RoDEval: A Robust Word Sense Disambiguation Evaluation Framework for Large Language Models

Luyang Zhang[1]†, Shuaimin Li[2]†, Yishuo Li[1], Kunpeng Zhang[1], Kaiyuan Zhang[1], Cong Wang[1], Wenpeng Lu[1]*

[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 Key Laboratory for High Performance Data Mining, Shenzhen Institutes of Advanced Technology, Chinese Academy of Sciences

luyang.zhang.qlu@gmail.com, wenpeng.lu@qlu.edu.cn

Abstract

Accurately evaluating the word sense disambiguation (WSD) capabilities of large language models (LLMs) remains challenging, as existing studies primarily rely on single-task evaluations and classification-based metrics that overlook the fundamental differences between generative LLMs and traditional classification models. To bridge this gap, we propose **RoDEval**, the first comprehensive evaluation framework specifically tailored for assessing LLM-based WSD methods. RoDEval introduces four novel metrics: Disambiguation Scope, Disambiguation Robustness, Disambiguation Reliability, and Definition Generation Quality Score, enabling a multifaceted evaluation of LLMs' WSD capabilities. Experimental results using RoDEval across five mainstream LLMs uncover significant limitations in their WSD performance. Specifically, incorrect definition selections in multiple-choice WSD tasks stem not from simple neglect or forget of correct options, but rather from incomplete acquisition of the all senses for polysemous words. Instead, disambiguation reliability is often compromised by the models' persistent overconfidence. In addition, inherent biases continue to affect performance, and scaling up model parameters alone fails to meaningfully enhance their ability to generate accurate sense definitions. These findings provide actionable insights for enhancing LLMs' WSD capabilities. The source code and evaluation scripts are open-sourced at https://github.com/DayDream405/RoDEval.

1 Introduction

Large language models (LLMs) have demonstrated remarkable progress across numerous domains (Hurst et al., 2024), attracting widespread attention and systematic evaluations in areas such as Reasoning, Code Generation, and Summarization (Chu et al., 2024; Liu et al., 2023; Song et al., 2024). Word Sense Disambiguation (WSD), a long-standing and critical task in natural language processing, aims to determine the correct meaning of a polysemous word within a given context. Recent studies highlight the significant potential of LLMs in addressing this challenge (OpenAI, 2023; Kalyan, 2024; Wu et al., 2024). However, most existing evaluations of WSD overlook the fundamental differences between LLMs and traditional classification models, relying instead on simplistic or classification-oriented evaluation methods (Kibria et al., 2024; Yae et al., 2024; Sumanathilaka et al., 2024). Such approaches fail to comprehensively evaluate the disambiguation capabilities of different LLM-based WSD methods.

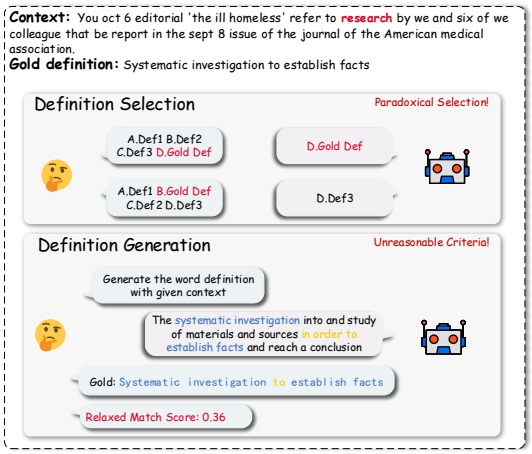


Figure 1: Challenges in evaluating LLMs' WSD capabilities. In Definition Selection, the LLM provides different answers due to variations in the order of options; in Definition Generation, although the generated definition is semantically similar to the gold answer, it is given an unreasonable relaxed matching score.

Specifically, unlike traditional classification models, the WSD performance of LLMs is highly sensitive to prompt formats (Sclar et al., 2023). Moreover, while multiple-choice question answering (MCQA) provides a simple yet effective means

* Corresponding author
† Equal contribution.

Proceedings of the 2025 Conference on Empirical Methods in Natural Language Processing, pages 17084–17115
November 4-9, 2025

of evaluating WSD capabilities, recent studies have shown that some LLMs exhibit inherent selection biases (e.g., a preference for specific option IDs), which, if unaddressed, can compromise the robustness of the evaluation (Zheng et al., 2024). Additionally, given the generative nature of LLMs, a comprehensive assessment requires multifaceted metrics that go beyond MCQA accuracy alone. Framing WSD purely as a classification task may misleadingly suggest that fine-tuned, state-of-the-art (SOTA) classification models outperform general-purpose LLMs such as ChatGPT, despite the latter's broader linguistic competence (Kibria et al., 2024). Finally, the risk of data contamination poses further challenges: since LLMs are typically pretrained on web-scale corpora, their performance on public benchmark datasets may be artificially inflated (Sainz et al., 2023; Balloccu et al., 2024; Xu et al., 2024), thereby undermining the reliability of evaluation results.

To address these challenges and establish a more robust evaluation of LLMs' WSD capabilities, we propose the **RoDEval** Framework, a **Ro**bust Word Sense **D**isambiguation **Eval**uation Framework for LLMs. Our framework begins by systematically evaluating each model's MCQA performance of WSD across diverse prompt templates, identifying the most effective prompt formulation for each individual model to ensure optimal disambiguation performance.

Building on this foundation, we introduce four novel evaluation metrics that collectively provide a comprehensive assessment of LLMs' WSD abilities: (1) **Disambiguation Scope** (DS): The proportion of words a model can disambiguate successfully. To rigorously evaluate a model's true disambiguation capability, we implement a controlled testing protocol: (i) we systematically constrain correct answers to appear only within specific positional ranges (e.g., first two or first four of options), (ii) vary these positional constraints across multiple trials, and (iii) only credit successful disambiguation when the model consistently identifies the correct sense across all positional variations. This conservative approach ensures reported DS scores reflect position-invariant understanding rather than option-order memorization or selection bias exploitation. (2) **Disambiguation Robustness** (DRoS): While our prompt selection and controlled testing protocol mitigates the impact of biases, these inherent model preferences persist as latent factors that may influence real-world disambiguation performance. We therefore argue that robustness against such confounding factors constitutes an essential dimension of WSD capability. The DRoS quantifies this aspect by measuring performance variance across: (i) Prompt Template Variations: multiple of prompt templates, and (ii) Option Sequence Configurations: different positional arrangements of correct answers. (3) **Disambiguation Reliability** (DreS): Unlike traditional classification models that output discrete labels, LLMs prefer to generate elaborate and often convincing explanations for their disambiguation. While this fluency enhances perceived trustworthiness (Elangovan et al., 2024), it becomes particularly problematic when models hallucinate—producing confident yet incorrect explanations that are more misleading than a simple "I don't know" response. To quantify this critical aspect of reliability, we propose DreS, which evaluates the alignment between a model's self-assessed confidence and its actual disambiguation accuracy. (4) **Definition Generation Quality Score** (DGQS): As generative language models, LLMs should be evaluated not only on classification-style (e.g., MCQA) WSD tasks, but also on their ability to generate high-quality sense definitions. Traditional metrics for generative tasks like relaxed matching F1, BLEU, and ROUGE fall short for this purpose, as they overly penalize lexical diversity while rewarding superficial n-gram overlaps (Papineni et al., 2002; Schluter, 2017). And semantic similarity metrics, such as BERTScore (Zhang* et al., 2020), suffer from a gold-standard bottleneck, which assumes a single "correct" definitional formulation, disregarding the inherent diversity of natural language, underestimating LLMs' definition generation capability. To address this gap, we propose DGQS which is based on semantic similarity and additionally incorporates semantic richness as a correction.

