WSDPO: A Generative Word Sense Disambiguation Framework with Chain-of-Thought and Preference Optimization

Kunpeng Kang[1,4,*] **Shuaimin Li**[2,*] **Kaiyuan Zhang**[1,4], **Luyang Zhang**[1,4], **Jiasheng Si**[1,4], **Bing Xu**[3], **Kehai Chen**[3], **Muyun Yang**[3], **Wenpeng Lu**[1,4,†]

[1]Key Laboratory of Computing Power Network and Information Security, Ministry of Education, Shandong Computer Science Center (National Supercomputer Center in Jinan), Qilu University of Technology (Shandong Academy of Sciences), Jinan, China

[2]Shenzhen Institutes of Advanced Technology, Chinese Academy of Sciences, Shenzhen, China

[3]Harbin Institute of Technology, Harbin, China

[4] Shandong Provincial Key Laboratory of Computing Power Internet and Service Computing, Shandong Fundamental Research Center for Computer Science, Jinan, China

kunpeng.kang@foxmail.com, sm.li2@siat.ac.cn, wenpeng.lu@qlu.edu.cn

Abstract

Word sense disambiguation (WSD) is a foundational task in natural language processing. Recent research has reformulated WSD for large language models (LLMs) as a generative task, where the model produces a definition to convey the intended meaning of an ambiguous word in context. In practice, most existing approaches implement this formulation through straightforward supervised fine-tuning, which tends to prioritize superficial context-to-gloss memorization over true contextual sense discrimination, leading to degraded performance on less frequent senses (LFS), particularly in unseen settings. To address this issue, we propose WSDPO, a training framework for generative WSD with chain-of-thought (CoT) and preference optimization. WSDPO consists of three stages: (1) disambiguation-aware CoT construction, which produces training data containing explicit disambiguation steps for the later stage; (2) disambiguation-guided supervised fine-tuning, which explicitly trains the model to discriminate word sense before generating the final definition; and (3) preference-based optimization, which further strengthens the model's ability to generate sense-faithful definitions by optimizing it using preference pairs constructed from multiple sampled CoT outputs. Extensive experiments across benchmark datasets and multiple backbone LLMs demonstrate that WSDPO achieves substantial performance gains on rare and unseen settings, and exhibits strong generalization in standard evaluation settings.[1]

1 Introduction

Word sense disambiguation (WSD) is a longstanding and fundamental task in natural language processing, aiming to identify the correct meaning of a polysemous word in the given context (Navigli, 2009; Zhang et al., 2025c; Navigli, 2026). WSD is crucial for semantic understanding and directly affects the performance of downstream tasks such as machine translation (Tran et al., 2025), sentiment analysis (Zhang et al., 2023), and information retrieval (Dadure et al., 2024).

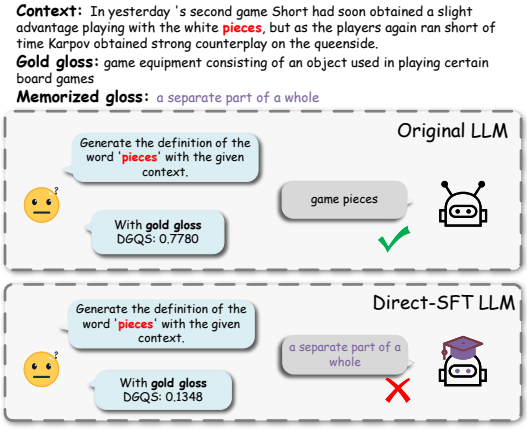


Figure 1: Challenge of direct-SFT LLMs in generative WSD. The original LLM is able to generate a definition that matches the intended sense in context, whereas the direct-SFT LLM tends to output a memorized gloss that is inconsistent with the context. Obviously, Direct-SFT weakens the LLM's disambiguation ability on low frequency and unseen senses.

Recently, large language models (LLMs) have demonstrated substantial progress across a range of high-level reasoning tasks, including code generation (Zhang et al., 2025b), mathematical reasoning (Yu et al., 2025), and Text-to-SQL (Liu et al., 2025). However, this progress does not fully translate to fine-grained semantic understanding in language, where WSD remains a persistent challenge (Kocoń et al., 2023; Meconi et al., 2025). Fol-

[*] Equal contribution

[†] Corresponding author

[1]The source code, datasets, and models are publicly available at https://github.com/KunpengKang/WSDPO.

Proceedings of the 64th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), pages 34870–34889

lowing this formulation, most existing approaches adopt supervised instruction tuning to map target words in context directly to their corresponding sense definitions (Yae et al., 2025; Zhang et al., 2025c).

Existing generative WSD approaches predominantly rely on supervised instruction tuning (Zhang et al., 2025d), which directly maps target words in context to their corresponding sense definitions. For example, (Periti et al., 2024) apply instruction tuning with parameter-efficient adaptation to enable LLMs to generate definitions. Subsequent work extends this paradigm to multilingual and cross-lingual settings, investigating the generalization of definition generation across 22 languages (Periti et al., 2025).

Although significant progress has been made, existing generative WSD methods still suffer from a critical limitation. They mainly rely on supervised fine-tuning (SFT) to directly generate glosses for ambiguous words, without explicitly modeling the reasoning process underlying sense disambiguation. In practice, however, definition generation is not a single-step mapping from context to gloss, but a two-stage process that first disambiguates the ambiguous word by explicitly identifying its intended sense, and then generates the corresponding definition (Zhang et al., 2025c).

However, existing generative WSD approaches fail to explicitly model this reasoning process, biasing direct SFT toward frequent gloss recall and resulting in degraded performance on less frequent senses (LFS) and unseen senses. For instance, as illustrated in Figure 1, during supervised fine-tuning, the direct-SFT LLM is exposed to the target word *"pieces"* paired with the gloss *"a separate part of a whole"*, while the gold gloss *"game equipment consisting of an object used in playing certain board games"* is never observed. As a result, the direct-SFT model fails on this case by reproducing the only gloss it has been trained on, rather than generating a contextually appropriate definition.

To address this issue, we introduce WSDPO, a novel generative WSD training framework with chain-of-thought (CoT) and preference optimization. WSDPO guides the model from disambiguation to sense-faithful definition generation through a sequence of carefully designed stages. First, it constructs disambiguation-aware chains-of-thought (CoT), which produces training data containing explicit disambiguation steps for the later stage, thereby decoupling sense identification from definition generation and providing clear supervision for both tasks. Building on this foundation, the model is then fine-tuned in a disambiguation-guided manner, explicitly training it to discriminate among word senses before producing the final definition, which strengthens its inherent disambiguation capability rather than encouraging superficial context-to-gloss memorization. Finally, WSDPO employs preference-based optimization, refining the model's behavior by ranking multiple CoT outputs and encouraging it to favor sense-faithful definitions over those that are merely frequent or memorized. Together, these stages create a cohesive training process that systematically enhances the model's ability to generate contextually accurate and semantically faithful definitions.

The main contributions of our work are summarized as follows:

- We propose WSDPO, a novel training framework for generative WSD with CoT-augmented preference optimization. WSDPO is the first to explicitly model the two-stage reasoning process and to align model behavior toward producing sense-faithful definitions through preference optimization.
- We introduce *disambiguation-aware CoT construction* for generative WSD, which decomposes definition generation into explicit disambiguation and generation steps to better reflect the two-stage reasoning process. We further incorporate *preference-based optimization* into generative WSD, optimizing the model over preference pairs to further encourage sense-faithful definition generation.
- We conduct extensive experiments on multiple benchmark datasets and backbone LLMs under regular, LFS, unseen word, and unseen definition settings. Results demonstrate that WSDPO can achieve strong overall performance, with particularly pronounced gains in long-tail scenarios.

2 Related Work

2.1 Generative WSD

Word sense disambiguation aims to determine the correct meaning of the target word in the context. Although existing LLM-based approaches enhance WSD performance through such as multi-agent debate (Zhang et al., 2025a) and knowledge distillation (Ming et al., 2025), they still treat WSD as

a label classification problem, which constrains the model to predefined sense inventories instead of generating contextualized definitions, limiting its ability to generalize beyond fixed label spaces and to capture fine-grained semantic variations in context, thereby underutilizing the semantic reasoning and generative capabilities of LLMs (Navigli, 2026). To overcome this limitation, a more natural formulation is to prompt models to directly generate sense definitions. Although prior work in this formulation typically treats the task as pure definition generation, directly learning a mapping from context to gloss, it inherently requires resolving the target word's sense before generation, making it fundamentally a generative WSD task. Most existing approaches train models to directly generate the gold gloss from context. Early work fine-tunes instruction-tuned encoder–decoder models on lexical resources and dictionary corpora (Giulianelli et al., 2023), and subsequent studies scale this paradigm to larger decoder-only LLMs using parameter-efficient fine-tuning on multiple dictionary datasets (Periti et al., 2024). More recent efforts extend definition generation to multilingual settings (Periti et al., 2025) and further explore explaining novel word senses using open-weights LLMs (Fedorova et al., 2025). However, these paradigms perform well on most frequent senses but degrade on rare and unseen cases, as they primarily reproduce glosses from context. We address this limitation by introducing an explicit disambiguation step during training.

