{d_i}$ is the main sememe serving as the root node of T_{d_i}; E_{d_i} denotes the set of directed edges that capture the hierarchical relations between sememes in S_{d_i}. Each edge in E_{d_i} is formulated as (s_c, τ, s_p), where $s_p, s_c \in S_{d_i}$ are the parent and child sememes, respectively, and $\tau \in \Theta$ (Θ is the sememe-level relational type set) denotes the relation type between s_c and s_p. The set of sememe trees for w is denoted as $\mathcal{T}_w = \{T_{d_i}^w \mid d_i \in D_w\}$.

4.2 SememeLRM Method

We propose SememeLRM, a novel method that leverages structured sememe knowledge to improve the performance on two core subtasks of LRM: LRC and LE. The overall architecture of SememeLRM is illustrated in Figure 3. Specifically, for a word pair (w_1, w_2), SememeLRM employs a Relational Graph Neural Network (R-GNN) to aggregate tree-structured sememe embeddings into sense-level embeddings, which are subsequently fed into a PLM-based encoder to enhance relation prediction between (w_1, w_2).

For sememe representation, we first initialize a learnable embedding matrix for S, where each sememe $s \in S$ is assigned a dense vector $\mathbf{e}_s \in \mathbb{R}^l$.

To obtain the embedding for a word sense d, sememe embeddings $\{\mathbf{e}_s \mid s \in S_d\}$ and edge features E_d are fed into the R-GNN module. Then, the final hidden state of the root sememe m_d is extracted as the sense-level embedding $\mathbf{e}_d$ for d.

To incorporate the sense-level embeddings into the PLM-based encoder, we design a prompt template structured as: "$\mathbf{e}_{[\text{CLS}]}$, $\mathbf{e}_{t_1}$, ..., $\mathbf{e}_{t_o}$, $\mathbf{e}_1^{w_1}$, ..., $\mathbf{e}_{l_1}^{w_1}$, $\mathbf{e}_{d_1}^{w_1}$, ..., $\mathbf{e}_{d_{n_1}}^{w_1}$, $\mathbf{e}_{\text{and}}$, $\mathbf{e}_1^{w_2}$, ..., $\mathbf{e}_{l_2}^{w_2}$, $\mathbf{e}_{d_1}^{w_2}$, ..., $\mathbf{e}_{d_{n_2}}^{w_2}$, $\mathbf{e}_{[\text{SEP}]}$", where $\mathbf{e}_{t_1}$ to $\mathbf{e}_{t_o}$, and $\mathbf{e}_{\text{and}}$ are embeddings for the hard prompt tokens, like "*Today, I finally discovered the relation between ... and ...*"; $\mathbf{e}_1^{w_1}$ to $\mathbf{e}_{l_1}^{w_1}$ and $\mathbf{e}_1^{w_2}$ to $\mathbf{e}_{l_2}^{w_2}$ are embeddings for the tokens of w_1 and w_2, respectively. $\mathbf{e}_{d_1}^{w_1}$ to $\mathbf{e}_{d_{n_1}}^{w_1}$ and $\mathbf{e}_{d_1}^{w_2}$ to $\mathbf{e}_{d_{n_2}}^{w_2}$ are sememe-based sense embeddings[3] for all n_1 senses of w_1 and n_2 senses of w_2, respectively. Notably, except for the sememe-based sense embeddings, all the other embeddings in the template are obtained from the PLM's embedding layer.

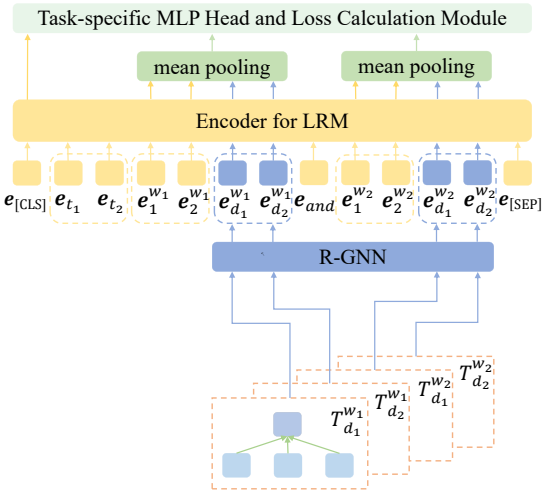

Figure 3: Illustration of the proposed SememeLRM. By introducing an R-GNN module to aggregate sememe embeddings into sense-level embeddings, SememeLRM incorporates structured sememe knowledge into the PLM-based encoder, enabling the generation of more effective relation representations for word pairs.

The final representations of w_1 and w_2, denoted as $\mathbf{h}_{w_1}$ and $\mathbf{h}_{w_2}$, are each obtained by mean pooling over the PLM's last hidden states corresponding to their respective token embeddings and sense embeddings. The final relation representation is defined as $\mathbf{h}_r = [\mathbf{h}_{[\text{CLS}]}; \mathbf{h}_{w_1}; \mathbf{h}_{w_2}; \mathbf{h}_{w_1} - \mathbf{h}_{w_2}]$, with $\mathbf{h}_{[\text{CLS}]}$ denoting the last hidden state of the [CLS] token. Then, $\mathbf{h}_r$ is fed into two independent MLPs to compute the LRC prediction distribution $\mathbf{p} = \text{softmax}(\text{MLP}_{\text{LRC}}(\mathbf{h}_r))$, and the LE score $es = \text{sigmoid}(\text{MLP}_{\text{LE}}(\mathbf{h}_r))$.

For LRC, our goal is to minimize the cross-entropy loss between the ground-truth distribution and $\mathbf{p}$:

$$\mathcal{L}_{\text{LRC}} = -\sum_i \mathbf{y}^*[i] \cdot \log(\mathbf{p}[i]), \quad (1)$$

where $\mathbf{y}^*$ is the ground-truth one-hot distribution.

[3]The dimension of sememe-based sense embeddings is aligned to that of the PLM embeddings.

Method	BLESS	K&H+N	EVALution	CogALexV	ROOT09
LexNET	89.3	98.5	60.0	44.5	81.3
SphereRE	93.8	99.0	62.0	47.1	86.1
KEML	94.4	**99.3**	66.0	50.0	87.8
RelBERT	92.1	94.9	70.1	66.4	91.0
NCGC	95.6	98.9	77.1	76.2	93.7
DeBERTa-large (ours)	95.4	98.8	78.7	76.4	94.2
DeBERTa-large†	95.4	98.9	78.6	76.1	94.3
DeBERTa-large + GET†	95.8	98.7	78.4	74.4	94.0
DeBERTa-xlarge†	95.7	98.7	78.4	76.1	94.5
DeBERTa-xlarge + GET†	95.9	98.8	<u>80.5</u>	76.5	**95.4**
LLaMA3-8B†	95.3	98.9	77.2	<u>79.0</u>	94.7
LLaMA3-8B-Instruct†	96.3	99.1	75.6	77.7	94.5
SememeLRM	**96.5** (+1.2%)	99.1 (+0.3%)	**81.6** (+3.7%)	78.9 (+3.3%)	<u>95.3</u> (+1.2%)
SememeLRM w/ automated STC	<u>96.4</u> (+1.0%)	<u>99.2</u> (+0.4%)	<u>80.5</u> (+2.3%)	**79.4** (+3.9%)	94.5 (+0.3%)

Table 2: Weighted F1-scores (%) on LRC Benchmarks, where † denotes the results from Sun et al. (2025), while other baseline results are mainly from Pitarch et al. (2023). Baselines are grouped into three categories by parameter scale relative to SememeLRM: smaller, comparable, and larger. The overall best results are shown in **bold**, with the second-best results <u>underscored</u>. For SememeLRM, the percentage improvements over DeBERTa-large (ours) are shown in parentheses. Following Santus et al. (2016a), random relation results in CogALexV are excluded from test set performance reporting.