Overall, our key contributions include:

- We propose the first WSD assessment framework specifically designed for LLMs, **RoDEval**, introducing four new metrics (**DS**, **DRoS**, **DReS**, **DGQS**), robustly evaluating the WSD capability of LLMs.
- Through systematic evaluation of mainstream LLMs (including four open-source models and a closed-source one), we reveal that despite exhibiting considerable definition generation capability (GDQS > 0.66) and disambiguation robustness (DRoS > 0.88), the best-

performing model completely disambiguates merely 58.2% of polysemous words in test sets, with 25% of its confident answers being incorrect.

- We conducted additional experiments to investigate the limitations of model performance, analyzing observed failure patterns and their potential causes to inform future improvements in LLM-based WSD.

2 Related Work

2.1 WSD Based on Traditional Models

Traditional WSD approaches include knowledge-based (KB) and supervised methods. KB systems (e.g., using WordNet or BabelNet) disambiguate word senses without annotated data (Mizuki and Okazaki, 2023; Kwon et al., 2021), while supervised models leverage labeled corpora and machine learning, often outperforming KB methods (Bevilacqua et al., 2021; Pasini, 2021; Raganato et al., 2017; Vial et al., 2019). Recent work combines both paradigms for improved performance (Huang et al., 2019; Kumar et al., 2019; Bevilacqua and Navigli, 2020; Lu et al., 2019; Zhang et al., 2022). However, these approaches remain limited by their dependence on external knowledge bases and annotated data, struggling with rare or underrepresented word senses.

2.2 WSD Based on LLMs

Recent works have demonstrated the effectiveness of LLMs in addressing key challenges in WSD, particularly in overcoming data scarcity and improving disambiguation performance. For instance, LLM-generated data has been used to create French WSD corpora (Mehdoui et al., 2024) and morpheme-annotated Chinese datasets, with MorBERT (Wang et al., 2024) achieving SOTA results on MiCLS dataset. Beyond data generation, LLMs serve as powerful knowledge augmenters that significantly enhance WSD systems. In visual word sense disambiguation (VWSD), LLMs have proven particularly valuable as contextual knowledge bases, capable of enriching phrase representations even for ambiguous words outside a retrieval module's coverage (Kritharoula et al., 2023). Furthermore, the ARPA (Papastavrou et al., 2024) advances this capability by employing LLMs to generate enhanced word embeddings, establishing new SOTA performance on VWSD.

2.3 WSD Evaluation for LLMs.

Recent studies begin evaluating LLMs' WSD capabilities through various evaluation approaches. A study conducts systematic evaluations showing that while LLMs perform reasonably well on WSD MCQA task, their accuracy consistently falls short of SOTA traditional classification models (Kibria et al., 2024). Other investigation reveals a strong correlation between model parameter size and WSD performance, suggesting scale-dependent capabilities in disambiguation tasks (Yae et al., 2024). Additional work demonstrates that prompt engineering techniques can lead to measurable improvements in WSD performances, though the enhanced results still fall short of traditional classification models (Sumanathilaka et al., 2024).

3 Methods

In this section, we present the four novel metrics of RoDEval framework, explaining how it ensures the robustness of evaluation results for the WSD task. Figure 2 shows the structure of RoDEval framework.

3.1 Disambiguation Scope

To address the fundamental question "Which words can the model disambiguate?", we propose Disambiguation Scope (DS), the proportion of words in a dataset that a model can reliably disambiguate. Unlike traditional metrics that evaluate at the sense level, DS operates at the word level, requiring models to demonstrate comprehensive understanding of all senses for each counted word. We implement DS through Differently Ordered Multiple-Choice Question(DOCQ), a robust evaluation protocol designed to mitigate the selection bias. To be specific, given the set of candidate sense definitions $D = \{d_1, d_2, \ldots, d_n\}$, the correct definition at the i-th ordinal position d_i. The DOCQ executes three key steps: (1) we systematically constrain $i \in [1, m]$, where $m \in \{1, 2, 3, 4\}$. (2) vary m across multiple trials. (3) only credit successful disambiguation when the model consistently identifies the correct option across all positional variations. The Disambiguation Scope DS_θ calculates the proportion of words ($w \in W$) whose disambiguation accuracy exceeds threshold θ across all positional variation experiments:

$$DS_\theta = \frac{|\{w \in W \mid \forall e \in E, \text{Acc}(w, e) \geq \theta\}|}{|W|}, \quad (1)$$

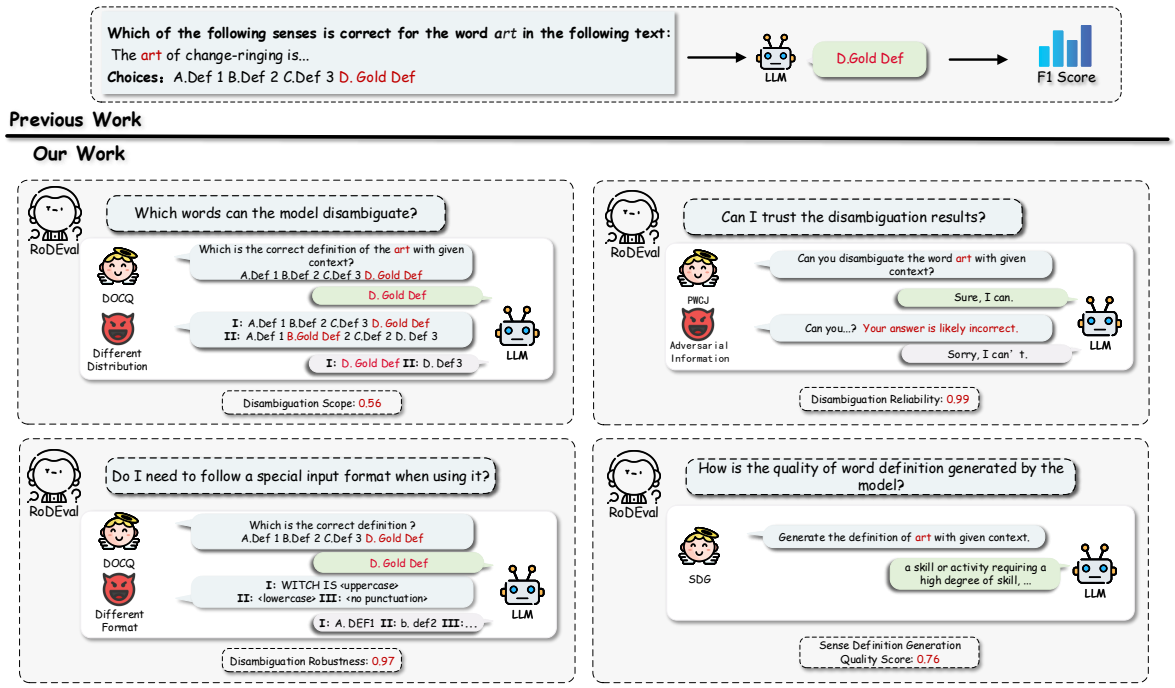


Figure 2: Overview of Evaluation Framework (RoDEval). It assesses the disambiguation scope (DS) through DOCQ task, showed in the first block; it evaluate the disambiguation reliability (DReS) through PWCJ task, showed in the second block; it evaluates the disambiguation robustness (DRoS) through DOCQ task, showed in the third block; and it measures the sense definition generation quality (DGQS) through DG task, showed in the last block.

where $|W|$ represents the total number of words, set E contains all positional configurations, $\text{Acc}(w, e)$ gives the disambiguation accuracy of word w under configuration e. And the threshold $\theta \in \{50, 75, 100\}$ specifies the minimum accuracy required to consider a word as successfully disambiguated. The prompt of DOCQ is illustrated in Figure 20 in the Appendix.

3.2 Disambiguation Robustness Score

While our framework mitigates measurable biases through controlled experimentation, latent model biases persist as inherent characteristics that may affect real-world disambiguation performance. To quantify this sensitivity, we propose the Disambiguation Robustness Score (DRoS), which evaluates performance variation across different prompt templates experiments and DOCQ. The DRoS is computed as:

$$\text{DRoS} = 1 - \sqrt{\frac{1}{N}\sum_{i=1}^{N}(\text{Acc}_\text{i} - \mu_{\text{Acc}})^2}, \quad (2)$$

where $N = |\mathcal{P} + \mathcal{O}|$ denotes the total number of evaluation configurations, with $\mathcal{P}$ representing the set of prompt templates and $\mathcal{O}$ the set of option position arrangements in DOCQ, Acc_i measures the disambiguation accuracy under the i-th experimental configuration and μ_{Acc} denotes average accuracy across all configurations. The details of templates are showed in Figure 4.