2.2 Long-tail WSD

One of the major challenges in Word Sense Disambiguation (WSD) is the data sparsity induced by the Zipfian distribution of senses in natural language (Kilgarriff, 2004). Previous work has attempted to mitigate this problem by constructing specialized datasets or tasks for rare and zero-shot senses (Holla et al., 2020; Blevins et al., 2021; Barba et al., 2021; Yoon et al., 2022), enriching sense representations with external lexical knowledge (Kumar et al., 2019; Scarlini et al., 2020; Buda et al., 2018; Zhang and Li, 2025), or modifying the learning process to better account for less frequent senses (Su et al., 2022). However, most existing studies address long tail effects in classification-based WSD settings, whereas little attention has been paid to long tail challenges in LLM-based generative WSD.

3 Method

In this section, we begin by defining the generative WSD task. Then, we describe the specific strategies implemented within our framework.

3.1 Task Formulation

Given a context containing an ambiguous target word, the goal of generative WSD is to generate a sense definition that corresponds to the intended meaning of the target word in context. Formally, let the context be a token sequence $C = (w_1, w_2, \ldots, w_m)$, and let w_t denote the target word occurring at position t. Generative WSD is formulated as a conditional text generation task, where the model produces a definition sequence $y = (y_1, y_2, \ldots, y_L)$ conditioned on C, which is formulated as

$$y = \mathrm{LLM}(C), \tag{1}$$

where y represents the intended sense of w_t in the given context.

3.2 WSDPO Architecture

As shown in Figure 2, the WSDPO framework consists of three main stages: (1) *disambiguation-aware CoT construction*, (2) *disambiguation-guided supervised fine-tuning*, and (3) *preference-based optimization*. First, given a training instance, a strong LLM is prompted to construct CoT traces, where each trace contains both the disambiguation process and the generative definition of the ambiguous word in context. The generated traces are then filtered through a quality assessment procedure, yielding a set of CoT-enhanced training instances. Next, the filtered CoT-enhanced data is used for supervised fine-tuning, enabling the model to better internalize the two-step generative WSD process: first performing sense disambiguation and then generating definition. Finally, to further strengthen the model's ability to produce sense-faithful definitions, we perform preference optimization on the SFT model. Multiple candidate outputs are sampled from the model to construct preference pairs, which are compared by quality, and the model is then optimized using the resulting preference signals. In the following sections, we provide a detailed description of these stages.

3.2.1 Disambiguation-Aware CoT Construction

In our framework, the *disambiguation-aware CoT construction* stage prepares the training data for the

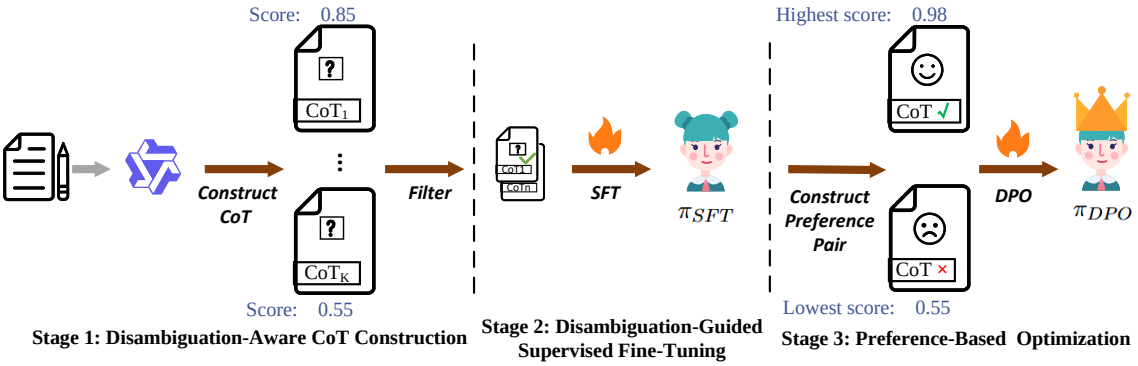


Figure 2: Overview of the training framework for generative WSD via Chain-of-Thought and Preference Optimization. The framework consists of three stages: (1) Disambiguation-aware CoT construction, where we directly construct CoT traces and filter them through score-based quality evaluation; (2) Disambiguation-guided supervised fine-tuning, where the model is trained on the filtered CoT traces; and (3) Preference-based optimization, where multiple CoT outputs are sampled to construct preference pairs and the model is optimized using DPO.

subsequent stage by explicitly modeling the two-stage reasoning process involved in sense disambiguation prior to definition generation. To avoid the high cost of human annotation, we employ an LLM to automatically generate CoT traces that explicitly include the sense disambiguation process for existing WSD datasets.

Specifically, each training instance consists of a context C, a target word w_t, and its gold gloss. Given this input, we use Qwen2.5-72B-Instruct (Qwen et al., 2024) to generate K diverse CoT traces that approximate the two-stage reasoning process, where the model first identifies the intended sense in context and then generates the corresponding definition. Although the gold gloss is provided to the CoT constructor as supervision, the generated CoT traces may vary in quality. Therefore, we assess the quality of each CoT trace by measuring the DGQS score (Zhang et al., 2025c) between the final generated gloss and the gold gloss, and we filter out low-quality CoT traces accordingly. The resulting high-quality CoT traces are then used to construct the CoT-augmented dataset for subsequent *disambiguation-guided supervised fine-tuning* and *preference-based optimization*. The prompts used for *disambiguation-aware CoT construction* and examples of the generated CoT traces are provided in Appendix A.

3.2.2 Disambiguation-Guided Supervised Fine-Tuning

After obtaining the filtered CoT-augmented training dataset from the previous stage, we perform supervised fine-tuning on this data. This stage aims to encourage the model to internalize a two-stage reasoning pattern, where it first performs sense disambiguation and then generates the corresponding definition, thereby improving its generalization to rare senses and unseen cases.

For each training instance, the user input prompt contains the context C and the target word w_t. The assistant output sequence is given by the filtered CoT trace, which includes both the disambiguation step and the definition generation step, reflecting the two-stage reasoning process observed in human interpretation. We denote the user input prompt as p and the assistant output CoT trace as t. The SFT objective is defined as a conditional next-token prediction loss:

$$\mathcal{L}_{\text{SFT}} = -\mathbb{E}_{(p,t)\sim\mathcal{D}_{\text{S}}} \left[\log \pi_{\text{base}}(t \mid p)\right], \quad (2)$$

where $\mathcal{D}_{\text{S}}$ denotes the CoT-enhanced supervised training set, p is the user prompt, t is the target CoT trace, and π_{base} represents the pretrained LLM. After SFT, we obtain the model π_{SFT}, which serves as the reference model for subsequent *preference-based optimization*. Details of the prompt format and CoT data is provided in Appendix B.

3.2.3 Preference-Based Optimization

At this stage, we address a limitation of supervised fine-tuning, namely that it does not consistently encourage the model to prefer higher-quality outputs over inferior ones (Zhang et al., 2025b; Qi et al., 2025). To further strengthen the model's ability to produce sense-faithful definitions, we apply Direct Preference Optimization (DPO) (Rafailov et al., 2023), encouraging it to favor sense-faithful definitions over those that are merely frequent or

memorized. A supplementary discussion on the mechanics of DPO, focusing on its loss formulation, implicit reward modeling, and token-wise credit assignment, can be found in Appendix C.

Preference Pair Construction. DPO requires a preference dataset consisting of paired positive and negative responses. In our setting, each preference instance consists of a prompt and a pair of CoT traces, including a positive and a negative example. Specifically, for each training instance, we sample N distinct CoT traces from the reference model π_{SFT} conditioned on the prompt. From each generated CoT trace, we extract the predicted gloss using regular expressions and compute its DGQS score (Zhang et al., 2025c) with respect to the gold gloss. The CoT traces corresponding to the same input are then ranked according to their DGQS scores. The trace with the highest score is selected as the positive example, while the trace with the lowest score is selected as the negative example.

Preference-Based Optimization Objective. DPO optimizes the policy to prefer higher-quality CoT traces over inferior ones, while implicitly regularizing the policy to stay close to the reference model π_{SFT}. Formally, given a prompt p that includes the context C and the target word w_t, we denote the preferred and dispreferred CoT traces as t^+ and t^-, respectively. The loss function is defined as:

$$\mathcal{L}_{\text{DPO}} = -\mathbb{E}_{(p,t^+,t^-)\sim\mathcal{D}_{\text{P}}} \log\sigma\left(\beta\log\frac{\pi_{\text{DPO}}(t^+\mid p)}{\pi_{\text{SFT}}(t^+\mid p)} - \beta\log\frac{\pi_{\text{DPO}}(t^-\mid p)}{\pi_{\text{SFT}}(t^-\mid p)}\right), \tag{3}$$

where $\mathcal{D}_{\text{P}}$ denotes the preference dataset, $\sigma(\cdot)$ is the sigmoid function, β is a hyperparameter controlling the strength of the implicit KL regularization, π_{SFT} is the reference model obtained via supervised fine-tuning, and π_{DPO} is the optimized model initialized from π_{SFT}.

4 Experimental Setup

4.1 Datasets

Training dataset. We use SemCor (Miller et al., 1993) as the base training corpus. SemCor contains 33,362 sentences and 226,036 target word instances, each manually annotated with WordNet 3.0 senses.

Evaluation datasets. To evaluate performance on less frequent senses and on unseen words and definitions, we follow previous work (Barba et al., 2021) to derive several evaluation subsets (LFS, 0-lex, 0-lex-def, and 0-def) from the ALL benchmark (Raganato et al., 2017) and include additional challenge benchmarks.