For LE, our goal is to minimize the mean squared error (MSE) loss:

$$\mathcal{L}_{\text{LE}} = (es - es^*)^2, \tag{2}$$

where $es^* \in [0, 1]$ is the ground-truth score.

5 Experiments

5.1 Experimental Settings

Datasets: Five widely used datasets are adopted for the LRC evaluation, including BLESS (Baroni and Lenci, 2011), K&H+N (Necşulescu et al., 2015), EVALution (Santus et al., 2015), CogALexV (Santus et al., 2016a), ROOT9 (Santus et al., 2016b), covering 10 types of lexical relations. We use HyperLex (Vulić et al., 2017) to evaluate the graded LE task. It has two different data splits based on lexical and random, respectively. We adopt the original train/validation/test splits of these datasets. Further details and statistical information about them are provided in Appendix B.

Baselines: For the LRC task, we compare SememeLRM against a variety of top-performing methods, including LexNET (Shwartz and Dagan, 2016), SphereRE (Wang et al., 2019), KEML (Wang et al., 2021), RelBERT (Ushio et al., 2021), NCGC (Pitarch et al., 2023), and GET (Sun et al., 2025). For the LE task, LEAR (Vulić and Mrkšić, 2018), HF (Yang et al., 2022), NCGC (Pitarch et al., 2023), TaxoLLaMA (Moskvoretskii et al., 2024), and GET (Sun et al., 2025) are adopted as baselines. Additionally, for both tasks, we compare against DeBERTa (He et al., 2021) and LLaMA3 (Grattafiori et al., 2024) baselines, which are fine-tuned by Sun et al. (2025) via an NCGC-like method.

Experimental Configuration: Considering the parameter scales of PLMs adopted by top-performing baselines (Sun et al., 2025), we select DeBERTa-large (He et al., 2021) as the base model for a fair comparison. It features 24 layers with a hidden size of 1024. Also, we employ R-GCN (Schlichtkrull et al., 2018) as the R-GNN module. Further experimental details are provided in Appendix C.

5.2 Overall Results

We separately evaluate the performance of SememeLRM when leveraging sememe data from HowNet and our automated STC pipeline. The overall test results on LRC and LE are shown in Table 2 and Table 3, respectively. From them, we have the following observations:

(1) From the overall performance, SememeLRM achieves the best weighted F1-scores on 2 out of 5 datasets for LRC and the HyperLex dataset for LE. It yields consistent improvements over DeBERTa-large across all benchmarks, with an average gain of 1.6%, and notably larger gains on the more challenging datasets, namely EVALution and CogALexV. This demonstrates that the incorporation of fine-grained structured sememe information provides PLMs with complementary lexico-semantic

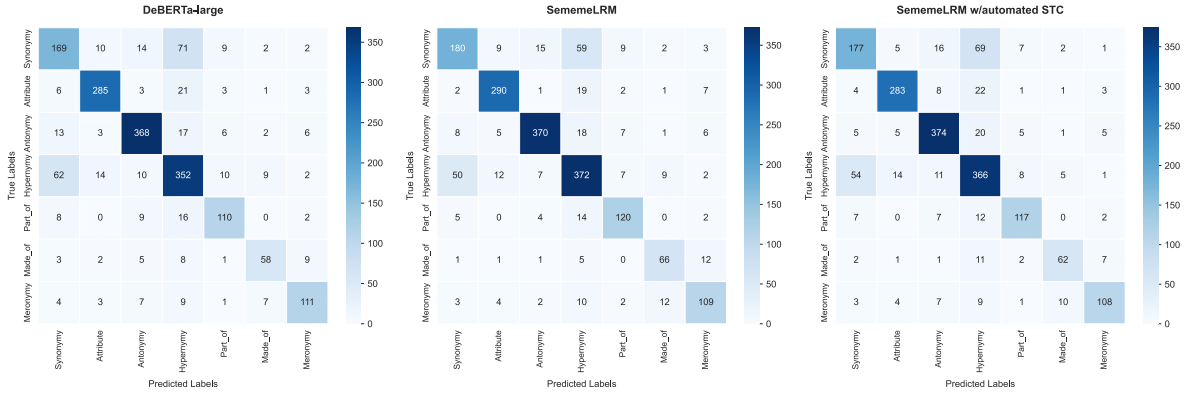

Figure 4: Confusion matrix comparison of SememeLRM and DeBERTa-large (ours) on EVALution dataset.

Method	Lexical	Random
LEAR	17.4	68.6
HF	-	69.0
NCGC	79.4	82.8
DeBERTa-large (ours)	88.0	90.2
DeBERTa-xlarge†	88.1	89.8
DeBERTa-xlarge + GET†	88.7	90.1
TaxoLLaMA	70.2	59.3
LLaMA3-8B†	87.3	90.5
SememeLRM	**88.8** (+0.9%)	**90.7** (+0.6%)
w/ automated STC	88.6 (+0.7%)	90.5 (+0.3%)
IAA	86.4	

Table 3: The Spearman ρ correlation on HyperLex. The IAA represents the inter-annotator agreement when assigning LE ratings. The notations follow Table 2.

knowledge, enabling more accurate prediction;

(2) From the parameter-efficiency perspective, just adding an R-GCN module (47M), SememeLRM, based on DeBERTa-large (350M), achieves performance comparable to or even superior to methods based on DeBERTa-xlarge (750M) and LLaMA3 (8B). This indicates that introducing structured sememe knowledge, instead of scaling up PLM size, facilitates more efficient and effective mining of lexical relations;

(3) Regarding the sources of sememe knowledge, SememeLRM, utilizing sememe information predicted by our STC pipeline, also attains a notable average improvement of 1.3% across the benchmarks. It achieves comparable or even better performance than the gold-standard sememe data from HowNet on four out of six benchmarks. On the remaining two, it still outperforms the baseline without sememe information by a clear margin. This extends the applicability to lexico-semantic computing in annotation-scarce scenarios.

6 Analysis

6.1 Analysis on How Structured Sememe Knowledge Benefits LRM

To better understand how structured sememe knowledge enhances LRC and LE, we compare the confusion matrices of SememeLRM and DeBERTa-large on EVALution. As shown in Figure 4, SememeLRM substantially reduces confusion between synonymy and hypernymy, a known challenge highlighted by Sun et al. (2025), while improving or maintaining the performance on others.

The improvement can be largely attributed to the structured organization of intralexical sememes, which reflects interlexical relations: synonyms usually share nearly identical sememe trees, whereas in a hypernym-hyponym pair, the hyponym's sememes generally subsume those of the hypernym, or their main sememes are also hierarchically related. In contrast, distributional word embeddings may struggle to represent such systematic differences, as synonyms, hypernyms, and hyponyms tend to appear in similar contexts, resulting in similar embedding representations that hinder PLMs from distinguishing these relational distinctions.