3.3 Disambiguation Reliability Score

A model's disambiguation reliability fundamentally depends on its cognitive awareness of its own WSD capabilities. When a model cannot accurately assess whether it can correctly disambiguate a polysemous word, its outputs become inherently untrustworthy. To quantify this critical property, we first design a Polysemous Word Cognition Judgment (PWCJ). The PWCJ task evaluates a model's cognitive awareness of polysemous by presenting a target word in context and explicitly prompting the model to judge whether it can determine the correct sense definition. Then we formally define two key events: (1) Event A: The model produces a correct disambiguation in the DOCQ task, occurring with probability $P(A)$, (2) Event B: The model asserts a positive capability judgment (e.g., "Yes, I can disambiguate this word") in the PWCJ, with probability $P(B)$. We consider the model to be reliable only when it gives a positive judgment and the dis-

ambiguation result is correct, that is, $P(A \mid B)$. The Disambiguation Reliability Score (DReS) is then derived as $P(A \mid B)$, computed via Bayes' Theorem (Joyce, 2003):

$$\text{DReS} = P(A \mid B) = \frac{P(B \mid A)P(A)}{P(B)}, \quad (3)$$

where $P(A)$, $P(B)$ and $P(B \mid A)$ can be obtained from the results of DOCQ and PWCJ. The prompt of PWCJ is showed in Figure 21 in the Appendix.

3.4 Definition Generation Quality Score

To evaluate generative LLMs, we introduce the Definition Generation task (DG): given a polysemous word in context, the model must directly generate its correct sense definition. The prompt of DG is showed in Figure 22 in the Appendix. And to address the limitations of conventional generation metrics in fairly evaluating LLM outputs, we propose the Definition Generation Quality Score (DGQS). This metric combines semantic similarity with a linguistic richness correction, effectively mitigating the underestimation of model's definition generation capability caused by penalizing linguistically valid variations from single gold-standard. The semantic similarity between model-generated definition and gold-standard is computed as :

$$\text{Sim}(e_1, e_2) = \frac{\sum_{i=1}^{d} e_{1i} e_{2i}}{\sqrt{\sum_{i=1}^{d} e_{1i}^2} \sqrt{\sum_{i=1}^{d} e_{2i}^2}}, \quad (4)$$

where e_1 denotes the embedding vector of the model-generated definition, e_2 represents the gold-standard definition embedding. We employ the paraphrase-MiniLM-L6-v2 for vector space mapping, which outperforms commonly used BERT (Reimers and Gurevych, 2019) on sentences and paragraphs.

The linguistic richness score (LRS) of the model-generated definition is computed as:

$$\text{LRS}(D) = \sum_{w \in D} \left(1 - \frac{f_w}{f_{\max}}\right), \quad (5)$$

where D is the generated definition being evaluated, w denotes a content word in D, f_w represents the occurrence frequency of word w across all model-generated definitions, and $f_{\max}$ is the maximum frequency observed in the model-generated definitions. For generative LLMs, it is not only required to distinguish the word senses, but also to generate reasonable definitions based on those senses. Therefore, in DG task, we reframe WSD as a generative task to test the model's ability to generate sense definitions.

Finally, the DGQS of model-generated definition is computed as:

$$\text{DGQS}(D) = \text{Sim}(e_D, e^*) + \alpha(D) \cdot \text{LRS}(D), \quad (6)$$

where e_D and e^* denote embeddings of generated definition D and gold-standard definition respectively. And the adaptive weight $\alpha(D)$ modulates richness contribution:

$$\alpha(D) = |1 - \text{Sim}(e_D, e^*)| \cdot \text{Sim}(e_D, e^*)^2. \quad (7)$$

Figure 23 in the Appendix B shows an example calculated using DGQS.

Model	Instruct.	Datasets		
		Eval07	Eval13	Eval15
LLaMA2-7b	General	0.0578	0.0866	0.0559
	Guided	0.0813	0.0688	0.0613
Human Evaluation		✗	✗	✗
LLaMA3.1-8b	General	0.0720	0.1169	0.0936
	Guided	0.1165	0.1095	0.0982
Human Evaluation		✗	✗	✗
LLaMA3.1-70b	General	0.1212	0.0892	0.0962
	Guided	0.1123	0.1567	0.0989
Human Evaluation		✗	✗	✗
GPT-4o	General	0.0914	0.1152	0.1167
	Guided	0.0820	0.0628	0.0409
Human Evaluation*		✗	✗	✗
DeepSeek-v3	General	0.0949	0.1086	0.1583
	Guided	0.1770	0.1285	0.0940
Human Evaluation		✗	✗	✓

Table 1: This table shows the ROUGE-L difference between General Instruction and Guided Instruction on SemEval2007, SemEval2013 and SemEval2015. A single tick (✓) points to the presence of at least one exact match, while a cross sign (✗) denotes that no exact match. Human Evaluation*: GPT-4o refuse to complete the most of instances under the Instruction Guided, so we alternate to evaluate Instruction General instead.

4 Experiments

4.1 Datasets

We select SemEval2007 (Pradhan et al., 2007), SemEval2013 (Navigli et al., 2013), SemEval2015 (Moro and Navigli, 2015), Senseval2 (Edmonds and Cotton, 2001) and Senseval3 (Snyder and Palmer, 2004) as test datasets. And to avoid potential data contamination, we use a data contamination detection method that leverages the memory

of LLMs (Golchin and Surdeanu, 2024) to evaluate all these datasets. Table 1 shows the part detection results. We find that the possibility of contamination for all tested models was extremely low across these five datasets, thereby substantiating the credibility of our evaluation conclusions. More detection results are showed in Appendix A.1.

4.2 Implementation Details

We select five mainstream large language models for evaluation, which include LLaMA2-7b (Touvron et al., 2023), LLaMA3.1-8b (Dubey et al., 2024), LLaMA3.1-70b-AQLM (Egiazarian et al., 2024; Dubey et al., 2024), DeepSeek-V3 (Liu et al., 2024) and GPT-4o (Hurst et al., 2024). The hyperparameters of all models were set according to the recommendations in the official documentation of each model. All experiments are repeated five times and completed on four NVIDIA GeForce RTX 3090 GPUs.

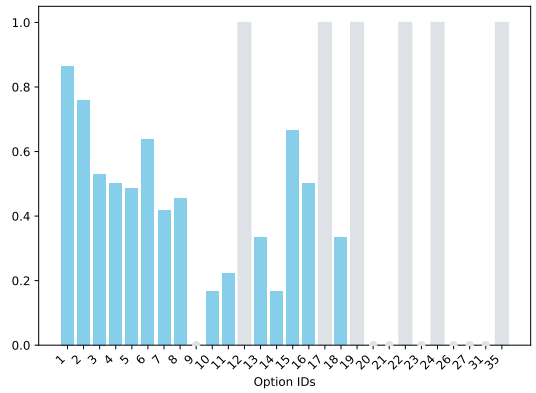


Figure 3: Option recall of DeepSeek-v3 on SemEval2007. Horizontal axis labels are the option ids. Grey denotes options that appear only once.

4.3 Experimental Results

This section first evaluates how prompt formats and selection biases impact WSD performance, empirically demonstrating the necessity of bias mitigation in evaluation protocols. We then present experimental results using our four novel metrics.