- **LFS**: a subset of the ALL dataset containing instances whose annotated sense is not the most frequent sense of the target word but has appeared at least once in the training corpus. It is used to evaluate model performance on less frequent senses.
- **0-lex**: a subset of the ALL dataset containing instances whose target lexeme never appears as a target word in the training corpus. It is used to evaluate generalization to unseen words.
- **0-lex-def**: a subset of the ALL dataset containing instances where the target lexeme appears in the training corpus, but the associated definition has never been observed for that lexeme. It is used to evaluate generalization to unseen sense definitions of seen words.
- **0-def**: a subset of the ALL dataset containing instances whose definitions have never been seen during training, regardless of whether the lexeme itself appears in the corpus. It is used to evaluate generalization to unseen definitions.
- **42D** (Maru et al., 2022): a manually curated challenge set spanning 42 distinct text domains, containing senses that are absent from SemCor and are not the most frequent in WordNet. It is used to evaluate strong generalization to real-world rare and unseen senses.
- **ALL** (Raganato et al., 2017): a unified WSD benchmark aggregating Senseval-2 (Edmonds and Cotton, 2001), Senseval-3 (Snyder and Palmer, 2004), SemEval-2007 (Pradhan et al., 2007), SemEval-2013 (Navigli et al., 2013), and SemEval-2015 (Moro and Navigli, 2015). It is used to assess regular WSD performance.

4.2 Evaluation Metrics and Baselines

The evaluation of our models on the aforementioned test datasets is conducted using the RoDEval metric suite (Zhang et al., 2025c). To assess definition generation quality, we adopt two

Models	LFS		0-lew		0-lew-def		0-def		42D		ALL	
	DGQS	R-1	DGQS	R-1	DGQS	R-1	DGQS	R-1	DGQS	R-1	DGQS	R-1
Direct-SFT Models												
LLaMA2-Dictionary	54.80	16.73	69.99	22.35	53.19	15.07	60.13	18.17	62.18	21.12	62.84	20.25
LLaMA3-Dictionary	54.86	16.25	70.10	23.58	53.55	15.28	60.48	18.56	61.71	20.68	62.47	19.54
LlamaDictionary-en	48.96	11.92	66.32	18.76	47.41	11.64	55.02	14.24	56.18	14.37	57.78	14.95
LlamaDictionary-sv-de-en	48.80	10.54	68.66	16.50	48.49	11.20	57.21	13.35	56.45	14.28	57.72	13.60
LlamaDictionary-it-es-pl-sv-ru-de-fr-en	51.96	11.58	70.44	17.26	51.98	11.49	60.23	13.84	58.40	14.17	61.15	14.69
LLaMA3.2-3B-Instruct												
Base	51.80	14.01	64.47	17.48	51.99	13.82	56.64	15.58	56.23	16.86	58.69	17.13
+ Direct SFT	58.94	34.30	65.70	29.42	48.24	18.96	56.19	23.90	53.77	22.65	**82.67**	**70.13**
+ CoT-enhanced SFT (ours)	69.67	47.11	69.92	31.82	55.46	**23.50**	62.84	26.91	58.73	24.02	80.40	62.39
+ CoT-enhanced SFT + DPO (ours)	**71.00**	**47.43**	**72.70**	**32.33**	**56.57**	23.43	**64.39**	**28.47**	**60.34**	**25.70**	82.32	64.61
Qwen2.5-3B-Instruct												
Base	53.76	10.86	67.09	17.75	52.90	10.84	60.02	13.89	54.38	12.55	58.37	12.91
+ Direct SFT	64.06	40.35	66.94	29.52	52.08	21.50	58.39	24.42	53.59	23.01	**83.00**	**70.74**
+ CoT-enhanced SFT (ours)	**69.44**	**46.12**	68.53	30.47	55.54	22.84	61.60	26.18	**59.28**	**24.44**	80.92	63.03
+ CoT-enhanced SFT + DPO (ours)	68.57	45.17	**70.91**	**31.04**	**55.82**	**23.38**	**62.54**	**26.87**	59.15	23.59	81.53	63.92

Table 1: Best scores are shown in bold, and second-best scores are underlined. Results are reported using DGQS and R-1 across datasets. Direct SFT models are evaluated under their original settings without access to our training data. For instruction-tuned LLMs, we implement Direct-SFT following prior work as a strong baseline, and compare it with our proposed CoT-enhanced SFT and CoT-enhanced SFT + DPO.

complementary metrics: the *Definition Generation Quality Score* (DGQS) and *ROUGE-1* (R-1). We adopt DGQS as the primary evaluation metric, since conventional generation metrics implicitly assume a single "correct" definitional form and overlook the inherent diversity of natural language expressions, leading to systematic underestimation of LLMs' definition generation capability. (Zhang et al., 2025c).

We compare our approach with two baseline settings: (i) the original instruction-tuned base LLMs evaluated under a zero-shot prompting setting, and (ii) Direct SFT models, representing the supervised training paradigm used in prior generative WSD work. Further implementation details and model specifications are provided in Appendix D.

4.3 Implementation Details

We conduct experiments using two instruction-tuned base LLMs, LLaMA3.2-3B-Instruct[2] (Grattafiori et al., 2024) and Qwen2.5-3B-Instruct[3] (Qwen et al., 2024), representing different model families. All models are evaluated under identical settings, with definition quality measured using the RoDEval evaluation framework (Zhang et al., 2025c). We adopt greedy decoding with temperature set to 0 to ensure deterministic generation and fair comparison across models and datasets. Additional implementation details are provided in Appendix E.

[2] https://huggingface.co/meta-llama/Llama-3.2-3B-Instruct

[3] https://huggingface.co/Qwen/Qwen2.5-3B-Instruct

5 Results and Analyses

5.1 Main Results

Table 1 reports detailed experimental results across LFS, unseen word, unseen definition, 42D, and ALL benchmarks.

Across both backbone LLMs, our method consistently outperforms the base model and Direct SFT on all evaluation settings, with the most substantial gains appearing on LFS and unseen definition datasets. On LLaMA3.2-3B-instruct, our approach improves DGQS by 12.06 on LFS and 8.33 on 0-lex-def, while on Qwen2.5-3B-instruct it yields gains of 5.38 on LFS and 3.74 on 0-lex-def. Our method also achieves the best performance on 42D and maintains strong overall results on ALL. Since DGQS gains are consistently larger than R-1 gains, the improvements mainly reflect stronger semantic consistency at the sense level rather than increased lexical overlap with the gold gloss. This aligns with the goal of generative WSD, namely producing definitions that express the intended sense rather than reproducing surface lexical forms.

A comparison between different training stages further clarifies the contribution of each component. CoT-enhanced SFT already brings clear DGQS gains over Direct SFT, especially on LFS and 0-lex-

def, indicating that explicitly supervising the disambiguation process helps the model better handle less frequent senses that are only sparsely observed during training. When DPO is added on top of SFT, we observe additional and often larger improvements under unseen word and unseen definition settings. This suggests that preference optimization encourages the model to select CoT traces that are more likely to lead to the correct sense definition, which results in more reliable definition generation and stronger robustness, especially in cases that are not covered by supervised learning alone. Overall, the results suggest a complementary relationship between the two stages: SFT helps the model learn to identify the correct sense through explicit supervision, while DPO further strengthens the tendency to choose sense-consistent reasoning paths during generation.

Across different evaluation splits, the performance gains differ between LFS, unseen word, and unseen definition settings. The largest gains appear on LFS, showing that our method enhances the ability to distinguish less frequent senses rather than overfitting to the most frequent ones. On 0-lex and 0-def, the improvements are smaller in absolute magnitude but remain stable across backbones, indicating that the model generalizes better to unseen words and unseen definitions without relying on memorized dictionary entries. Strong performance on 42D, which contains senses entirely absent from SemCor, further confirms that the learned disambiguation behavior transfers to truly unseen sense situations.

We also compare our method with dictionary-trained and multilingual Direct SFT models. These models benefit from broader lexical coverage and indeed improve performance in some unseen cases, but their gains are less pronounced on LFS and unseen-definition subsets, where disambiguation ability is more critical. In contrast, our method achieves larger DGQS improvements without introducing any additional training data, suggesting that its effectiveness stems from learning stronger sense disambiguation behavior rather than from expanded definition exposure.

5.2 Influence of CoT Quality Filtering

Table 2 reports the results of training WSDPO with and without filtering low-quality CoT traces. We observe that filtering generally brings consistent improvements over the unfiltered setting under both SFT and SFT+DPO. The gains are particularly notable on the LFS split, where the DGQS score increases from 62.60 to 66.34 under SFT and from 64.53 to 66.59 under SFT+DPO. This suggests that removing noisy or misleading CoT traces provides cleaner disambiguation supervision and helps the model better handle less frequent senses that are only sparsely observed during training. At the same time, we also note that the unfiltered setting sometimes yields slightly better performance on unseen senses, which indicates that retaining a more diverse set of disambiguation traces may improve the model's robustness in out-of-distribution scenarios.

Compared with SFT, the gains from filtering are smaller after applying DPO, indicating that preference optimization already mitigates part of the noise in low-quality CoT traces because preference optimization encourages the model to favor higher-quality reasoning paths. Even so, the best performance is still obtained when filtering and DPO are used together, showing that the two mechanisms are complementary.