To substantiate this, we first verify whether systematic sememe differences are indeed reflected in benchmarks for LRM. To this end, we manually summarize three patterns corresponding to synonymy, co-hyponymy, and antonymy: (a) The two words share a sense pair with identical sememe trees; (b) The two words share a sense pair with the same main sememe, and neither sememe combination is a subset of the other; (c) The two words share a sense pair with only one pair of sememes within the antonymy relation, while all other sememes are identical. As shown in Table 4, word pairs within the target relation exhibit significantly

Pattern	Relation	Proportion
(a)	Synonymy	51.2
	Non-synonymy	12.4
(b)	Co-hyponymy	49.9
	Non-co-hyponymy	11.1
(c)	Antonymy	17.8
	Non-antonymy	0.1

Table 4: Proportions(%) of samples satisfying each sememe pattern for target versus non-target relations.

Pattern	Relation	Correct	Error
(a)	Synonymy	56.8	37.6
(b)	Co-hyponymy	50.0	30.4
(c)	Antonymy	19.5	2.9

Table 5: Proportions (%) of pattern-matching samples in the Correct and Error groups of SememeLRM.

higher proportions matching the corresponding patterns than non-target pairs, validating the existence of such systematic compositional differences in practical LRM datasets.

We further verify whether the model can learn and leverage these systematic differences for prediction. Specifically, we split the test set into Correct/Error groups based on predictions from SememeLRM, and compute the proportion of samples matching the corresponding patterns in each group. As shown in Table 5, the Correct group consistently exhibits higher percentages than the Error group, indicating that the model indeed captures, learns, and leverages these patterns for prediction. Although the patterns explored here are relatively simple, it is foreseeable that the model can also learn more complex difference patterns for more accurate prediction. These results demonstrate that structured sememe knowledge exactly provides complementary compositional signals that help address the shortcomings of distributional representations, thereby improving the performance.

However, SememeLRM still makes certain erroneous predictions. To address this, we conduct a detailed error analysis in Appendix F, revealing that the errors primarily stem from three sources, including taxonomy inconsistencies across resources, the lack of relevant senses in HowNet, and the polysemousness of words.

6.2 Analysis on Why STC-Predicted Sememe Trees Can Rival Gold-Standard

As discussed in Section 5.2, the STC pipeline yields performance comparable to gold-standard data from HowNet. To further explore why, we first quantify how well the STC-based sememe trees cover the gold-standard ones. Specifically, for each instance in the benchmarks, three coverage metrics are considered: main sememe node coverage, sememe node coverage, and (*parent*, *rel*, *child*) triple coverage. The average coverage rates achieved are 70.7%, 73.9%, and 65.9%, respectively. This suggests that the predicted sememe trees capture most of the node-level and structural information in the gold-standard trees, thereby yielding similar effectiveness as mentioned above. Moreover, as a manually constructed SKB, HowNet inevitably contains annotation inconsistencies, and its gold-standard trees should not be treated as the sole correct answer. The competitive performance of STC-predicted trees, despite not fully matching the gold standard, also suggests that alternative yet linguistically plausible sememe annotations can be equally effective for downstream tasks.

In addition to the quantitative analysis, we also conduct a case study. We find that some STC-based sememe trees, although not fully accurate, still aid the model in predicting the correct lexical relations. For instance, for the nouns *defeat* and *success*, the predicted sememes for *defeat* (`{result, fail}`) are consistent with the gold-standard, while the predicted sememes for *success* (`{circumstances, succeed, human}`) differ from the gold-standard (`result, succeed}`). Despite such differences, the model can still recognize their antonymy relation via the opposition between `fail` and `succeed`.

Beyond this, the case study also reveals that the automated STC pipeline helps alleviate the problem of missing word senses in HowNet. As shown in Table 15, HowNet lacks a sense like "*protective covering on top of a motor vehicle*" for *roof*. This deficiency prevents the model from correctly identifying the meronymy relation between *car* and *roof*. By automatically constructing sememe trees for *roof* based on glosses from WordNet, a new sense is supplemented with the sememe combination `{part, land_vehicle, head}`, which helps the model correctly identify the meronymy relation. This finding also indirectly validates the value of our automated STC pipeline in expanding and supplementing SKBs.

6.3 Analysis on the Effectiveness and Efficiency of Different Sense Embeddings

To further illustrate the advantages of sememe-based sense embeddings for LRC and LE, we

Method	LRC	LE
DeBERTa-large (Ours)	88.7	89.1
Gloss	88.8 (+0.1%)	89.4 (+0.3%)
Sememe (SememeLRM)	90.3 (+1.8%)	89.8 (+0.8%)

Table 6: Average performance comparison on LRC and LE between gloss-based and sememe-based representation for word senses.

compare SememeLRM against a DeBERTa-large model incorporating gloss information from WordNet (Miller, 1995). Experimental results are concisely summarized in Table 6, with full details in Appendix D. These results demonstrate that using fine-grained tree-structured semantic units outperforms textual sense descriptions in representing word senses and lexical relations. This superiority stems from the unique definition paradigm of sememes: on the one hand, sememe differences across words systematically reflect lexical relations; on the other hand, the finiteness of sememe labels facilitates generalization of such differences. These intrinsic properties help reduce the complexity of model learning and enhance robustness in nuanced lexical relation identification.

Additionally, our statistics show that sememe-based sense embeddings are more efficient: per data sample, only 13.3 sememe tokens are input to the R-GNN and 4.3 sense-related tokens to the PLM, in stark contrast to the 95.4 tokens required by gloss-based methods for PLM input.

6.4 Ablation Analysis

Sememe Tree Components: We analyze the contribution of node, edge, and edge type information in sememe trees. Table 7 shows that the hierarchical structure of sememes contributes the most to the performance gains. Compared with only using sememe labels, the integration of structured information enables more precise descriptions of word senses and better reflects semantic differences between them, leading to superior performance. Built upon this, the incorporation of edge type information further improves the performance.

R-GNN Module Variants: We also investigate the impact of different R-GNN variants on the performance. Table 7 shows that both R-GAT (Busbridge et al., 2019)-based and R-GCN-based methods yield consistent performance gains. This suggests that using R-GNNs is a reliable approach for modeling tree-structured sememe knowledge, and exploring more effective R-GNNs to leverage this knowledge is a worthwhile research direction.

Method	LRC	LE
sememe set	89.2	89.5
GCN + sememe tree	89.9	89.9
R-GAT + sememe tree	90.1	**90.1**
R-GCN + sememe tree (SememeLRM)	**90.3**	89.8
R-GCN + sememe tree + all outputs fed	90.1	89.6

Table 7: Average performance comparison for ablation analysis, with details provided in Appendix D. GCN (Kipf and Welling, 2017) is used to ignore the edge type information.

Embedding Input Strategies: We compare two sense embedding input strategies for the PLM: feeding all R-GNN outputs versus the outputs for main sememes. As shown in Table 7, the two methods yield comparable results, but the latter achieves higher efficiency with fewer input tokens. This also indicates that effective modeling of structural information is the key to the improvements on LRC and LE, rather than explicitly highlighting core semantic categories in sense representations.