4.3.1 Impact of Models' Biases

Format Bias. Our analysis of prompt format bias examines five distinct template variations, as illustrated in Figure 4. The template formats include: standard English grammar with proper capitalization and punctuation (Normal), lowercase formatting (Lower), uppercase formatting (Upper), title case formatting (Title), and punctuation-free formatting (No-punc). Specifically, to avoid introducing new ambiguities, we implement a robust rule set for punctuation-free formatting, with detailed rules described in the Appendix A.2. The results in Table 2 reveal that GPT-4o exhibits varying performance across different prompt formats, with a decline in performance observe in almost all non-standard formats compare to grammatically conventional prompts. This bias likely stems from the model's training data distribution, where well-formed English predominates. **The results of other models show that prompt biases persist across all tested models on WSD.** Consequently, it is imperative to not only mitigate prompt bias in evaluation, but also incorporate robustness assessment when measuring models' WSD capabilities. The results of other models are showed in Appendix A.3.

Dataset	Normal	Lower	Upper	Title	No-punc	σ^2
Eval07	70.22	67.4	65.28	66.23	66.87	2.78
Eval13	80.50	78.79	79.84	79.39	79.34	0.33
Eval15	81.23	82.07	83.50	83.50	82.18	0.78
Eval2	75.67	69.76	70.63	70.49	75.59	6.92
Eval3	72.16	69.89	71.07	69.44	73.99	2.70

Table 2: F1 scores (%) of GPT-4o on DOCQ using four prompt templates.

Selection Bias. Figure 3 reveals selection biases of DeepSeek-v3. It can be observed that after excluding the options with recalls of 0 and 1, the model generally has a higher recall when the correct option ids are 1 and 2. **This indicates that the models also have a "preconceived notion", meaning that incorrect answers located before the correct answer can interfere with the model's judgment.** Specifically, We count the number of options selected by these models showed in Figures 10-14. Upon the models' output, we find that all models except LLaMA2-7b have a preference for option ids 1 and 2. Combining the results of option recall, we conclude that these models tend to choose these two options when they cannot identify the correct answer. This tendency may be related to the structure of the internal classifiers of the models. And Figure 10 show that the number of outputs labeled as "2" from the LLaMA2-7b is disproportionately large. This anomaly may be related to the imbalance distribution of labels in the model's training data. Detail results are showed in Appendix A.4.

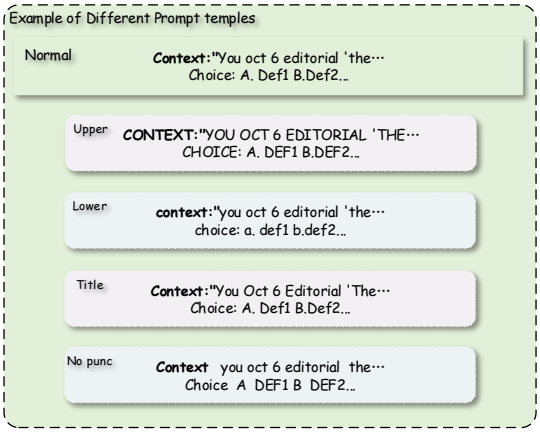


Figure 4: An example of different prompt templates.

4.3.2 Evaluation Results of LLMs' WSD Capabilities

Disambiguation Scope (DS). Table 3 shows the **limited DS** of the tested models. Specifically, GPT-4o, DeepSeek-v3 and LLaMA3.1-70b are able to correctly identify the senses of more than half of the polysemous words. In comparison, LLaMA3.1-8b and LLaMA2-7b can only disambiguate a very small number of words. And the difference between DS_{100} and $F1$ of LLaMA3.1-8b is as high as 0.4, which could lead to completely different evaluation conclusions. Even for DeepSeek-v3, which is least affected by biases, is still 0.1276. Furthermore, we observe that the difference between DS_{100} and DS_{75} is much smaller than the DS_{75} and DS_{50}. **This indicates that if a model has grasped the majority (>75%) of a word's senses, it is more likely to grasp all of this word's senses.** Therefore, when researchers attempt to enhance the model's WSD capabilities, they should strive to enable the model to learn all senses of a word. To further investigate why models select the wrong option, we increase the number of correct options in DOCQ for experimentation in Appendix A.5. The results illustrate that as the number of correct options increases, the model's accuracy rises in line with statistical probability. This suggests that increasing the number of correct options cannot help the model select the correct answer; the model's errors are not due to ignoring or forgetting the correct answer. We make the accuracy of each word of the model publicly available in the code repository.

Disambiguation Reliability Score (DReS). Table 4 shows that **even the best-performing models, DeepSeek-v3 and GPT-4o, have nearly a 30% probability of "lying" in the WSD.** They often appear "**overconfident**" in their responses, which results in lower reliability. This is a dangerous signal, as for models with better performance, providing a wrong answer can have far more serious consequences than simply saying, "Sorry, I don't know." Becoming "self-aware" may be one of the important directions for future models. Additionally, there is a positive correlation between the model's reliability and its parameter count.

Model	Disambiguation Scope DS_{50}	DS_{75}	DS_{100}	F1
LLaMA2-7b	0.0180	0.0055	0.0055	0.3222
LLaMA3.1-8b	0.2212	0.1930	0.1888	0.5889
LLaMA3.1-70b	0.6357	0.5462	0.5300	0.7328
GPT-4o	0.6625	0.5808	0.5641	0.7172
DeepSeek-v3	0.8408	0.7069	0.5816	0.7092

Table 3: DS and typically used F1 of five models. $Scope_\theta$ points to words with an accuracy (%) greater than θ are considered as ones that an LLM can disambiguate.

Model	Metrics	Datasets Eval07	Eval13	Eval15	Eval2	Eval3
LLaMA2-7b	DreS	0.3711	0.5377	0.4682	0.4992	0.4562
	p_A	0.4131	**0.5602**	0.5773	0.5359	0.4551
	p_B	0.6989	0.3223	0.8004	0.8436	0.4324
LLaMA3.1-8b	DreS	0.5647	0.6950	0.5481	0.6264	0.5795
	p_A	**0.5098**	0.6880	0.6703	0.6490	0.5935
	p_B	0.1868	0.8357	0.9354	0.9114	0.9243
LLaMA3.1-70b	DreS	0.6069	0.7931	0.6402	0.7038	0.6732
	p_A	0.6132	0.7883	0.7798	0.7349	0.6903
	p_B	0.9560	0.9908	0.9628	0.9869	0.9346
GPT-4o	DreS	0.7017	0.8178	0.6692	0.7404	0.7343
	p_A	0.6989	0.8187	0.8160	0.7686	0.7308
	p_B	0.8989	0.9014	0.8962	0.9215	0.8870
DeepSeek-v3	DreS	0.7460	0.8207	0.6763	0.7790	0.7244
	p_A	**0.6637**	0.8127	0.8141	**0.7581**	0.7286
	p_B	0.5538	0.8953	0.8131	0.5670	0.8492

Table 4: Reliability of five models. p_A denotes that the proportion of senses the model can disambiguate to the total count., p_B denotes that the proportion of senses that the model believes it can disambiguate out of the total number. Underlined numbers indicate the model is **overconfident** ($p_B - p_A > 0.1$), while bold numbers indicate **a lack of confidence** about their WSD abilities ($p_A - p_B > 0.1$).

Disambiguation Robustness Score (DRoS). Table 5 demonstrates that **while all evaluated LLMs exhibit strong robustness (mean DRoS = 0.94 $\pm$ 0.05), they continue to exhibit prompt and selection biases.** Specifically, GPT-4o and DeepSeek-

v3 showcase the highest DRoS, highlighting its easier to achieve the expected performance through usage. This emphasizes that stronger learning capabilities can help reduce the model's preferences and biases. Moreover, small-parameter models, such as LLaMA2-7b and LLaMA3.1-8b, have limited learning capabilities and can only understand certain fixed prompt formats, as a result of lower DRoS. In most cases, the larger the model's parameter size, the weaker the model's biases and preferences, and the stronger the robustness.