Datasets	Base	without Filtering		with Filtering	
		SFT	SFT+DPO	SFT	SFT+DPO
LFS	51.80	62.60	64.53	66.34	**66.59**
0-lex	64.47	66.63	67.69	67.94	**68.63**
0-lex-def	51.99	55.18	**56.55**	54.57	55.62
0-def	56.64	60.99	**62.39**	62.39	62.27
42D	56.23	59.98	**61.66**	59.95	60.29
ALL	58.69	68.90	72.68	76.74	**77.63**

Table 2: DGQS comparison of WSDPO models trained with CoT traces under settings with and without filtering. Filtering removes low-quality CoT traces before training. All experiments use LLaMA3.2-3B-Instruct with 50k training samples.

5.3 Data Scaling Law for WSDPO

To study the effect of training data size on WSDPO, we conduct a data scaling experiment by training the model on subsets of the CoT-augmented corpus (25k, 50k, and the full set). The models are evaluated on splits with different distribution properties, and the results are reported in Table 3.

We observe that increasing the amount of training data yields steady DGQS gains on the LFS split, suggesting that additional CoT supervision helps the model better handle less frequent senses that are seen during training. The All benchmark shows a similar upward trend, which is consistent with the fact that most definitions in this split are covered by the training data. In contrast, on splits

Model	LFS	0-lex	0-lex-def	0-def	42D	ALL
Base	51.80	64.47	51.99	56.64	56.23	58.69
25k	64.66	70.30	56.25	62.79	58.62	74.66
50k	66.59	68.63	55.62	62.27	60.29	77.63
Full	**71.00**	**72.70**	**56.57**	**64.39**	**60.34**	**82.32**

Table 3: DGQS results on LLaMA3.2-3B-Instruct across six benchmark datasets. The Base model is the original instruction-tuned model, and the 25k / 50k / Full rows report WSDPO trained with different data scales.

involving unseen words or unseen definitions (0-lex, 0-lex-def, and 0-def), even a relatively small subset of training data already brings substantial improvements over the base method, while further increasing the data size yields smaller additional gains. This suggests that WSDPO benefits not mainly from broader exposure to training glosses, but rather from the disambiguation supervision provided by the CoT traces. Taken together, these results reveal a complementary scaling pattern: for less frequent but still seen senses, performance continues to improve as more training data is used; for unseen words and unseen definitions, most of the benefits can already be obtained with a relatively small amount of CoT-based supervision, showing that WSDPO remains effective even under limited training data.

5.4 More Data Strengthens Frequency Bias in Direct SFT

To better understand how different training paradigms behave on LFS instances, we conduct a fine-grained analysis of the definitions generated by Direct SFT and CoT-enhanced SFT models.

Rather than merely distinguishing between memorized and novel definitions, we further provide a fine-grained categorization of memorized outputs based on the relative frequency of the recalled gloss in the training corpus. Specifically, we divide them into four categories: (1) Memorized Higher-Frequency, where the recalled gloss appears more frequently than the gold gloss; (2) Memorized Equal-Frequency; (3) Memorized Lower-Frequency; and (4) Novel Definition Generation, where the model produces a definition that has not previously appeared in the training data. This more detailed decomposition enables us to examine whether the model exhibits a tendency to preferentially revert to more frequent sense interpretations when confronted with LFS instances, rather than genuinely discriminating rare senses.

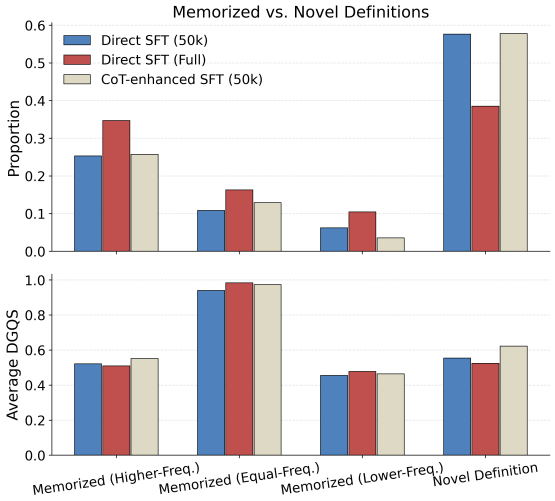


Figure 3: Behavioral analysis of memorized and novel definition generation on the LFS dataset. The upper panel reports the proportions of these output types, while the lower panel presents the average DGQS for each type.

The upper part of Figure 3 shows the distribution of these output types. Direct SFT models produce a substantially larger proportion of *Memorized-Higher-Frequency* cases, indicating a bias toward replacing rare senses with more frequent ones. In contrast, the CoT-enhanced SFT model reduces reliance on higher-frequency memorized glosses and generates more novel definitions, suggesting that it better preserves sense distinctions for rare senses. The lower part of Figure 3 reports DGQS scores within each category. While the two paradigms behave similarly when the recalled gloss has equal frequency, Direct SFT exhibits lower DGQS on *Memorized-Lower-Frequency* and *Novel Definition Generation* cases. This indicates that increasing training data under Direct SFT primarily strengthens memorization of frequent glosses, while providing limited improvement in sense interpretation for rare senses, whereas our CoT-enhanced paradigm yields higher-quality novel definitions for LFS instances.

A similar analysis on unseen definition settings is provided in Appendix G.

5.5 Case Study

To better demonstrate the effectiveness of our framework, we present a qualitative case study comparing our model with the Direct SFT baseline on a representative example.

Table 4 shows an instance where correctly interpreting the target word requires going beyond

a general definition and incorporating contextual cues. In this example, the golden reference defines *disciplined* as "obeying the rules". However, the context discusses educational philosophy, where *disciplined* more precisely refers to instruction that is strictly structured and regulated.

Context: "The whole notion of creativity in education was (and is) part and parcel of a romantic rebellion against *disciplined* instruction, which was (and is) regarded as authoritarian, a repression and frustration of the latent talents and the wonderful, if as yet undefined, potentialities inherent in the souls of all our children."
Target Word: *disciplined*
Golden Reference: *obeying the rules*
Our: *characterized by strict structure and regulation*

Table 4: An example instance where the context requires a more specific semantic interpretation than the concise reference gloss.

For this example, the Direct SFT model generates "obeying the rules", achieving a DGQS of 1.00. In contrast, our model produces "characterized by strict structure and regulation", which more accurately captures the notion of disciplined instruction in this context but receives a lower score of 0.80.

This contrast shows that Direct SFT tends to reproduce memorized glosses that match the reference form but may overlook context-specific meaning, whereas our model generates more contextually faithful definitions, reflecting the diversity of valid sense expressions. It also reveals a limitation of metric-based evaluation: metrics are heavily dependent on a single reference gloss, which can penalize semantically valid predictions that deviate from it, even when they better reflect the context. This suggests that effective generative WSD requires explicit sense disambiguation rather than surface-level gloss matching.

6 Conclusion

In this paper, we focus on key limitations that constrain current progress in generative WSD. Existing approaches typically formulate the task as a single-step mapping from context to gloss, which does not align with the two-stage process underlying human sense interpretation and definition generation. To address this issue, we propose a training framework for generative WSD with Chain-of-Thought augmented preference optimization. The framework explicitly models the two-stage reasoning process, in which sense disambiguation is performed first and definition generation follows, and employs preference optimization to further strengthen model performance. Extensive experiments demonstrate that WSDPO achieves substantial performance gains in rare and unseen settings, while also exhibiting strong generalization under standard evaluation conditions. In future work, we plan to explore more efficient training frameworks to reduce token generation overhead, further improve training efficiency, and extend the approach to broader semantic reasoning and definition generation scenarios.

Limitations

Although this work is the first to propose a training framework for generative WSD with Chain-of-Thought and Preference Optimization and demonstrates consistent performance gains over existing training paradigms, it still has several limitations. First, by explicitly decoupling the generative WSD process into disambiguation and definition generation, our framework introduces longer outputs and additional reasoning steps, which may lead to increased inference latency. Second, we have not yet applied the framework to larger-scale LLMs due to computational and resource constraints, and thus its scalability to larger models remains to be systematically evaluated. Third, current evaluation metrics largely rely on a single reference gloss, which may underestimate LLMs' performance when multiple semantically valid definitions exist. In future work, we plan to explore more robust evaluation strategies that better capture semantic diversity and contextual appropriateness.

Acknowledgements

This work is supported by the National Natural Science Foundation of China (No.62376130, No.62402258, No.62276077), the Taishan Scholars Program of Shandong Province (No.TSPD20240814, No.TSQN202507242), Program of New Twenty Policies for Universities of Jinan (No.202333008), the Pilot Project for Integrated Innovation of Science, Education, and Industry of Qilu University of Technology (Shandong Academy of Sciences) (No.2025ZDZX01), Shandong Talent Introduction Program (No.WSR2025005), and the Open Project of the Key Laboratory of Computing Power Network and Information Security, Ministry of Education (No.2023ZD027).

References

Edoardo Barba, Tommaso Pasini, and Roberto Navigli. 2021. ESC: Redesigning WSD with extractive sense comprehension. In *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 4661–4672.

Terra Blevins, Mandar Joshi, and Luke Zettlemoyer. 2021. FEWS: Large-scale, low-shot word sense disambiguation with the dictionary. In *Proceedings of the 16th Conference of the European Chapter of the Association for Computational Linguistics*, pages 455–465.