7 Conclusion

In this paper, we explore the utilization of structured sememe knowledge to enhance LRM. We first present an automated STC pipeline to predict sememe trees; Then, we propose the SememeLRM method to fully leverage structured sememe knowledge for the representation learning of word senses and lexical relations; Experimental results show that it achieves a notable 1.6% improvement on average across benchmarks, even outperforming LLM-based methods with 20 times more parameters on most benchmarks. Further results suggest that sememe trees predicted by our pipeline can rival the gold-standard in HowNet, extending their applicability to lexico-semantic computing. An in-depth analysis also demonstrates that structured sememe knowledge yields more substantial performance gains than textual definitions, further revealing the effectiveness and efficiency of SememeLRM. Overall, this work makes comprehensive and significant progress for leveraging complete sememe trees in downstream tasks, helping unlock the value of such expert knowledge.

In the near future, we would explore more refined strategies to exploit structured sememe knowledge and extend the benefits of our framework to a broader range of lexico-semantic tasks.

Limitations

Our work is subject to several limitations:

(1) This work primarily focuses on the English language, where HowNet provides a substantial amount of sememe annotations for English lexicons, and large-scale datasets are available to support training of our automated STC pipeline. For low-resource languages, however, the scarcity of such resources may weaken the effectiveness of SememeLRM;

(2) While this work performs an in-depth exploration of sememe tree utilization for the LRM task, the original fine-grained edge type information is simplified by grouping them into a small set of general categories (detailed in Appendix A). This choice not only reduces the complexity in understanding sememe trees but also prioritizes the parameter efficiency of downstream models. Nevertheless, how to leverage such fine-grained information more effectively and efficiently remains an open question, deserving further exploration in future work;

(3) With regard to the efficiency of training, SememeLRM utilizes an additional R-GNN module to obtain structured sememe features. This incurs higher memory consumption and longer training times than the baseline methods. We also conduct a cost-benefit analysis in Appendix E to discuss this.

Acknowledgements

This paper is supported by the National Natural Science Foundation of China (No. 62036001).

References

Marco Baroni and Alessandro Lenci. 2011. How we BLESSed distributional semantic evaluation. In *Proceedings of the GEMS 2011 Workshop on GEometrical Models of Natural Language Semantics*, pages 1–10, Edinburgh, UK. Association for Computational Linguistics.

Leonard Bloomfield. 1926. A set of postulates for the science of language. *International Journal of American Linguistics*, 15:195 – 202.

Rajesh Bordawekar and Oded Shmueli. 2017. Using word embedding to enable semantic queries in relational databases. In *Proceedings of the 1st workshop on data management for end-to-end machine learning*, pages 1–4.

Zied Bouraoui and Steven Schockaert. 2019. Automated rule base completion as bayesian concept induction. In *The Thirty-Third AAAI Conference on Artificial Intelligence, AAAI 2019, The Thirty-First Innovative Applications of Artificial Intelligence Conference, IAAI 2019, The Ninth AAAI Symposium on Educational Advances in Artificial Intelligence, EAAI 2019, Honolulu, Hawaii, USA, January 27 - February 1, 2019*, pages 6228–6235. AAAI Press.

Dan Busbridge, Dane Sherburn, Pietro Cavallo, and Nils Y. Hammerla. 2019. Relational graph attention networks. *CoRR*, abs/1904.05811.

Qingxiu Dong, Lei Li, Damai Dai, Ce Zheng, Jingyuan Ma, Rui Li, Heming Xia, Jingjing Xu, Zhiyong Wu, Baobao Chang, Xu Sun, Lei Li, and Zhifang Sui. 2024. A survey on in-context learning. In *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing*, pages 1107–1128, Miami, Florida, USA. Association for Computational Linguistics.

Zhendong Dong. 1988. Knowledge description: what, how and who. In *Proceedings of International Symposium on Electronic Dictionary*, volume 18.

Zhendong Dong, Qiang Dong, and Changling Hao. 2010. HowNet and its computation of meaning. In *Coling 2010: Demonstrations*, pages 53–56, Beijing, China. Coling 2010 Organizing Committee.

Jiaju Du, Fanchao Qi, Maosong Sun, and Zhiyuan Liu. 2020. Lexical sememe prediction using dictionary definitions by capturing local semantic correspondence. *CoRR*, abs/2001.05954.

Gregory Finley, Stephanie Farmer, and Serguei Pakhomov. 2017. What analogies reveal about word vectors and their compositionality. In *Proceedings of the 6th Joint Conference on Lexical and Computational Semantics (*SEM 2017)*, pages 1–11, Vancouver, Canada. Association for Computational Linguistics.

Goran Glavaš and Ivan Vulić. 2018. Discriminating between lexico-semantic relations with the specialization tensor model. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 2 (Short Papers)*, pages 181–187, New Orleans, Louisiana. Association for Computational Linguistics.

Ward H Goodenough. 1956. Componential analysis and the study of meaning. *Language*, 32(1):195–216.

Aaron Grattafiori, Abhimanyu Dubey, Abhinav Jauhri, Abhinav Pandey, Abhishek Kadian, Ahmad Al-Dahle, Aiesha Letman, Akhil Mathur, Alan Schelten, and 1 others. 2024. The llama 3 herd of models. *arXiv preprint arXiv:2407.21783*.

Yihong Gu, Jun Yan, Hao Zhu, Zhiyuan Liu, Ruobing Xie, Maosong Sun, Fen Lin, and Leyu Lin. 2018. Language modeling with sparse product of sememe experts. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 4642–4651, Brussels, Belgium. Association for Computational Linguistics.

Karen Hambardzumyan, Hrant Khachatrian, and Jonathan May. 2021. WARP: Word-level Adversarial ReProgramming. In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, pages 4921–4933, Online. Association for Computational Linguistics.

Pengcheng He, Xiaodong Liu, Jianfeng Gao, and Weizhu Chen. 2021. Deberta: decoding-enhanced bert with disentangled attention. In *9th International Conference on Learning Representations, ICLR 2021, Virtual Event, Austria, May 3-7, 2021*. OpenReview.net.

Marti A. Hearst. 1992. Automatic acquisition of hyponyms from large text corpora. In *Proceedings of the 14th Conference on Computational Linguistics - Volume 2*, COLING '92, page 539–545, USA. Association for Computational Linguistics.

Bairu Hou, Fanchao Qi, Yuan Zang, Xurui Zhang, Zhiyuan Liu, and Maosong Sun. 2020. Try to substitute: An unsupervised Chinese word sense disambiguation method based on HowNet. In *Proceedings of the 28th International Conference on Computational Linguistics*, pages 1752–1757, Barcelona, Spain (Online). International Committee on Computational Linguistics.

Aishwarya Jadhav, Yifat Amir, and Zachary Pardos. 2020. Lexical relation mining in neural word embeddings. In *Proceedings of the 28th International Conference on Computational Linguistics*, pages 1299–1311, Barcelona, Spain (Online). International Committee on Computational Linguistics.

Thomas N. Kipf and Max Welling. 2017. Semi-supervised classification with graph convolutional networks. In *5th International Conference on Learning Representations, ICLR 2017, Toulon, France, April 24-26, 2017, Conference Track Proceedings*. OpenReview.net.