Model	Datasets					
	Eval07	Eval13	Eval15	Eval2	Eval3	AVG.
LLaMA2-7b	0.9138	0.8598	0.8731	0.8769	0.8915	0.8830
LLaMA3.1-8b	0.9331	0.8939	0.9012	0.9125	0.9258	0.9133
LLaMA3.1-70b	0.9545	0.9786	0.9695	0.9674	0.9652	0.9670
GPT-4o	**0.9569**	0.9897	0.9768	0.9775	**0.9726**	**0.9747**
DeepSeek-v3	0.9448	**0.9916**	**0.9847**	**0.9800**	0.9703	0.9743

Table 5: DRoS of five models. Bold numbers indicate the highest DRoS, while underlined numbers indicate the second-highest.

Definition Generation Quality Score (DGQS) Table 6 shows that **once the model's parameter scale reach 70 billion, its word sense generation capability stabilize and it can produce high-quality sense definitions (DGQS > 0.6).** This suggests that the formation of a model's generation capability precedes its WSD ability and requires a smaller parameter scale. **However, continued parameter scaling yields diminishing returns, with DGQS plateauing beyond 70B ($\Delta < 0.03$), demonstrating that subsequent enhancements in WSD capability cannot be achieved through parameter increases alone.**" Additionally, the results illustrate that while relaxed matching F1 can differentiate the quality of generated definitions across models, it significantly underestimates the generative ability for all models. Similarly, ROUGE exhibit the same issue. BLEU not only show minimal differences among models, making it difficult to distinguish their capabilities, but also exhibit significant bias in evaluating generation quality. For instance, LLaMA2-7b achieves at least the second-highest BLEU scores across all datasets. These demonstrate the effectiveness and necessity of DGQS. The rest of results are displayed in Appendix A.6.

4.3.3 Evaluation Results of fine-tuned WSD models

To explore the potential WSD capability improvements brought by fine-tuning methodologies, we additionally select three SOTA models fine-tuned specifically for DG task, namely LLaMA2-Dictionary (Periti et al., 2024), LLaMA3-Dictionary (Periti et al., 2024), and T5-definition (Giulianelli et al., 2023). As shown in Table 6, the fine-tuned models demonstrate significant performance gains compare to their base models, yet still slightly underperform the best-performing LLMs.

Model	Typical Generative Metrics					
	F1	BLEU	R-1.	R-2.	R-l.	DGQS
LLaMA2-7b	0.1655	**0.0138**	0.0947	0.0112	0.0840	0.4835
LLaMA3.1-8b	0.2140	0.0110	0.1534	0.0273	0.1427	0.5616
LLaMA3.1-70b	0.2733	0.0128	0.2121	0.0440	0.1932	0.6207
GPT-4o	0.2755	0.0126	0.2331	**0.0644**	0.2139	0.6492
DeepSeek-v3	**0.3056**	0.0128	**0.2354**	0.0561	**0.2142**	**0.6510**
*LLaMA2-Dictionary	0.2055	0.0120	0.1728	0.0465	0.1622	0.6003
*LLaMA3-Dictionary	0.2216	0.0123	0.1526	0.0341	0.1412	0.5756
*T5-definition	0.2712	0.0083	0.1236	0.0398	0.1210	0.5481

Table 6: DGQS and typical Generative metrics of five models on SemEval2007. F1 represents the F1 score for relaxed matching, while R-1, R-2, and R-L are ROUGE-1, ROUGE-2, and ROUGE-L, respectively. Bold numbers indicate the highest score, while underlined numbers indicate the second-highest. * indicates fine-tuned models.

Furthermore, we observe that the fine-tuned models exhibit notably anomalous behavior in that they struggle to recognize other task requirements or even different prompt formats. To investigate the underlying cause of this phenomenon, we conduct detailed experiments. Our investigation into this phenomenon reveals that fine-tuning models for specific WSD tasks, such as definition generation, inevitably leads to degradation in their core disambiguation capability. And the system prompts used during fine-tuning not only fail to significantly enhance model performance but may even have detrimental effects. Moreover, even when evaluate exclusively on the DG task, the fine-tuned models exhibit a decline in capabilities that is obscured by superficial performance metrics. We hypothesize that this issue stems from an overemphasis on generation steps at the expense of disambiguation fundamentals during training data construction. This finding highlights a critical challenge that must be addressed when employing fine-tuning to enhance model performance on definition generation tasks in future work. Detailed experimental procedures are provided in Appendix A.7.

4.4 Error Analysis

To systematically investigate the model's failure patterns, we conduct a detailed analysis of poly-

semous words in the DeepSeek-v3 DOCQ experiments, stratifying the analysis by occurrence frequency, number of candidate senses, and proportion of part-of-speech (POS) composition; the analysis is further stratified by disambiguation accuracy, categorizing words into high-accuracy (>0.75) and low-accuracy (<0.25) groups. Table 7 reveals that the high-accuracy word demonstrate significantly higher occurrence frequencies compared to the low-accuracy word. It should be noted that these datasets exclusively comprises authentic natural language texts, implying that the observed word frequencies in the datasets reflect real-world usage patterns to a considerable extent. This finding suggests the model exhibits stronger disambiguation capabilities for high-frequency words. Furthermore, the model achieves more accurate disambiguation for words with fewer candidate senses.

	Frequency	Sense Number	POS. Composition
High-Accuracy	1.87	6.18	'n': 0.47, 'v': 0.4, 'a': 0.03
Low-Accuracy	1.31	7.93	'n': 0.46, 'v': 0.39, 'a': 0.03

Table 7: Comparison of Linguistic Features Between High-Accuracy (>0.75) and Low-Accuracy (<0.25) Words in DOCQ Experiments. Where n denotes nouns, v for verbs, and a for adjectives.

Since no discernible patterns emerge in the POS composition analysis, we investigate further by computing accuracy for each POS category. Table 8 reveals that the model's disambiguation accuracy for verbs is notably inferior to that for other parts of speech. This performance gap may stem from verbs' more complex morphological variations and flexible usage patterns in natural language. Future improvements in LLMs' WSD capabilities should prioritize enhanced verb handling.

POS.	Noun	Verb	Adjective
Acc.	0.8177	0.7179	0.8212

Table 8: POS. Specific Disambiguation Accuracy in DOCQ Experiments (DeepSeek-v3)

5 Conclusion

This paper proposes the first evaluation framework specifically designed for evaluate word sense disambiguation (WSD) capabilities in large language models (LLMs). This framework comprehensively considers the distinctive characteristics of LLMs and their differences from traditional classification models, effectively eliminating interference from model biases in experimental results. It provides a holistic evaluation of LLMs' WSD capabilities across four key dimensions: disambiguation scope, disambiguation reliability, disambiguation robustness, and sense definition generation abilities.

The results demonstrate that, despite exhibiting considerable definition generation capability (DGQS > 0.66) and disambiguation robustness (DRoS > 0.88), the best-performing model achieves complete disambiguation for merely 58.2% of polysemous words in test sets, with 25% of its confident answers being incorrect. This indicates significant limitations in current LLMs' WSD capabilities.

Furthermore, we observe that increasing the number of correct options in MCQA tasks does not substantially improve the model's disambiguation scope, suggesting that incorrect selections do not result from simple oversight or forgetting of correct options. While the models show high disambiguation robustness, inherent model biases continue to affect WSD performance (DRoS < 1). Moreover, models often display overconfident, leading to low disambiguation reliability, which means they are more prone to generating misleading answers when conducting WSD tasks. Notably, when model parameters exceed 70B, further parameter increases do not yield significant improvements in sense definition generation capability, suggesting that subsequent enhancements in generation ability may not be achievable simply through parameter scaling.

Limitations

Our study has several limitations that warrant discussion. First, the evaluation datasets used in this work are limited to standard English WSD benchmarks, and we have not tested the LLMs' WSD capabilities of other languages. However, our framework is designed to support custom datasets, and we make all code publicly available in our repository to facilitate testing with diverse language datasets.

Second, our analysis of prompt bias employs only five constrained prompt templates. While this sufficiently demonstrates the existence of prompt biases and our framework effectively mitigates their impact, we acknowledge that individual models might achieve optimal performance with different, model-specific prompt formulations that we have not explored.