Mateusz Buda, Atsuto Maki, and Maciej A. Mazurowski. 2018. A systematic study of the class imbalance problem in convolutional neural networks. *Neural Networks*, 106:249–259.

Pankaj Dadure, Partha Pakray, and Sivaji Bandyopadhyay. 2024. Mathematical information retrieval: A review. *ACM Computing Surveys*, 57(3):1–34.

DeepSeek-AI, Qihao Zhu, Daya Guo, Zhihong Shao, Dejian Yang, Peiyi Wang, Runxin Xu, Y. Wu, Yukun Li, Huazuo Gao, Shirong Ma, Wangding Zeng, Xiao Bi, Zihui Gu, Hanwei Xu, Damai Dai, Kai Dong, Liyue Zhang, Yishi Piao, and 21 others. 2024. Deepseek-Coder-V2: Breaking the barrier of closed-source models in code intelligence. *Preprint*, arXiv:2406.11931.

Philip Edmonds and Scott Cotton. 2001. Senseval-2: overview. In *Proceedings of SENSEVAL-2 2nd International Workshop on Evaluating Word Sense Disambiguation Systems*, pages 1–5.

Mariia Fedorova, Andrey Kutuzov, Francesco Periti, and Yves Scherrer. 2025. Explaining novel senses using definition generation with open language models. In *Findings of the Association for Computational Linguistics: EMNLP 2025*, pages 22294–22302.

Mario Giulianelli, Iris Luden, Raquel Fernandez, and Andrey Kutuzov. 2023. Interpretable word sense representations via definition generation: The case of semantic change analysis. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics*, pages 3130–3148.

Aaron Grattafiori, Abhimanyu Dubey, Abhinav Jauhri, Abhinav Pandey, Abhishek Kadian, Ahmad Al-Dahle, Aiesha Letman, Akhil Mathur, Alan Schelten, Alex Vaughan, Amy Yang, Angela Fan, Anirudh Goyal, Anthony Hartshorn, Aobo Yang, Archi Mitra, Archie Sravankumar, Artem Korenev, Arthur Hinsvark, and 542 others. 2024. The Llama 3 herd of models. *Preprint*, arXiv:2407.21783.

Nithin Holla, Pushkar Mishra, Helen Yannakoudakis, and Ekaterina Shutova. 2020. Learning to learn to disambiguate: Meta-learning for few-shot word sense disambiguation. In *Findings of the Association for Computational Linguistics: EMNLP 2020*, pages 4517–4533.

Edward J. Hu, Yelong Shen, Phillip Wallis, Zeyuan Allen-Zhu, Yuanzhi Li, Shean Wang, Lu Wang, and Weizhu Chen. 2021. Lora: Low-rank adaptation of large language models. *Preprint*, arXiv:2106.09685.

Adam Kilgarriff. 2004. How dominant is the commonest sense of a word? In *Proceedings of the International Conference on Text, Speech and Dialogue*, pages 103–111.

Jan Kocoń, Igor Cichecki, Oliwier Kaszyca, Mateusz Kochanek, Dominika Szydło, Joanna Baran, Julita Bielaniewicz, Marcin Gruza, Arkadiusz Janz, Kamil Kanclerz, and 1 others. 2023. ChatGPT: Jack of all trades, master of none. *Information Fusion*, 99:101861.

Sawan Kumar, Sharmistha Jat, Karan Saxena, and Partha Talukdar. 2019. Zero-shot word sense disambiguation using sense definition embeddings. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 5670–5681.

Woosuk Kwon, Zhuohan Li, Siyuan Zhuang, Ying Sheng, Lianmin Zheng, Cody Hao Yu, Joseph Gonzalez, Hao Zhang, and Ion Stoica. 2023. Efficient memory management for large language model serving with pagedattention. In *Proceedings of the 29th Symposium on Operating Systems Principles*, pages 611–626.

Hanbing Liu, Haoyang Li, Xiaokang Zhang, Ruotong Chen, Haiyong Xu, Tian Tian, Qi Qi, and Jing Zhang. 2025. Uncovering the impact of chain-of-thought reasoning for direct preference optimization: Lessons from text-to-SQL. In *Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics*, pages 21223–21261.

Ilya Loshchilov and Frank Hutter. 2017. Decoupled weight decay regularization. *arXiv preprint arXiv:1711.05101*.

Marco Maru, Simone Conia, Michele Bevilacqua, and Roberto Navigli. 2022. Nibbling at the hard core of Word Sense Disambiguation. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics*, pages 4724–4737.

Domenico Meconi, Simone Stirpe, Federico Martelli, Leonardo Lavalle, and Roberto Navigli. 2025. Do Large Language Models Understand Word Senses? In *Proceedings of the 2025 Conference on Empirical Methods in Natural Language Processing*, pages 33885–33904.

George A Miller, Claudia Leacock, Randee Tengi, and Ross T Bunker. 1993. A semantic concordance. In *Proceedings of the 1993 Human Language Technology Workshop*.

Liqiang Ming, Sheng-hua Zhong, and Yuncong Li. 2025. Towards general-domain word sense disambiguation: Distilling large language model into compact disambiguator. In *Proceedings of the 2025 Conference on*

Empirical Methods in Natural Language Processing, pages 884–897.

Andrea Moro and Roberto Navigli. 2015. Semeval-2015 task 13: Multilingual all-words sense disambiguation and entity linking. In *Proceedings of the 9th International Workshop on Semantic Evaluation*, pages 288–297.

Roberto Navigli. 2009. Word sense disambiguation: A survey. *ACM Computing Surveys*, 41(2):1–69.

Roberto Navigli. 2026. Is word sense disambiguation dead in the llm era? In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 40, pages 39753–39762.

Roberto Navigli, David Jurgens, and Daniele Vannella. 2013. Semeval-2013 task 12: Multilingual word sense disambiguation. In *Proceedings of the 7th International Workshop on Semantic Evaluation*, pages 222–231.

Long Ouyang, Jeffrey Wu, Xu Jiang, Diogo Almeida, Carroll Wainwright, Pamela Mishkin, Chong Zhang, Sandhini Agarwal, Katarina Slama, Alex Ray, and 1 others. 2022. Training language models to follow instructions with human feedback. *Advances in neural information processing systems*, 35:27730–27744.

Francesco Periti, David Alfter, and Nina Tahmasebi. 2024. Automatically generated definitions and their utility for modeling word meaning. In *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing*, pages 14008–14026.

Francesco Periti, Roksana Goworek, Haim Dubossarsky, and Nina Tahmasebi. 2025. Definition generation for word meaning modeling: Monolingual, multilingual, and cross-lingual perspectives. In *Proceedings of the 2025 Conference on Empirical Methods in Natural Language Processing*, pages 26015–26035.

Sameer Pradhan, Edward Loper, Dmitriy Dligach, and Martha Palmer. 2007. Semeval-2007 task-17: English lexical sample, SRL and all words. In *Proceedings of the 4th International Workshop on Semantic Evaluations*, pages 87–92.

Zhenting Qi, Fan Nie, Alexandre Alahi, James Zou, Himabindu Lakkaraju, Yilun Du, Eric Xing, Sham Kakade, and Hanlin Zhang. 2025. Evolm: In search of lost language model training dynamics. *Preprint*, arXiv:2506.16029.

Qwen, An Yang, Baosong Yang, Beichen Zhang, Binyuan Hui, Bo Zheng, Bowen Yu, Chengyuan Li, Dayiheng Liu, Fei Huang, Guanting Dong, Haoran Wei, Huan Lin, Jian Yang, Jianhong Tu, Jianwei Zhang, Jianxin Yang, Jiaxin Yang, Jingren Zhou, and 26 others. 2024. Qwen2.5 technical report. *ArXiv*, abs/2412.15115.

Rafael Rafailov, Joey Hejna, Ryan Park, and Chelsea Finn. 2024. From r to Q^*: Your language model is secretly a q-function. *arXiv preprint arXiv:2404.12358*.

Rafael Rafailov, Archit Sharma, Eric Mitchell, Christopher D Manning, Stefano Ermon, and Chelsea Finn. 2023. Direct preference optimization: Your language model is secretly a reward model. In *Proceedings of the 37th International Conference on Neural Information Processing Systems*, pages 53728–53741.

Alessandro Raganato, Jose Camacho-Collados, and Roberto Navigli. 2017. Word sense disambiguation: A unified evaluation framework and empirical comparison. In *Proceedings of the 15th Conference of the European Chapter of the Association for Computational Linguistics*, pages 99–110.

Bianca Scarlini, Tommaso Pasini, and Roberto Navigli. 2020. With more contexts comes better performance: Contextualized sense embeddings for all-round word sense disambiguation. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing*, pages 3528–3539.

Benjamin Snyder and Martha Palmer. 2004. The english all-words task. In *Proceedings of the 3rd International Workshop on the Evaluation of Systems for the Semantic Analysis of Text*, pages 41–43.

Ying Su, Hongming Zhang, Yangqiu Song, and Tong Zhang. 2022. Rare and zero-shot word sense disambiguation using Z-reweighting. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics*, pages 4713–4723.

Van-Hien Tran, Raj Dabre, Hour Kaing, Haiyue Song, Hideki Tanaka, and Masao Utiyama. 2025. Exploiting word sense disambiguation in large language models for machine translation. In *Proceedings of the 1st Workshop on Language Models for Low-Resource Languages*, pages 135–144.