Zornitsa Kozareva and Eduard Hovy. 2010. A semi-supervised method to learn and construct taxonomies using the web. In *Proceedings of the 2010 Conference on Empirical Methods in Natural Language Processing*, pages 1110–1118, Cambridge, MA. Association for Computational Linguistics.

Peng-Hsuan Li, Tsan-Yu Yang, and Wei-Yun Ma. 2020. CA-EHN: Commonsense analogy from E-HowNet. In *Proceedings of the Twelfth Language Resources and Evaluation Conference*, pages 2984–2990, Marseille, France. European Language Resources Association.

Wei Li, Xuancheng Ren, Damai Dai, Yunfang Wu, Houfeng Wang, and Xu Sun. 2018. Sememe prediction: Learning semantic knowledge from unstructured textual wiki descriptions. *CoRR*, abs/1808.05437.

Aixin Liu, Aoxue Mei, Bangcai Lin, Bing Xue, Bingxuan Wang, Bingzheng Xu, Bochao Wu, Bowei Zhang, Chaofan Lin, Chen Dong, and 1 others. 2025a. Deepseek-v3.2: Pushing the frontier of open large language models. *arXiv preprint arXiv:2512.02556*.

Ruiheng Liu, Jinyu Zhang, Yanqi Song, Yu Zhang, and Bailong Yang. 2025b. Filling memory gaps: Enhancing continual semantic parsing via SQL syntax variance-guided llms without real data replay. In *AAAI-25, Sponsored by the Association for the Advancement of Artificial Intelligence, February 25 - March 4, 2025, Philadelphia, PA, USA*, pages 24641–24649. AAAI Press.

Ilya Loshchilov and Frank Hutter. 2017. Fixing weight decay regularization in adam. *CoRR*, abs/1711.05101.

John Lyons. 1968. *Introduction to theoretical linguistics*, volume 510. Cambridge university press.

Boer Lyu, Lu Chen, Su Zhu, and Kai Yu. 2021. LET: linguistic knowledge enhanced graph transformer for chinese short text matching. In *Thirty-Fifth AAAI Conference on Artificial Intelligence, AAAI 2021, Thirty-Third Conference on Innovative Applications of Artificial Intelligence, IAAI 2021, The Eleventh Symposium on Educational Advances in Artificial Intelligence, EAAI 2021, Virtual Event, February 2-9, 2021*, pages 13498–13506. AAAI Press.

Ken McRae, George S Cree, Mark S Seidenberg, and Chris McNorgan. 2005. Semantic feature production norms for a large set of living and nonliving things. *Behavior research methods*, 37(4):547–559.

George A Miller. 1995. Wordnet: a lexical database for english. *Communications of the ACM*, 38(11):39–41.

Viktor Moskvoretskii, Ekaterina Neminova, Alina Lobanova, Alexander Panchenko, and Irina Nikishina. 2024. TaxoLLaMA: WordNet-based model for solving multiple lexical semantic tasks. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 2331–2350, Bangkok, Thailand. Association for Computational Linguistics.

Silvia Necşulescu, Sara Mendes, David Jurgens, Núria Bel, and Roberto Navigli. 2015. Reading between the lines: Overcoming data sparsity for accurate classification of lexical relationships. In *Proceedings of the Fourth Joint Conference on Lexical and Computational Semantics*, pages 182–192, Denver, Colorado. Association for Computational Linguistics.

Kim Anh Nguyen, Maximilian Köper, Sabine Schulte im Walde, and Ngoc Thang Vu. 2017. Hierarchical embeddings for hypernymy detection and directionality. In *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing*, pages 233–243, Copenhagen, Denmark. Association for Computational Linguistics.

Maximillian Nickel and Douwe Kiela. 2017. Poincaré embeddings for learning hierarchical representations. *Advances in neural information processing systems*, 30.

Lucia Pitarch, Jordi Bernad, Lacramioara Dranca, Carlos Bobed Lisbona, and Jorge Gracia. 2023. No clues good clues: out of context lexical relation classification. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 5607–5625, Toronto, Canada. Association for Computational Linguistics.

Fanchao Qi, Liang Chang, Maosong Sun, Sicong Ouyang, and Zhiyuan Liu. 2020a. Towards building a multilingual sememe knowledge base: Predicting sememes for babelnet synsets. *Proceedings of the AAAI Conference on Artificial Intelligence*, 34(05):8624–8631.

Fanchao Qi, Chuancheng Lv, Zhiyuan Liu, Xiaojun Meng, Maosong Sun, and Hai-Tao Zheng. 2022. Sememe prediction for BabelNet synsets using multilingual and multimodal information. In *Findings of the Association for Computational Linguistics: ACL 2022*, pages 158–168, Dublin, Ireland. Association for Computational Linguistics.

Fanchao Qi, Ruobing Xie, Yuan Zang, Zhiyuan Liu, and Maosong Sun. 2021. Sememe knowledge computation: a review of recent advances in application and expansion of sememe knowledge bases. *Frontiers of Computer Science*, 15(5):155327.

Fanchao Qi, Chenghao Yang, Zhiyuan Liu, Qiang Dong, Maosong Sun, and Zhendong Dong. 2019. Openhownet: An open sememe-based lexical knowledge base. *CoRR*, abs/1901.09957.

Fanchao Qi, Lei Zhang, Yanhui Yang, Zhiyuan Liu, and Maosong Sun. 2020b. WantWords: An open-source online reverse dictionary system. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing: System Demonstrations*, pages 175–181, Online. Association for Computational Linguistics.

Guanghui Qin and Jason Eisner. 2021. Learning how to ask: Querying LMs with mixtures of soft prompts. In *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 5203–5212, Online. Association for Computational Linguistics.

Stephen Roller, Douwe Kiela, and Maximilian Nickel. 2018. Hearst patterns revisited: Automatic hypernym detection from large text corpora. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, pages 358–363, Melbourne, Australia. Association for Computational Linguistics.

Enrico Santus, Anna Gladkova, Stefan Evert, and Alessandro Lenci. 2016a. The CogALex-V shared task on the corpus-based identification of semantic relations. In *Proceedings of the 5th Workshop on Cognitive Aspects of the Lexicon (CogALex - V)*, pages 69–79, Osaka, Japan. The COLING 2016 Organizing Committee.

Enrico Santus, Alessandro Lenci, Tin-Shing Chiu, Qin Lu, and Chu-Ren Huang. 2016b. Nine features in a random forest to learn taxonomical semantic relations. In *Proceedings of the Tenth International Conference on Language Resources and Evaluation (LREC'16)*, pages 4557–4564, Portorož, Slovenia. European Language Resources Association (ELRA).

Enrico Santus, Frances Yung, Alessandro Lenci, and Chu-Ren Huang. 2015. EVALution 1.0: an evolving semantic dataset for training and evaluation of distributional semantic models. In *Proceedings of the 4th Workshop on Linked Data in Linguistics: Resources and Applications*, pages 64–69, Beijing, China. Association for Computational Linguistics.

Naomi Sato, Masaru Isonuma, Kimitaka Asatani, Shoya Ishizuka, Aori Shimizu, and Ichiro Sakata. 2022. Lexical entailment with hierarchy representations by deep metric learning. In *Findings of the Association for Computational Linguistics: EMNLP 2022*, pages 3517–3522, Abu Dhabi, United Arab Emirates. Association for Computational Linguistics.