Third, we observe significant label imbalance in polysemous word distributions within existing datasets, reflecting real-world semantic frequency distributions. This inherent data characteristic introduces potential fairness issues in MCQA evaluation difficulty across different words. However, naively introducing distractor options would similarly create comparable evaluation biases. We identify this as an important direction for future methodological improvements.

In conclusion, the generative and open-ended nature of LLMs means their WSD performance cannot be fully captured by traditional correct/incorrect binary judgments. The comprehensive evaluation of LLMs' WSD capabilities remains an open research question that requires continued investigation.

Acknowledgments

This work is supported by the National Natural Science Foundation of China (No.62376130), Shandong Provincial Natural Science Foundation (No.ZR2022MF243), Program of New Twenty Policies for Universities of Jinan (No.202333008), and the Pilot Project for Integrated Innovation of Science, Education, and Industry of Qilu University of Technology (Shandong Academy of Sciences) (No.2025ZDZX01).

References

Simone Balloccu, Patrícia Schmidtová, Mateusz Lango, and Ondrej Dusek. 2024. Leak, cheat, repeat: Data contamination and evaluation malpractices in closed-source LLMs. In *Proceedings of the 18th Conference of the European Chapter of the Association for Computational Linguistics*, pages 67–93.

Michele Bevilacqua and Roberto Navigli. 2020. Breaking through the 80% glass ceiling: Raising the state of the art in word sense disambiguation by incorporating knowledge graph information. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 2854–2864.

Michele Bevilacqua, Tommaso Pasini, Alessandro Raganato, and Roberto Navigli. 2021. Recent trends in word sense disambiguation: A survey. In *Proceedings of the 30th International Joint Conference on Artificial Intelligence*, pages 4330–4338.

Zheng Chu, Jingchang Chen, Qianglong Chen, Weijiang Yu, Haotian Wang, Ming Liu, and Bing Qin. 2024. TimeBench: A comprehensive evaluation of temporal reasoning abilities in large language models. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics*, pages 1204–1228.

Abhimanyu Dubey, Abhinav Jauhri, Abhinav Pandey, Abhishek Kadian, Ahmad Al-Dahle, Aiesha Letman, Akhil Mathur, Alan Schelten, Amy Yang, Angela Fan, et al. 2024. The Llama 3 herd of models. *arXiv preprint arXiv:2407.21783*.

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

Vage Egiazarian, Andrei Panferov, Denis Kuznedelev, Elias Frantar, Artem Babenko, and Dan Alistarh. 2024. Extreme compression of large language models via additive quantization. *arXiv preprint arXiv:2401.06118*.

Aparna Elangovan, Ling Liu, Lei Xu, Sravan Babu Bodapati, and Dan Roth. 2024. ConSiDERS-the-human evaluation framework: Rethinking human evaluation for generative large language models. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1137–1160.

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 (Volume 1: Long Papers)*, pages 3130–3148.

Shahriar Golchin and Mihai Surdeanu. 2024. Time travel in LLMs: Tracing data contamination in large language models. In *Proceedings of the 12nd International Conference on Learning Representations*.

Luyao Huang, Chi Sun, Xipeng Qiu, and Xuanjing Huang. 2019. GlossBERT: BERT for word sense disambiguation with gloss knowledge. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing*, pages 3509–3514.

Aaron Hurst, Adam Lerer, Adam P Goucher, Adam Perelman, Aditya Ramesh, Aidan Clark, AJ Ostrow, Akila Welihinda, Alan Hayes, Alec Radford, et al. 2024. GPT-4o system card. *arXiv preprint arXiv:2410.21276*.

James Joyce. 2003. Bayes' theorem.

Katikapalli Subramanyam Kalyan. 2024. A survey of GPT-3 family large language models including ChatGPT and GPT-4. *Natural Language Processing Journal*, 6:100048.

Raihan Kibria, Sheikh Dipta, and Muhammad Adnan. 2024. On functional competence of llms for linguistic disambiguation. In *Proceedings of the 28th Conference on Computational Natural Language Learning*, pages 143–160.

Anastasia Kritharoula, Maria Lymperaiou, and Giorgos Stamou. 2023. Large language models and multimodal retrieval for visual word sense disambiguation. *arXiv preprint arXiv:2310.14025*.

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.

Sunjae Kwon, Dongsuk Oh, and Youngjoong Ko. 2021. Word sense disambiguation based on context selection using knowledge-based word similarity. *Information Processing & Management*, 58(4):102551.

Aixin Liu, Bei Feng, Bing Xue, Bingxuan Wang, Bochao Wu, Chengda Lu, Chenggang Zhao, Chengqi Deng, Chenyu Zhang, Chong Ruan, et al. 2024. Deepseek-v3 technical report. *arXiv preprint arXiv:2412.19437*.

Jiawei Liu, Chunqiu Steven Xia, Yuyao Wang, and LINGMING ZHANG. 2023. Is your code generated by chatgpt really correct? rigorous evaluation of large language models for code generation. In *Proceedings of the Advances in Neural Information Processing Systems*, volume 36, pages 21558–21572.

Wenpeng Lu, Fanqing Meng, Shoujin Wang, Guoqiang Zhang, Xu Zhang, Antai Ouyang, and Xiaodong Zhang. 2019. Graph-based chinese word sense disambiguation with multi-knowledge integration. *Computers, Materials and Continua*, 61(1):197–212.

Mouheb Mehdoui, Amel Fraisse, and Mounir Zrigui. 2024. Leveraging large language models to build a cutting-edge French word sense disambiguation corpus.

Sakae Mizuki and Naoaki Okazaki. 2023. Semantic specialization for knowledge-based word sense disambiguation. In *Proceedings of the 17th Conference of the European Chapter of the Association for Computational Linguistics*, pages 3457–3470.

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, 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.

OpenAI. 2023. GPT-4 technical report. *arXiv preprint arXiv:2303.08774*.

Aristi Papastavrou, Maria Lymperaiou, and Giorgos Stamou. 2024. Arpa: A novel hybrid model for advancing visual word disambiguation using large language models and transformers. *arXiv preprint arXiv:2408.06040*.

Kishore Papineni, Salim Roukos, Todd Ward, and Wei-Jing Zhu. 2002. Bleu: a method for automatic evaluation of machine translation. In *Proceedings of the 40th annual meeting of the Association for Computational Linguistics*, pages 311–318.

Tommaso Pasini. 2021. The knowledge acquisition bottleneck problem in multilingual word sense disambiguation. In *Proceedings of the 29th International Conference on International Joint Conferences on Artificial Intelligence*, pages 4936–4942.

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.

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.

Alessandro Raganato, Claudio Delli Bovi, and Roberto Navigli. 2017. Neural sequence learning models for word sense disambiguation. In *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing*, pages 1156–1167.

Nils Reimers and Iryna Gurevych. 2019. Sentence-BERT: Sentence embeddings using Siamese BERT-networks. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing*, pages 3982–3992.

Oscar Sainz, Jon Campos, Iker García-Ferrero, Julen Etxaniz, Oier Lopez de Lacalle, and Eneko Agirre. 2023. NLP evaluation in trouble: On the need to measure LLM data contamination for each benchmark. In *Findings of the Association for Computational Linguistics: EMNLP 2023*, pages 10776–10787.

Natalie Schluter. 2017. The limits of automatic summarisation according to rouge. In *Proceedings of the 15th Conference of the European Chapter of the Association for Computational Linguistics*, pages 41–45.

Melanie Sclar, Yejin Choi, Yulia Tsvetkov, and Alane Suhr. 2023. Quantifying language models' sensitivity to spurious features in prompt design or: How I learned to start worrying about prompt formatting. *arXiv preprint arXiv:2310.11324*.

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.

Hwanjun Song, Hang Su, Igor Shalyminov, Jason Cai, and Saab Mansour. 2024. FineSurE: Fine-grained summarization evaluation using LLMs. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics*, pages 906–922.