Jung H Yae, Nolan C Skelly, Neil C Ranly, and Phillip M LaCasse. 2025. Leveraging large language models for word sense disambiguation. *Neural Computing and Applications*, 37(6):4093–4110.

Hee Suk Yoon, Eunseop Yoon, John Harvill, Sunjae Yoon, Mark Hasegawa-Johnson, and Chang Yoo. 2022. SMSMix: Sense-maintained sentence mixup for word sense disambiguation. In *Findings of the Association for Computational Linguistics: EMNLP 2022*, pages 1493–1502.

Erxin Yu, Jing Li, Ming Liao, Qi Zhu, Boyang Xue, Minghui Xu, Baojun Wang, Lanqing Hong, Fei Mi, and Lifeng Shang. 2025. Self-error-instruct: Generalizing from errors for LLMs mathematical reasoning. In *Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics*, pages 8504–8519.

Junwei Zhang and Xiaolin Li. 2025. Quantum-inspired non-homologous representation constraint mechanism for long-tail senses of word sense disambiguation. In *Proceedings of the AAAI Conference on Artificial Intelligence*, pages 25877–25885.

Kaiyuan Zhang, Qian Liu, Luyang Zhang, Chaoqun Zheng, Shuaimin Li, Bing Xu, Muyun Yang, Xinxiao Qiao, and Wenpeng Lu. 2025a. MADAWSD: Multi-agent debate framework for adversarial word sense disambiguation. In *Proceedings of the 2025 Conference on Empirical Methods in Natural Language Processing*, pages 22294–22313.

Kechi Zhang, Ge Li, Yihong Dong, Jingjing Xu, Jun Zhang, Jing Su, Yongfei Liu, and Zhi Jin. 2025b. CodeDPO: Aligning code models with self generated and verified source code. In *Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics*, pages 15854–15871.

Luyang Zhang, Shuaimin Li, Yishuo Li, Kunpeng Kang, Kaiyuan Zhang, Cong Wang, and Wenpeng Lu. 2025c. RoDEval: A robust word sense disambiguation evaluation framework for large language models. In *Proceedings of the 2025 Conference on Empirical Methods in Natural Language Processing*, pages 17095–17126.

Shengyu Zhang, Linfeng Dong, Xiaoya Li, Sen Zhang, Xiaofei Sun, Shuhe Wang, Jiwei Li, Runyi Hu, Tianwei Zhang, Fei Wu, and Guoyin Wang. 2025d. Instruction tuning for large language models: A survey. *Preprint*, arXiv:2308.10792.

Xulang Zhang, Rui Mao, Kai He, and Erik Cambria. 2023. Neuro-symbolic sentiment analysis with dynamic word sense disambiguation. In *Findings of the Association for Computational Linguistics: EMNLP 2023*, pages 8772–8783.

Yaowei Zheng, Richong Zhang, Junhao Zhang, Yanhan Ye, and Zheyan Luo. 2024. LlamaFactory: Unified efficient fine-tuning of 100+ language models. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics*, pages 400–410.

Han Zhong, Zikang Shan, Guhao Feng, Wei Xiong, Xinle Cheng, Li Zhao, Di He, Jiang Bian, and Liwei Wang. 2024. Dpo meets ppo: Reinforced token optimization for rlhf. *arXiv preprint arXiv:2404.18922*.

A Chain-of-Thought Traces

The prompt used for constructing the disambiguation process, together with representative step-by-step Chain-of-Thought (CoT) traces generated during definition generation, is provided in this section.

A.1 Prompt for CoT Construction

Table 9 shows the prompt template used for CoT construction. In our design, the gold gloss is provided as the target definition during CoT construction. This guidance encourages the model to produce coherent and diverse reasoning paths during sampling, while keeping the final output aligned with the intended sense.

A.2 Constructed Chain-of-Thought Trace

An illustrative CoT trace generated by Qwen2.5-72B-Instruct on a SemCor instance is shown in Table 10. The trace begins by identifying the approximate sense based on contextual cues, followed by a step-by-step analysis of the surrounding context, and finally derives a definition that is consistent with the intended usage in context.

A.3 Samples from the CoT-Enhanced Model

We further provide an example from the 42D split, comparing the responses generated in the *disambiguation-guided supervised fine-tuning* stage and in the *preference-based optimization* stage. The outputs from the SFT model and the SFT+DPO model are shown in Table 11. In this example, the SFT+DPO model produces a more complete reasoning process, in which the model considers the syntactic role of the target word and integrates relevant contextual and background knowledge, thereby correcting the shallow and surface-level interpretation produced by the SFT model.

B Training Data Details

In this section, we report the data scale and construction details used in the *disambiguation-guided supervised fine-tuning* and *preference-based optimization* stages, including the number of CoT traces generated and the effect of quality filtering.

B.1 Disambiguation-Guided Supervised Fine-Tuning Data

The SemCor dataset contains 226,036 annotated target-word instances. For each instance, we sample K CoT traces from the CoT constructor, resulting in an initial CoT-augmented corpus of 1,356,216 synthetic instances. K is set to 6.

However, although the gold gloss is provided as supervision, the sampled CoT traces vary in reasoning quality. We therefore compute a DGQS score for each generated trace and apply quality-based filtering. As shown in Figure 4, the score distribution exhibits a clear drop beyond 0.8, so we retain only traces with DGQS ≥ 0.8.

After filtering, 1,073,399 traces (79.2% of the generated corpus) are retained as high-quality CoT supervision and constitute the final training set used for supervised fine-tuning.

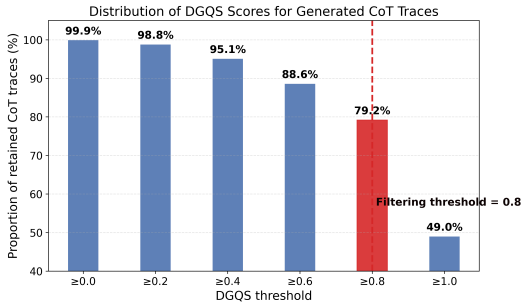


Figure 4: Distribution of DGQS scores over all sampled CoT traces. The proportion of retained traces decreases gradually below 0.8, but drops sharply beyond this point. We therefore retain only CoT traces with DGQS $\geqslant 0.8$ for training.

B.2 Preference-Based Optimization Data

In the preference optimization stage, we construct preference pairs from the supervised fine-tuned model. For each instance in SemCor, we sample $N = 12$ CoT traces from the SFT model. For every sampled trace, we compute its DGQS score with respect to the gold gloss, and we construct a preference pair by selecting the highest-scoring trace as the preferred response and the lowest-scoring trace as the dispreferred one. In principle, this procedure yields 226,036 preference pairs.

However, the effective number of training pairs is smaller and depends on the backbone model. During pair construction, we discard instances whose sampled traces exhibit only negligible DGQS differences, since such cases provide weak or noisy preference signals for optimization. After filtering, we obtain 221,519 preference pairs for LLaMA3.2-3B-Instruct and 208,008 pairs for Qwen2.5-3B-Instruct, which serve as the final datasets used for DPO training.

B.3 Prompt Templates Used in SFT and DPO Training

The same input structure is used across the *disambiguation-guided supervised fine-tuning* (SFT) stage and the *preference-based optimization* (DPO) stage. In the SFT stage, the model is trained to generate a full CoT trace under this prompt format. In the DPO stage, the same prompt is reused to sample multiple CoT outputs from the SFT model, from which preference pairs are constructed.

Using a unified prompt specification ensures that the model is trained and optimized under a consistent input setting rather than benefiting from prompt engineering differences. The prompt formats for SFT and DPO are shown in Tables 5 and 6.

Role	Content
User	Please determine the correct definition of the target word in the context. Context: `<context text>` Target word: `<target word>`
Assistant	`<filtered CoT trace>`

Table 5: Prompt template used for SFT training.

Role	Content
User	Please determine the correct definition of the target word in the context. Context: `<context text>` Target word: `<target word>`
Chosen Assistant	`<positive CoT trace>`
Rejected Assistant	`<negative CoT trace>`

Table 6: Prompt template used for DPO preference-pair construction.

C Direct Preference Optimization

This section outlines the theoretical framework of Direct Preference Optimization (DPO), specifically illustrating the mechanism by which it implicitly enforces a regularization constraint to prevent the optimized policy from deviating excessively from the reference model.

C.1 Learning Objective

In the standard formulation of Reinforcement Learning from Human Feedback (RLHF) that includes a Kullback-Leibler (KL) divergence constraint, the training objective is expressed as (Ouyang et al., 2022):

$$\max_{\pi_\theta} \mathbb{E}_{x\sim\mathcal{D}, y\sim\pi_\theta(y|x)}[r_\phi(x,y)] - \beta\mathbb{D}_{\mathrm{KL}}[\pi_\theta(y \mid x) \,\|\, \pi_{\mathrm{ref}}(y \mid x)] \tag{4}$$

In this equation, π_{ref} serves as the anchor or initial model distribution, π_θ is the policy currently being optimized, and r_ϕ acts as the explicitly parameterized reward model.