Michael Schlichtkrull, Thomas N Kipf, Peter Bloem, Rianne Van Den Berg, Ivan Titov, and Max Welling. 2018. Modeling relational data with graph convolutional networks. In *European semantic web conference*, pages 593–607. Springer.

Sijia Shen, Feiyan Jiang, Peiyan Wang, Yubo Feng, Yuchen Jiang, and Chang Liu. 2025. Do LLMs know and understand domain conceptual knowledge? In *Findings of the Association for Computational Linguistics: EMNLP 2025*, pages 5967–5976, Suzhou, China. Association for Computational Linguistics.

Vered Shwartz and Ido Dagan. 2016. Path-based vs. distributional information in recognizing lexical semantic relations. In *Proceedings of the 5th Workshop on Cognitive Aspects of the Lexicon (CogALex - V)*, pages 24–29, Osaka, Japan. The COLING 2016 Organizing Committee.

Vered Shwartz, Yoav Goldberg, and Ido Dagan. 2016. Improving hypernymy detection with an integrated path-based and distributional method. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 2389–2398, Berlin, Germany. Association for Computational Linguistics.

C. Spearman. 1904. The proof and measurement of association between two things. *The American Journal of Psychology*, 15(1):72–101.

Robyn Speer and Catherine Havasi. 2012. Representing general relational knowledge in ConceptNet 5. In *Proceedings of the Eighth International Conference on Language Resources and Evaluation (LREC'12)*,

pages 3679–3686, Istanbul, Turkey. European Language Resources Association (ELRA).

Jingwen Sun, Zhiyi Tian, Yu He, Jingwei Sun, and Guangzhong Sun. 2025. Introducing graph context into language models through parameter-efficient fine-tuning for lexical relation mining. In *Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 10359–10374, Vienna, Austria. Association for Computational Linguistics.

Yuanhe Tian, Fei Xia, and Yan Song. 2024. Large language models are no longer shallow parsers. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 7131–7142, Bangkok, Thailand. Association for Computational Linguistics.

Stephen Ullmann. 1973. *Meaning and Style: Collected Papers*. Barnes & Noble Books.

Asahi Ushio, Jose Camacho-Collados, and Steven Schockaert. 2021. Distilling relation embeddings from pretrained language models. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pages 9044–9062, Online and Punta Cana, Dominican Republic. Association for Computational Linguistics.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, Lukasz Kaiser, and Illia Polosukhin. 2017. Attention is all you need. In *Advances in Neural Information Processing Systems 30: Annual Conference on Neural Information Processing Systems 2017, December 4-9, 2017, Long Beach, CA, USA*, pages 5998–6008.

Ivan Vulić, Daniela Gerz, Douwe Kiela, Felix Hill, and Anna Korhonen. 2017. HyperLex: A large-scale evaluation of graded lexical entailment. *Computational Linguistics*, 43(4):781–835.

Ivan Vulić and Nikola Mrkšić. 2018. Specialising word vectors for lexical entailment. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long Papers)*, pages 1134–1145, New Orleans, Louisiana. Association for Computational Linguistics.

Chengyu Wang, Xiaofeng He, and Aoying Zhou. 2019. SphereRE: Distinguishing lexical relations with hyperspherical relation embeddings. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 1727–1737, Florence, Italy. Association for Computational Linguistics.

Chengyu Wang, Minghui Qiu, Jun Huang, and Xiaofeng He. 2021. KEML: A knowledge-enriched meta-learning framework for lexical relation classification. In *Thirty-Fifth AAAI Conference on Artificial Intelligence, AAAI 2021, Thirty-Third Conference on Innovative Applications of Artificial Intelligence, IAAI 2021, The Eleventh Symposium on Educational Advances in Artificial Intelligence, EAAI 2021, Virtual Event, February 2-9, 2021*, pages 13924–13932. AAAI Press.

Hansi Wang, Yue Wang, Qiliang Liang, and Yang Liu. 2025. How sememic components can benefit link prediction for lexico-semantic knowledge graphs? In *Proceedings of the 2025 Conference on Empirical Methods in Natural Language Processing*, pages 14665–14684, Suzhou, China. Association for Computational Linguistics.

Anna Wierzbicka. 1996. *Semantics: Primes and Universals*. Oxford University Press.

Rong Xiang, Emmanuele Chersoni, Qin Lu, Chu-Ren Huang, Wenjie Li, and Yunfei Long. 2021. Lexical data augmentation for sentiment analysis. *Journal of the Association for Information Science and Technology*, 72(11):1432–1447.

Ruobing Xie, Xingchi Yuan, Zhiyuan Liu, and Maosong Sun. 2017. Lexical sememe prediction via word embeddings and matrix factorization. In *Proceedings of the 26th International Joint Conference on Artificial Intelligence*, IJCAI'17, page 4200–4206. AAAI Press.

An Yang, Anfeng Li, Baosong Yang, Beichen Zhang, Binyuan Hui, Bo Zheng, Bowen Yu, Chang Gao, Chengen Huang, Chenxu Lv, and 1 others. 2025. Qwen3 technical report. *arXiv preprint arXiv:2505.09388*.

Dongqiang Yang, Ning Li, Li Zou, and Hongwei Ma. 2022. Lexical semantics enhanced neural word embeddings. *Knowledge-Based Systems*, 252:109298.

Yining Ye, Fanchao Qi, Zhiyuan Liu, and Maosong Sun. 2022. Going "deeper": Structured sememe prediction via transformer with tree attention. In *Findings of the Association for Computational Linguistics: ACL 2022*, pages 128–138, Dublin, Ireland. Association for Computational Linguistics.

Xiang Zhang, Bradley Hauer, and Grzegorz Kondrak. 2022. Improving HowNet-based Chinese word sense disambiguation with translations. In *Findings of the Association for Computational Linguistics: EMNLP 2022*, pages 4530–4536, Abu Dhabi, United Arab Emirates. Association for Computational Linguistics.

A Evaluation Details for SP

The automated STC pipeline is evaluated on BabelSememe (Qi et al., 2020a), with splits of 34,964/3,228/3,228 (train/validation/test) instances[4]. To further enhance the performance, the SememeDef (Wang et al., 2025) dataset, containing 107,405 instances[5], is additionally incorporated as

[4]We use the version from Ye et al. (2022).

[5]We use a version that excludes data from BabelSememe.

Category for SRC	Relation Type in HowNet
agent	agent, co_agent, experiencer, host, person_pro, possessor
patient	beneficiary, content, of_part, partner, part_of_touch, patient, patient_attribute, patient_part, possession, result_whole, source_whole, target
product	content_product, patient_product, result, result_content, result_event, result_is_a
instrument	instrument, means, method
location	location, location_fin, location_ini, location_thru, part_position, place_sect, source
time	duration, duration_after_event, duration_before_event, since_period, since_point, time, time_after, time_before, time_feature, time_fin, time_ini, time_range, time_sect
state	existent, state, state_fin, state_ini, succeeding, transition
cause	according_to, cause, concession, condition, purpose
modifier	accompaniment, adjunct, aspect, belong, but, comment, content_compare, contrast, descriptive, direction, domain, emphasis, event_process, host_of, manner, material, material_of, modifier, restrictive, whole
quantity	cost, degree, distance, frequency, patient_value, quantity, quantity_compare, range, scope, sequence, times
related_to	co_event, concerning, equiv, is_a, relate_to, relevant

Table 8: Mapping of relation types in HowNet sememe trees to the categorized ones for SRC.