TGDK Sumanathilaka, Nicholas Micallef, and Julian Hough. 2024. Can LLMs assist with ambiguity? A quantitative evaluation of various large language models on word sense disambiguation. *arXiv preprint arXiv:2411.18337*.

Hugo Touvron, Louis Martin, Kevin Stone, Peter Albert, Amjad Almahairi, Yasmine Babaei, Nikolay Bashlykov, Soumya Batra, Prajjwal Bhargava, Shruti Bhosale, et al. 2023. Llama 2: Open foundation and fine-tuned chat models. *arXiv preprint arXiv:2307.09288*.

Loïc Vial, Benjamin Lecouteux, and Didier Schwab. 2019. Sense vocabulary compression through the semantic knowledge of WordNet for neural word sense disambiguation. In *Proceedings of the 10th Global WordNet Conference*, pages 108–117.

Yue Wang, Hua Zheng, Yaqi Yin, Hansi Wang, Qiliang Liang, and Yang Liu. 2024. Morpheme sense disambiguation: A new task aiming for understanding the language at character level. In *Proceedings of the 2024 Joint International Conference on Computational Linguistics, Language Resources and Evaluation*, pages 11605–11618.

Jinyang Wu, Feihu Che, Xinxin Zheng, Shuai Zhang, Ruihan Jin, Shuai Nie, Pengpeng Shao, and Jianhua Tao. 2024. Can large language models understand uncommon meanings of common words? *arXiv preprint arXiv:2405.05741*.

Cheng Xu, Shuhao Guan, Derek Greene, M Kechadi, et al. 2024. Benchmark data contamination of large language models: A survey. *arXiv preprint arXiv:2406.04244*.

Jung H Yae, Nolan C Skelly, Neil C Ranly, and Phillip M LaCasse. 2024. Leveraging large language models for word sense disambiguation. *Neural Computing and Applications*, 37:4093—-4110.

Guobiao Zhang, Wenpeng Lu, Xueping Peng, Shoujin Wang, Baoshuo Kan, and Rui Yu. 2022. Word sense disambiguation with knowledge-enhanced and local self-attention-based extractive sense comprehension. In *Proceedings of the 29th International Conference on Computational Linguistics*, pages 4061–4070.

Tianyi Zhang*, Varsha Kishore*, Felix Wu*, Kilian Q. Weinberger, and Yoav Artzi. 2020. Bertscore: Evaluating text generation with bert. In *International Conference on Learning Representations*.

Chujie Zheng, Hao Zhou, Fandong Meng, Jie Zhou, and Minlie Huang. 2024. Large language models are not robust multiple choice selectors. In *Proceedings of the 12nd International Conference on Learning Representations*.

A Appendix

A.1 Data Contamination

To avoid the risk of data contamination from the public datasets used to evaluate LLMs, we use a data contamination detection method that leverages model's memory (Golchin and Surdeanu, 2024). This method prompt model to complete fragments of the original data in the datasets by constructing two different prompts. The "Instruction General" prompt does not contain any information about the original dataset, while the "Instruction Guided" prompt includes the data source's dataset and the original labels to guide the model in completing the data. By comparing the model outputs under these two prompts, we can determine whether there is a risk of data contamination.

Table 9 shows the results on five evaluated models. According to this method, a difference in ROUGE-L greater than 0.1 indicates a risk of data contamination. Therefore, the datasets we used do not pose a risk of data contamination. Additionally, to ensure the re-usability of the RoDEval, we did not adopt the method used in the original paper of using GPT to make secondary judgments on the model's responses. However, we also conduct human evaluations of these output, and only one instance from SemEval2015 is exact match DeepSeek-v3's response. Given the negligible difference in ROUGE-L between the Instruction General and Instruction Guided, we speculate that the models might have encountered the context of this data outside this dataset, rather than experiencing data contamination. Therefore, we conclude that the five public datasets we used do not pose a risk of data contamination for these five models. All of these models' responses are public in the code repository.

A.2 Robust Punctuation-free formatting Rules

We manually inspect 100 randomly sampled data instances and find that naively replacing all special characters with spaces introduces seven distinct error patterns, which either create new ambiguities or render contexts unintelligible. These patterns should be retained during replacement, so we add the following matching protection rules (regex). The error patterns and corresponding regular expression rules are summarized in the table 10. The concrete examples of different error patterns alongside their corresponding results after applying robust rules are as follow:

- **Original Example:** Brazil is the country which exercises most leadership in the region, mentioned by **19%** of respondents **(up from 18% last year)**, followed by the United States **(9%, unchanged from last year)** and Venezuela **(9%, down from 11% last year)**.

  Punctuation-free: Brazil is the country which exercises most leadership in the region mentioned by **19%** of respondents **(up from 18% last year)** followed by the United States **(9%, unchanged from last year)** and Venezuela **(9%, down from 11% last year)**

- **Original Example:** Alimta was also compared with gemcitabine **(another anticancer medicine)**, both in combination with cisplatin , in a study involving **1,725** patients who had not received chemotherapy for lung cancer in the past.

  Punctuation-free: Alimta was also compared with gemcitabine **(another anticancer medicine)** both in combination with cisplatin in a study involving **1,725** patients who had not received chemotherapy for lung cancer in the past

- **Original Example: U.S.** special climate envoy Todd Stern rejected language requiring binding cuts of **greenhouse-gas** emissions for industrialized countries compared with voluntary ones by major emerging economies if they were not funded by the developed world.

  Punctuation-free: U.S special climate envoy Todd Stern rejected language requiring binding cuts of **greenhouse-gas** emissions for industrialized countries compared with voluntary ones by major emerging economies if they were not funded by the developed world

A.3 Prompt Bias

Table 11 and Tables 12-14 show the performance of all models under different prompt templates, where σ^2 denotes the variance. The experiments include five types: **Normal** refers to using standard English grammar in the template; **Lower** means all words and letters in the template are in lowercase; **Upper** means all words and letters are in uppercase; **Title** capitalizes the first letter of each word; and **Nopunc** replaces all punctuation marks with spaces.

Model	Instruct.	Datasets				
		Eval07	Eval13	Eval15	Eval2	Eval3
LLaMA2-7b	General	0.0578	0.0866	0.0559	0.0791	0.0638
	Guided	0.0813	0.0688	0.0613	0.0924	0.0558
Human Evaluation		✗	✗	✗	✗	✗
LLaMA3.1-8b	General	0.0720	0.1169	0.0936	0.0897	0.0700
	Guided	0.1165	0.1095	0.0982	0.0763	0.0800
Human Evaluation		✗	✗	✗	✗	✗
LLaMA3.1-70b	General	0.1212	0.0892	0.0962	0.0642	0.0608
	Guided	0.1123	0.1567	0.0989	0.1176	0.1133
Human Evaluation		✗	✗	✗	✗	✗
GPT-4o	General	0.0914	0.1152	0.1167	0.0799	0.0916
	Guided	0.0820	0.0628	0.0409	0.0613	0.0312
Human Evaluation*		✗	✗	✗	✗	✗
DeepSeek-v3	General	0.0949	0.1086	0.1583	0.0951	0.1001
	Guided	0.1770	0.1285	0.0940	0.0526	0.0611
Human Evaluation		✗	✗	✓	✗	✗

Table 9: Data Contamination. The data in the table shows the ROUGE-L difference between General Instruction and Guided Instruction. A single tick (✓) points to the presence of at least one exact match, while a cross sign (✗) denotes that no exact match. Human Evaluation*: GPT-4o refuse to complete the most of instances under the Instruction Guided, so we alternate to evaluate Instruction General instead.

Error Pattern	Regex
Thousand separators (e.g., 1,000)	\d{1,3}(?:,\d{3})+
Parenthetical supplements (e.g., (29 points))	\([^)+ \)
Hyphenated compounds (e.g., three-pointers)	\b\w+-\w+\b
Special markers (e.g., -LRB-, -RRB-)	-\w+-
Decimals (e.g., 0.63)	\d+\.\d+
Abbreviations (e.g., e.g.)	\b[A-Za-z]+[̇A-Za-z]+\.?\b
Percentages (e.g., 50%)	\d+%

Table 10: Error Patterns for Naive Punctuation-to-Space Substitution and Corresponding Regular Expression Rules.