DPO elegantly bypasses the need for an explicit reward model by mathematically reparameterizing the objective. By expressing the optimal policy directly as a function of the reward, the optimization problem transforms into the following differentiable loss function:

$$\mathbb{E}_{(x,y_w,y_l)\sim\mathcal{D}}\left[\log\sigma\left(\beta\log\frac{\pi_\theta(y_w \mid x)}{\pi_{\mathrm{ref}}(y_w \mid x)} - \beta\log\frac{\pi_\theta(y_l \mid x)}{\pi_{\mathrm{ref}}(y_l \mid x)}\right)\right] \tag{5}$$

Within this loss formulation, y_w and y_l correspond to the preferred and dispreferred sequences, respectively. The coefficient β dictates the severity of the KL penalty. Through this setup, DPO relies solely on a dataset of pairwise preferences, permitting the policy to be aligned via standard supervised fine-tuning while matching the effectiveness of traditional RLHF.

C.2 Implicit Reward

A key property of DPO is that the generative policy inherently acts as its own reward model. For any specific input-output sequence (x, y), the equivalent implicit reward is calculated as:

$$r_\theta(x,y) = \beta\log\frac{\pi_\theta(y \mid x)}{\pi_{\mathrm{ref}}(y \mid x)} \tag{6}$$

During the optimization process, improving the policy π_θ is mathematically equivalent to refining this implicit reward function. Because the formulation computes the reward based on the log-probability ratio, π_{ref} acts as a persistent denominator. Consequently, the model is inherently penalized if π_θ shifts too far from π_{ref}. This structural characteristic perfectly demonstrates how DPO achieves implicit KL regularization without requiring a standalone divergence penalty term.

C.3 Token-level Credit Assignment

While the core DPO formulation evaluates the preference over a complete output sequence, the autoregressive structure of the language model allows this global reward to be mathematically unrolled. Using the chain rule of probability, the sequence-level reward can be factored step-by-step:

$$r_\theta(x, y) = \sum_{t=1}^{T} \beta \log \frac{\pi_\theta(y_t \mid x, y_{1:t-1})}{\pi_{\text{ref}}(y_t \mid x, y_{1:t-1})} \quad (7)$$

This step-wise breakdown enables the derivation of localized reward signals for individual tokens. Even though DPO is traditionally supervised using only full-sequence labels, recent studies indicate that the model learns to distribute these credit signals compositionally to crucial tokens during the generation trajectory (Rafailov et al., 2024). Consequently, this implicit credit assignment produces dense, token-level reward estimations that can be exploited for subsequent fine-grained alignment stages (Zhong et al., 2024).

D Baseline Paradigm Details

For the *base* paradigm, we directly prompt the original instruction-tuned LLMs to generate sense definitions, using prompt templates consistent with the RoDEval framework (Zhang et al., 2025c), without any task-specific fine-tuning. This setting reflects the intrinsic sense-reasoning and definition-generation capability of pretrained LLMs.

For the *Direct SFT* paradigm, we re-implement the standard supervised fine-tuning strategy under our dataset and evaluation setup, serving as a strong baseline. In this paradigm, the model is trained to directly map a target word in context to its gold gloss, without explicit disambiguation reasoning or preference alignment.

Meanwhile, we additionally reference a family of dictionary-trained definition-generation models that follow the same direct generation paradigm but adopt two representative strategies for mitigating performance degradation on LFS and zero-shot settings. The first strategy seeks to *expand lexical coverage* by exposing the model to a broader range of sense inventories and definitional formulations. LLaMA2-Dictionary and LLaMA3-Dictionary (Periti et al., 2024) are trained on large-scale heterogeneous lexical resources, including Oxford Dictionary, WordNet, and Wiktionary, thereby enriching the distribution and diversity of definitional knowledge available during supervised training. The second strategy instead focuses on *expanding linguistic coverage* in order to enhance cross-lingual robustness and sense generalization. The *LLaMADictionary* variants (Periti et al., 2025) are trained exclusively on Wiktionary but differ in linguistic scope: *LlamaDictionary-en* uses English-only supervision, whereas *LlamaDictionary-sv-de-en* and *LlamaDictionary-it-es-pl-sv-ru-de-fr-en* progressively incorporate multilingual training signals, enabling definitional transfer across languages.

Taken together, these models constitute alternative approaches within the direct generation paradigm that alleviate the long-tail and unseen-sense problem through *data-scale and language-scope expansion*, in contrast to our method, which instead improves sense generalization by restructuring the learning process itself.

E Implementation Details

All experiments are conducted on a server equipped with four NVIDIA GeForce RTX 4090 GPUs, each with 24GB of memory. Training is carried out using the Llama Factory framework (Zheng et al., 2024) with FlashAttention 2.0 (DeepSeek-AI et al., 2024), while inference is performed using vLLM (Kwon et al., 2023). To ensure computational efficiency, all reported results are obtained from a single run.

We fine-tune LLaMA3.2-3B-Instruct and Qwen2.5-3B-Instruct under two training paradigms, namely Direct SFT and CoT-enhanced SFT, using LoRA (Hu et al., 2021) for parameter-efficient adaptation. The optimizer is AdamW (Loshchilov and Hutter, 2017) with default settings. A cosine-decay learning-rate schedule with linear warmup over the first 5% of training steps is applied, and the context window is fixed to 1024 tokens.

Across all training phases, we use a global batch size of 64. The LoRA configuration is kept consistent, with rank 8, $\alpha = 16$, and target modules `q_proj`, `v_proj`, `k_proj`, `o_proj`, `gate_proj`, `up_proj`, and `down_proj`. The learning rates for the SFT and DPO phases are set to 1×10^{-4} and 1×10^{-6}, respectively.

Models are trained for 4 epochs during the SFT stage, and the final checkpoint is selected as the reference model for DPO. Training then continues for 6 additional epochs in the DPO stage. During

Preference Pair Construction, the sampling N is set to 12, with temperature $= 1.0$

F Extended Baselines and Ablation Studies

To thoroughly validate the effectiveness of our proposed WSDPO framework and address the impact of individual components, we conduct a series of extended baseline comparisons and an ablation study on the source of reasoning traces.

F.1 Effectiveness of Explicit CoT Reasoning

To verify the intrinsic value of the two-stage reasoning process independent of our fine-tuning framework, we evaluate the unfinetuned base models using our disambiguation-aware CoT prompts. As shown in the top block of Table 7, applying CoT prompting consistently outperforms the standard zero-shot baseline across most metrics, particularly for the LLaMA backbone. This confirms that the "disambiguation-then-generation" paradigm is inherently superior to direct generation, thereby validating our motivation to explicitly distill this reasoning capability into the models.

F.2 Comparison with Large-Scale LLMs

To establish a performance upper bound, we further compare our 3B WSDPO model against massive unfinetuned models, specifically Qwen2.5-72B-Instruct and DeepSeek-V3.2. As presented in the middle block of Table 7, while implicit semantic understanding naturally scales with model size, our WSDPO (3B) model significantly closes the performance gap and even surpasses these 70B+ giant models on specific unseen and long-tail tasks (e.g., LFS and ALL). This demonstrates that our training paradigm efficiently distills disambiguation capabilities, enabling a smaller model to generalize better to complex scenarios than much larger generic models.

Model / Setting	LFS	0-lex	0-lex-def	0-def	42D	ALL
Base Models (Zero-shot vs. CoT)						
Qwen2.5-3B (Zero-shot)	53.76	67.09	51.99	56.64	54.38	58.37
Qwen2.5-3B (CoT)	54.56	63.91	52.75	57.58	57.64	57.67
LLaMA3.2-3B (Zero-shot)	51.80	64.47	51.99	56.64	56.23	58.69
LLaMA3.2-3B (CoT)	56.94	67.35	54.64	60.91	60.37	61.83
Large-Scale Models (Upper Bound)						
Qwen2.5-72B (Zero-shot)	60.99	72.65	**59.43**	**66.18**	62.97	67.03
DeepSeek-V3.2 (Zero-shot)	61.17	72.38	59.09	66.15	**66.08**	67.26
WSDPO (Ours, 3B)	**71.00**	**72.70**	56.57	64.39	60.34	**82.32**

Table 7: DGQS comparison with extended baselines and large-scale LLMs. "CoT" denotes zero-shot CoT prompting on unfinetuned base models.

F.3 Ablation Study

Finally, we investigate the impact of the CoT data source by comparing traces distilled from the 72B teacher model against those generated via rejection sampling from the 3B base model itself. As shown in Table 8, self-sampling yields inferior performance, particularly on less frequent senses (LFS). This indicates that the smaller model struggles to autonomously generate high-quality, hallucination-free reasoning paths for difficult instances. Consequently, distilling CoT traces from a stronger teacher model is crucial to ensuring the quality and reliability of the reasoning signals.

CoT Source	LFS	0-lex	0-lex-def	0-def	42D	ALL
Self-Sampling (3B)	68.71	71.25	**56.78**	63.58	**60.59**	78.67
Distillation (72B)	**71.00**	**72.70**	56.57	**64.39**	60.34	**82.32**

Table 8: Ablation on the source of CoT traces for the LLaMA3.2-3B model: Self-sampling vs. Distillation.

G Additional Analysis on Unseen Settings

We further analyze model behavior on unseen word and unseen definition evaluation splits, including 0-lex, 0-lex-def, 0-def, and 42D. For each split, Figures 5(a)-5(d) report the proportion of Memorized Definition Recall and Novel Definition Generation, as well as the average DGQS within each category.