Sememe Tree Sequence Generation Task

Task Description
You are given a word definition, its sememe set, and the root node sememe.
Your task is to generate the sememe tree sequence for the given word definition.
You should refer to the provided reference examples below to understand the structural patterns.

Output Format Rules
1. The list starts with "START" (a virtual root node of the sememe tree, depth = 1);
2. Generate the sequence using pre-order depth-first traversal: record the current node first, then traverse all its children in order;
3. After traversing each child node, use "BACK+number" to indicate backtracking from the current node (number = depth of the current node before backtracking);
4. After traversing all nodes, backtrack through every ancestor including "START" itself, ending with "BACK1" to terminate the list;
5. The output only contains the given sememes, "START", and "BACK+number";
6. The answer must be wrapped in `<answer></answer>` tags and formatted as a Python list of strings.

Output Structure
First, reason through the problem inside `<think></think>` tags following this 4-step format:
Step 1 - Semantic Analysis: Analyze the definition, map each sememe to its semantic role, and compare with the reference examples to identify similar structural patterns;
Step 2 - Tree Structure: Draw the hierarchical tree with depth annotations, explicitly noting which example's pattern it resembles;
Step 3 - Pre-order DFS Traversal: List the visit order and backtrack sequence step by step;
Step 4 - Verification: Confirm all sememes included ✓, BACK numbers correct ✓, ends with BACK1 ✓.
Then, output only the Python list inside `<answer></answer>` tags.

Reference Examples
{Examples}

Input
- Text Definition: {def_text}
- Sememe Set: {sememe_set}
- Root Node Sememe: {main_sememe}

Output

Table 9: Prompt template for SSP, where the sememe set from ASP and the main sememe from MSP are provided.

Method	BLEU	Strict	Edge	Vertex
TaSTG	17.0	39.7	41.2	48.2
Qwen3-30B-A3B-Instruct	35.8	51.3	52.2	60.8
Deepseek-V3.2	**36.7**	**52.3**	**53.4**	**62.1**

Table 10: Performance comparison on the SSP task, where LLM-Based Methods are under a 3-shot learning setting, with BLEU/Strict denoting accuracy of overall tree generation and Edge/Vertex for edge/node accuracy. Please refer to Ye et al. (2022) for detailed descriptions of the evaluation metrics.

supplementary training data. Each instance in these datasets includes both word sense definition information and a complete sememe tree annotation.

Notably, statistics show that these sememe trees involve 95 relation types. Such a large number increases the difficulty of accurate prediction. Also, it leads to a significant growth in the parameter size of downstream models (e.g., R-GNN) designed to leverage such information. To solve this issue, we observe that the concept relation types in HowNet roughly align with semantic roles (Dong et al., 2010), and accordingly adopt 9 semantic role labels as relation categories. We also add a *modifier* category for non-predicative relations such as those between concepts and their attribute values, and a *related_to* category to cover any remaining types. In this way, we group these 95 types into 11 coarse-grained categories based on their corresponding semantic roles, as detailed in Table 8. This reduction can lower the complexity for models to understand sememe trees.

We adopt DeBERTa-large (He et al., 2021) as the base model for ASP, MSP, and SRC. The evaluation results of each subtask are presented as follows:

For the ASP task, the pipeline achieves 62.7% MAP and 60.3% F1-score. This performance is comparable to that of the SOTA method MSGI (Qi et al., 2022), which attains 67.2% MAP and 57.7% F1-score by leveraging multilingual and multimodal information from BabelSememe. This indicates that supplementary training data can effectively enhance the performance of definition-only methods, enabling them to rival SOTA models with richer information inputs;

Next, MSP attains a 68.9% F1-score, significantly outperforming ASP. This suggests that MSP is less challenging than ASP, and framing it as an independent module can facilitate a more accurate prediction of the root node in sememe trees;

Building on the outputs of ASP and MSP, we evaluate the LLM-based methods on sememe tree sequence generation[6] with a temperature of 0.6 and a top-p value of 0.95. The prompt template for SSP is presented in Table 9. As shown in Table 10, benefiting from both the incorporation of ASP and MSP predictions and LLM's superior tree-structured organization capability, the LLM-based methods outperform the SOTA method TaSTG (Ye et al., 2022);

Finally, SRC achieves remarkable results with an F1-score of 98.2%, effectively classifying parent-child sememe relations. It validates that our reduction strategy reduces model learning difficulty and preserves the critical relation type information to benefit downstream tasks.

B Details of LRC and LE Datasets

To facilitate research on distributional semantics via analogies, BLESS (Baroni and Lenci, 2011) is firstly constructed based on WordNet (Miller, 1995), McRae norms (McRae et al., 2005), and ConceptNet (Speer and Havasi, 2012). EVALution (Santus et al., 2015) is an expansion of BLESS, adding synonymy, antonymy, and other relation types. CogALexV (Santus et al., 2016a) is a challenging subset of EVALution, with words stemmed. ROOT09 (Santus et al., 2016b) is an expansion of CogALexV. K&H+N (Necşulescu et al., 2015) is a non-BLESS-derived dataset in this evaluation suite, built upon K&H (Kozareva and Hovy, 2010) and WordNet (Miller, 1995). These datasets cover various lexical relation types, such as random[7], synonymy, hypernymy, antonymy, and meronymy. Table 11 presents the train/validation/test data statistics of each relation type in the LRC datasets.

For the graded LE task, HyperLex (Vulić et al., 2017) is a widely used dataset that contains noun and verb pairs with entailment ratings. It has two different data splits based on lexical and random, respectively. The lexical split contains 1,133/85/269 word pairs for the train/validation/test sets, and the random split includes 1,831/130/655 ones across the three subsets.

All datasets for LRC and LE are open source, released under either the Creative Commons 4.0 or

[6]The sememe tree can be recovered from the tree sequence by reversing the pre-order traversal, with the `START` symbol marking the initiation of traversal and the `BACK` symbol indicating subtree completion. For example, the tree sequence for the most frequent sense of *husband* is: `"START human spouse BACK3 family BACK3 male BACK3 BACK2 BACK1"`.

[7]The random relation is added to ensure the relation set is exhaustive.

Relation Type	BLESS	K&H+N	EVALution	CogALexV	ROOT09
Random	8,529/609/3,008	18,319/1,313/6,746	-	2,228/-/3,059	4,479/327/1,566
Synonymy	-	-	759/50/277	167/-/235	-
Hypernymy	924/63/350	3,048/202/1,042	1,327/94/459	255/-/382	2,232/149/809
Co_hyponymy	2,529/154/882	18,134/1,313/6,349	-	-	2,222/162/816
Antonymy	-	-	1,095/90/415	241/-/360	-
Meronymy	2,051/146/746	755/48/240	377/25/142	-	-
Event	2,657/212/955	-	-	-	-
Attribute	1,892/143/696	-	903/72/322	-	-
Part_of	-	-	481/28/145	163/-/224	-
Made_of	-	-	218/13/86	-	-

Table 11: Statistics for the number of word pairs of each relation type across the train/validation/test splits in the LRC datasets.