Across these unseen settings, Direct SFT produces a larger proportion of memorized outputs, while our CoT-enhanced SFT generates more novel definitions with higher DGQS. This shows that our method relies less on recalling previously seen glosses and is better able to infer the intended sense from context when the word or the definition does not appear in the training data.

For the regular dataset ALL (Figure 5(e)), our method still generates fewer memorized outputs and yields higher DGQS for novel definitions compared with Direct SFT. This suggests that the model does not simply repeat stored glosses even when the gold definition exists in the training corpus, but instead produces contextually appropriate definitions.

Overall, the results on unseen settings and the regular setting are consistent with the LFS findings, and provide further support that our training framework improves sense disambiguation rather than encouraging memorized gloss reproduction.

System

You are a senior linguist who specializes in generative word sense disambiguation.
Given a natural language context containing a target word and a reference definition representing the target word's correct sense,
your task is to infer the meaning of the target word in that specific context.
You must then generate a clear and natural definition of the target word that fits the context,
along with a detailed disambiguation process outlining how you resolved any ambiguity in the target word's meaning based on the context.

The reference definition is provided solely to support your internal reasoning,
but you must not mention, cite, or refer to it in your process.
Your final definition must be exactly the same as the reference definition or convey the same meaning.

To facilitate definition extraction using regular expressions, please express your final definition in the following Markdown format:

"'Definition
the concise definition of the target word in this context, no extra words
"'

User

Context:
Target word:
Reference Definition:

Table 9: Prompt given to CoT construction.

Context: Do you measure its relation to reduced absenteeism, turnover, accidents, and grievances, and to improved quality and output?
Target word: output
Reference Definition: production of a certain amount

Constructed Chain-of-Thought Trace
In the given context, the target word "output" is used in a professional or business setting, where it is being compared to other factors such as absenteeism, turnover, accidents, and grievances. The term "output" in this context is likely referring to the production or the amount of work produced by a system or organization.

To disambiguate the meaning of "output" in this specific context, we need to consider the following:

1. **Contextual Clues**: The context involves the measurement of various aspects of organizational performance, including "improved quality and output." This suggests that "output" is a measure of productivity or production.

2. **Collocation**: The word "output" is often used in business and economic context to refer to the production of goods or services. The collocation with "improved quality" further reinforces the idea that "output" is related to the quantity of production.

3. **Logical Inference**: In a business setting, "output" is typically used to describe the result of a production process. When discussing improvements, it makes sense to consider "output" as the amount of goods or services produced.

Given these points, the target word "output" in this context can be clearly defined as:

‘‘‘Definition
production of a certain amount
’’’

Table 10: Example of Qwen2.5-72B-instruct generated CoT for an instance of SemCor train set.

Question: Please determine the correct definition of the target word in the context.
Context: The next felltop reached is Hugh Seat, 2257 feet, the turning point of the walk: high ground continues but then gradually *declines* into Wensleydale.
Target word: *declines*

CoT-enhanced SFT

In the given context, the target word ""declines"" is used to describe the gradual reduction in elevation as one moves from the high ground to the Wensleydale area. The context suggests a continuous path that starts at a high point and then gradually decreases in height.
To disambiguate the meaning of ""declines"" in this context, we can consider the following points:
1. **Elevation Change**: The context mentions ""high ground"" and then ""gradually declines,"" indicating a change in elevation.

2. **Directional Movement**: The term ""declines"" suggests a downward movement, which aligns with the idea of the path getting lower.

3. **Gradual Nature**: The word ""gradually"" emphasizes that the decrease in elevation is not sudden but occurs over a period of time.
Given these points, the most appropriate definition of ""declines"" in this context is:

“‘Definition
grow lower (✗)
”’

CoT-enhanced SFT + DPO

In the given context, the word ""declines"" refers to the gradual reduction in elevation as one moves from the high ground of Hugh Seat to the lower ground of Wensleydale. The term ""declines"" indicates a downward movement or a decrease in height. Disambiguation Process:
1. **Context Analysis**: The context describes a geographical feature, specifically the elevation changes as one moves from Hugh Seat to Wensleydale. The key phrase ""high ground continues but then gradually declines"" suggests a transition from a higher point to a lower one.

2. **Semantic Role**: The word ""declines"" is used to describe the change in elevation. It is a verb that indicates a downward movement.

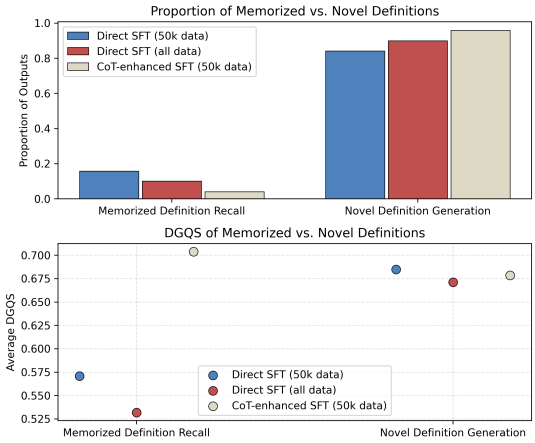
3. **Syntactic Role**: ""Declines"" is the main verb in the clause ""gradually declines into Wensleydale,"" which describes the direction and nature of the change.

4. **Reference to Elevation**: The context clearly states that the elevation is high at Hugh Seat and then reduces, which aligns with the meaning of ""declines"" as a downward movement.

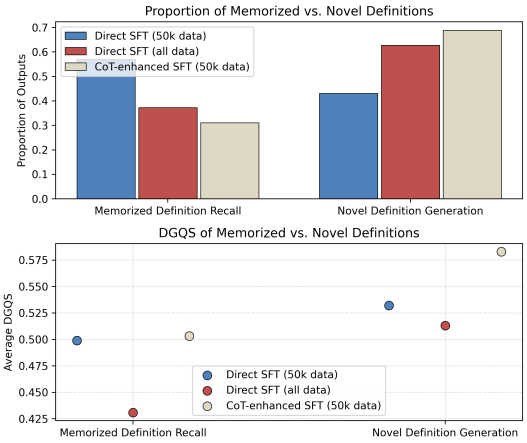
5. **Geographical Context**: The term ""Wensleydale"" is a valley, which is typically lower in elevation compared to the high ground of Hugh Seat. This further supports the interpretation of ""declines"" as a downward movement.
Given this analysis, the definition of ""declines"" in this context is:

“‘Definition
go down in general (✓)
”’

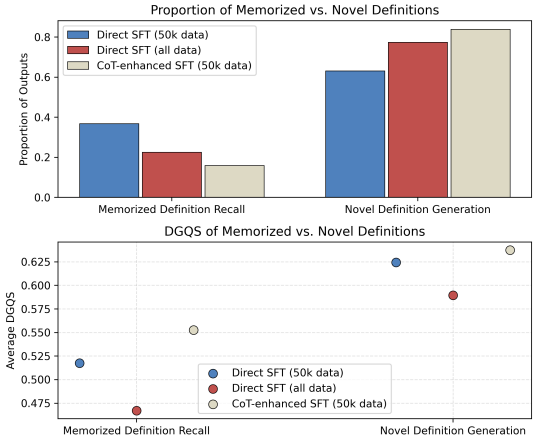
Table 11: Comparison of outputs from the CoT-enhanced SFT model and the CoT-enhanced SFT+DPO model (LLaMA3.2-3B-Instruct) on an example from 42D.



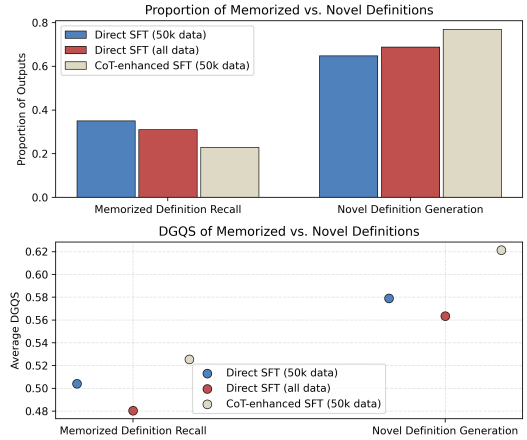
(a) 0-lex: unseen target words



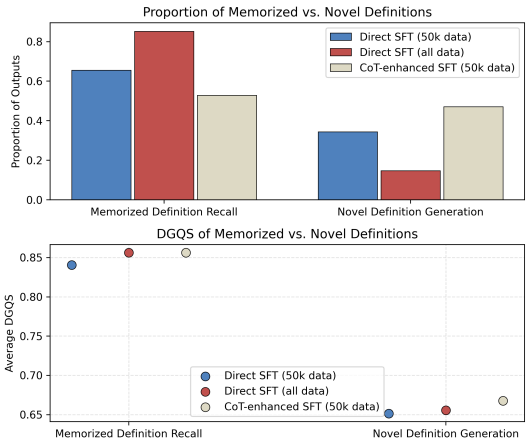
(b) 0-lex-def: unseen gold definitions



(c) 0-def: unseen gloss supervision



(d) 42D: unseen words and unseen definitions



(e) All: most gold definitions are seen during training

Figure 5: Behavioral analysis of memorized definition recall and novel definition generation across different data conditions. (a) 0-lex: target words are unseen during training. (b) 0-lex-def: gold definitions are unseen during training. (c) 0-def: unseen gloss supervision. (d) 42D: both target words and gold definitions are unseen. (e) All: most gold definitions are seen during training and serve as the in-distribution reference setting.