Method	LRC					LE	
	BLESS	K&H+N	EVALution	CogALexV	ROOT09	Lexical	Random
DeBERTa-large (ours)	95.4	98.8	78.7	76.4	94.2	88.0	90.2
Gloss	95.9	99.0	78.5	76.9	93.6	88.3	90.4
Sememe (SememeLRM)	**96.5**	**99.1**	**81.6**	**78.9**	**95.3**	**88.8**	**90.7**

Table 12: Performance comparison on LRC and LE between gloss-based and sememe-based representation for word senses.

Apache 2.0 license.

C Experimental Configuration

We adopt DeBERTa-large (He et al., 2021) as the base model, which consists of 24 layers with 1024 hidden units. Also, R-GCN (Schlichtkrull et al., 2018) is employed as the R-GNN module, with its hidden dimension set to match that of the PLM's hidden layer. Our statistics show that 98.5% of sememe trees have at most 4 layers. Thus, we set the R-GNN to 3 layers so that the features of all the other nodes in the tree can be aggregated to the root node in the vast majority of cases. The task-specific MLP module consists of 1 hidden layer, with the dimension set to double that of the PLM.

Consistent configurations are adopted across all datasets: the batch size is set to 32, the number of training epochs to 10, and the learning rate to 2e-5. We use an AdamW optimizer (Loshchilov and Hutter, 2017) and a cosine decay scheduler with a warmup ratio of 0.1. Additionally, the max number of sememe trees per word is set to 5, and that of gloss tokens per word to 120.

During training, models are evaluated on the validation set every 30 steps, and the checkpoint achieving the best validation performance is selected for the final evaluation on the test set. For datasets without a predefined validation set, the final checkpoint is used for evaluation.

We separately evaluate the performance of SememeLRM when leveraging sememe data from HowNet and our automated STC pipeline. Considering that HowNet has limited vocabulary coverage, for words not covered by HowNet, we supplement them with sememe trees predicted by the STC pipeline. We also augment each sememe tree with a special node linked to the root to record the Part-of-Speech information.

For evaluation metrics, following existing research (Pitarch et al., 2023; Sun et al., 2025), we adopt the weighted F1-score for LRC and Spearman ρ correlation (Spearman, 1904) for LE.

All experiments are conducted with the deep learning framework PyTorch on a NVIDIA Virtual GPU (32GB memory).

D Complete Results for Analysis

Complete performance comparison between gloss-based and sememe-based methods is shown in Table 12. Detailed results for ablation analysis are shown in Table 13.

E Cost-Benefit Analysis

We quantify the computational costs of SememeLRM and compare it with the DeBERTa-large baseline under the same experimental conditions (NVIDIA Virtual GPU), as presented in Table 14. Due to the incorporation of an R-GCN module for structured sememe feature extraction and the adoption of a larger input dimen-

Method	LRC					LE	
	BLESS	K&H+N	EVALution	CogALexV	ROOT09	Lexical	Random
DeBERTa-large (ours)	95.4	98.8	78.7	76.4	94.2	88.0	90.2
sememe set	95.4	98.8	78.9	78.0	94.8	88.6	90.4
GCN + sememe tree	96.2	98.7	80.5	78.7	95.4	88.5	**91.2**
R-GAT + sememe tree	96.2	98.9	80.7	**79.0**	**95.7**	**89.3**	90.8
R-GCN + sememe tree (SememeLRM)	**96.5**	**99.1**	**81.6**	78.9	95.3	88.8	90.7
R-GCN + sememe tree + all outputs fed	96.3	98.9	81.5	78.3	95.6	88.5	90.7

Table 13: Detailed performance comparison for ablation analysis.

Method	Trainable Params	Training Time	LRC	LE
DeBERTa-large (ours)	436M	1.2 min	88.7	89.1
SememeLRM	489M	2.0 min	90.3	89.8

Table 14: Cost-benefit comparison of SememeLRM and DeBERTa-large. Training time denotes per epoch on BLESS.

(***priest***, ***clergyman***)	
Senses for *priest*:	1. {human, occupation, religion}.
Senses for *clergyman*:	1. {human, occupation, religion}.
True label:	hypernymy
Predicted label:	synonymy
(***Australia***, ***island***)	
Senses for *Australia*:	1. {place, proper_name, politics, country, australia}; 2. {place, proper_name, australia}.
Senses for *island*:	1. {land, waters, surround}.
True label:	hypernymy
Predicted label:	meronymy
(***car***, ***roof***)	
Senses for *car*:	1. {land_vehicle, automatic}.
Senses for *roof*:	1. {part, house, head}.
True label:	meronymy
Predicted label:	random
(***fork***, ***hammer***)	
Senses for *fork*:	1. {tool, pick, agricultural}; 2. {tool, eat}; 3. {part, physiology, animal, body}; 4. {part, physiology, plant, limb}; 5. {route, branch}; 6. {part, route, cross}.
Senses for *hammer*:	1. {tool, beat}; 2. {tool, round, sport}; 3. {beat}.
True label:	co-hyponymy
Predicted label:	attribute

Table 15: Examples of incorrect classifications of SememeLRM. Word senses are from OpenHowNet (Qi et al., 2019), presented in the form of sememe sets. Within each sememe set, the first sememe serves as the root node in the corresponding sememe tree. The structured information is excluded to facilitate table presentation. The first two examples are primarily due to the taxonomy inconsistencies between the LRM datasets and HowNet; the third can be attributed to the lack of relevant senses in HowNet; the final case stems from the polysemousness of words.

sion for the MLP, SememeLRM incurs an additional 53M trainable parameters, with a corresponding increase in per-epoch training time. Nevertheless, SememeLRM attains a notable average improvement of 1.6% across all benchmarks. Impressively, through such a relatively limited computational investment, it can even match or outperform DeBERTa-xlarge (750M) and LLaMA3 (8B)-based methods. It is thus worthwhile to introduce structured sememe knowledge for such knowledge-intensive tasks.

F Error Analysis

We collect the misclassifications by SememeLRM and investigate the underlying causes of these errors. Our analysis reveals that the errors primarily stem from three sources:

First, taxonomy inconsistencies between the LRM datasets and HowNet can lead to incorrect predictions. For instance, as shown in the first two examples in Table 15, *priest* is a synonym of *clergyman* in HowNet, as their sememe combinations are the same; additionally, HowNet categorizes *Australia* as a place rather than an *island*. Due to such differences in taxonomic systems, while SememeLRM outputs reasonable relation labels, the intended benefits from introducing structured sememe knowledge may not be fully reflected in current evaluation metrics. Accordingly, it may be worthwhile to explore and develop more appropriate evaluation metrics in future work.

Second, some errors can be attributed to the lack of relevant senses in HowNet, like the third example in Table 15: *roof* in HowNet lacks a sense like "*protective covering on top of a motor vehicle*" in WordNet (Miller, 1995). This deficiency prevents the model from correctly identifying the meronymy relation between *car* and *roof*. This finding also indirectly suggests the value of our automated STC pipeline in expanding and supplementing the SKB resource.

Finally, the polysemousness of words can also result in erroneous predictions. For the last example in Table 15, an excessive number of senses per word may interfere with the model's judgment, preventing it from accurately determining that both *fork* and *hammer* fall under the category of *tool*. This finding further suggests that it would be valuable to explore the construction of evaluation benchmarks with pre-specified word senses.