Dataset	**Normal**	**Lower**	**Upper**	**Title**	**No-punc**	σ^2
Eval07	41.88	43.76	43.53	42.05	41.56	0.82
Eval13	56.47	56.27	56.77	56.46	56.58	0.03
Eval15	59.42	56.93	58.86	57.29	57.32	0.97
Eval2	52.44	53.51	54.31	54.39	53.33	0.51
Eval3	44.84	47.32	47.76	48.41	47.63	1.51

Table 11: F1 scores(%) of LLaMA2-7b on DOCQ using five different prompt templates. σ^2 is the population variance.

Dataset	Normal	Lower	Upper	Title	No-punc	σ^2
Eval07	51.11	51.83	53.65	49.27	52.74	2.23
Eval13	67.88	64.35	68.75	68.59	68.33	2.69
Eval15	66.29	65.31	67.25	67.24	66.24	0.54
Eval2	63.04	61.16	61.99	62.66	62.09	0.41
Eval3	58.19	56.53	57.57	57.63	57.43	0.29

Table 12: F1 scores (%) of LLaMA3.1-8b on DOCQ using four prompt templates.

Dataset	Normal	Lower	Upper	Title	No-punc	σ^2
Eval07	61.58	64.02	62.55	63.65	63.40	0.77
Eval13	77.65	77.99	78.10	77.25	76.10	0.52
Eval15	77.68	73.41	76.52	77.90	77.33	2.71
Eval2	71.75	72.71	67.96	71.68	71.64	2.70
Eval3	68.16	68.53	68.37	68.86	69.04	0.10

Table 13: F1 scores (%) of LLaMA3.1-70b on DOCQ using four prompt templates.

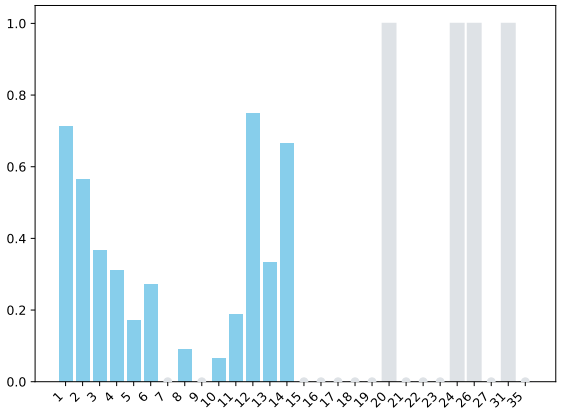
The results indicate that as the number of model parameters increases from 7B to 70B and beyond, the prompt bias of the models significantly decreases. In particular, GPT-4o and DeepSeek-V3 are almost unaffected by prompt templates. This may be because as the number of parameters increases, the model's representation ability becomes stronger, and the range of understandable language expands, thereby reducing the dependence on specific prompt formats.

Moreover, it can be seen from the tables that all models generally perform best under the prompt template that most conforms to human grammar (normal), because the training corpus typically follows the human grammar. However, the experimental results also show that all models are affected by prompt templates to varying degrees, especially the absence of punctuation. This poses a requirement for users, as they usually do not follow punctuation rules when using the models, which may result in performance below expectations. This demonstrates the necessity for our framework to first conduct experiments on prompt formatting.

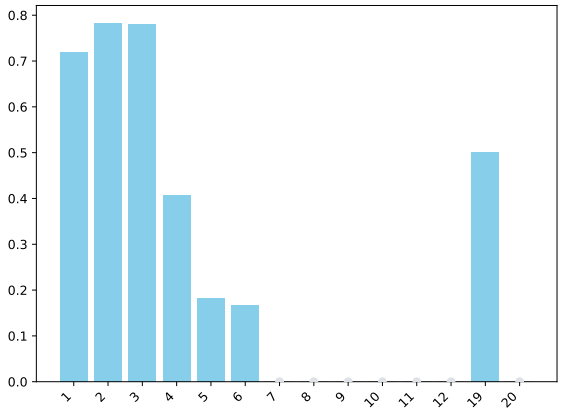
A.4 Selection Bias

In this section, to explore the manifestations and reasons of selection bias, we present the recall and number of selected option in the DOCQ task. The results show that models generally have higher recall when the correct option is 1 or 2. Additionally, LLaMA2-7b (Figure 5) exhibits high recall for options 11 to 14 across most datasets. LLaMA3.1-8b (Figure 6) has high recall for the first ten options, but options 5 and 8 typically have lower recall compared to the other eight options. When the parameter scale reaches 70b (LLaMA3.1-70b, GPT-4o, and DeepSeek-v3), the models' option recall show a clear negative correlation with the option ids. These phenomena indicate that the models' ability to handle each option is not entirely linearly related to the option ids.

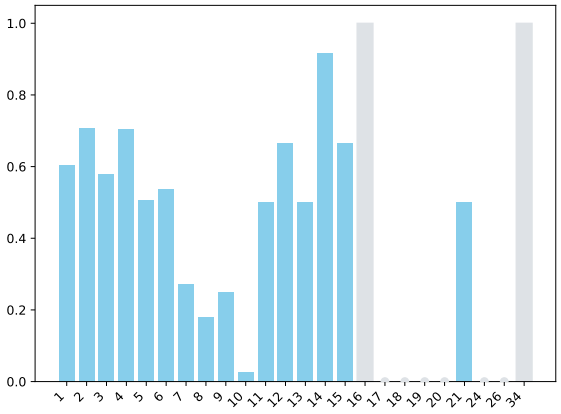
Figures 10-14 show that the models exhibit a clear bias for options 1 and 2, with the number of selections for these options far exceeding their recall. This indicates that when the models are uncertain about the correct answer, they tend to choose these two options. However, comparing Figure 8 and Figure 14, Figure 8 and Figure 14 reveals that even though these two models select option 1 significantly more often than other options, their recalls remain close to 1.0. This suggests that even with randomization of the candidate sense sequences, the correct answer still appears in option 1 much more frequently than in others, indicating that a large number of polysemous words in the dataset have very few candidate senses. This suggests that there may be a severe label imbalance in the training data of the models, which in turn may cause the models to exhibit a selection bias during WSD.



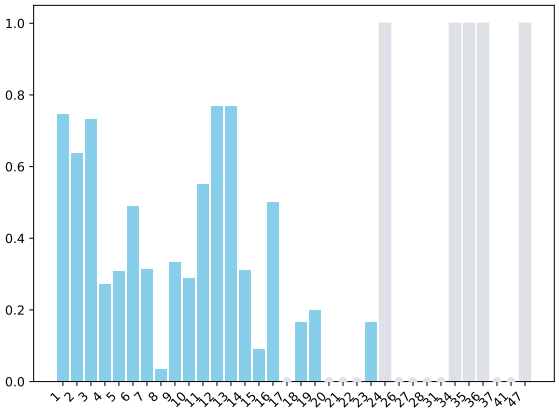
(a) SemEval2007



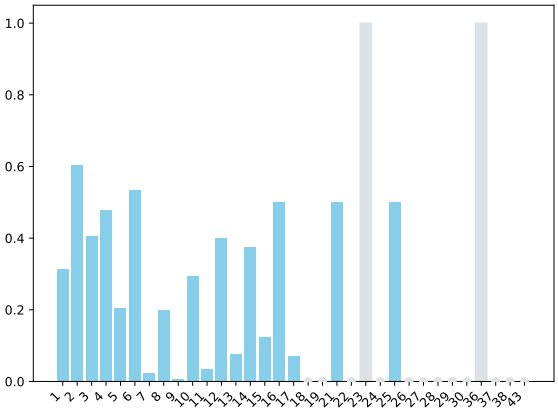
(b) SemEval2013



(c) SemEval2015

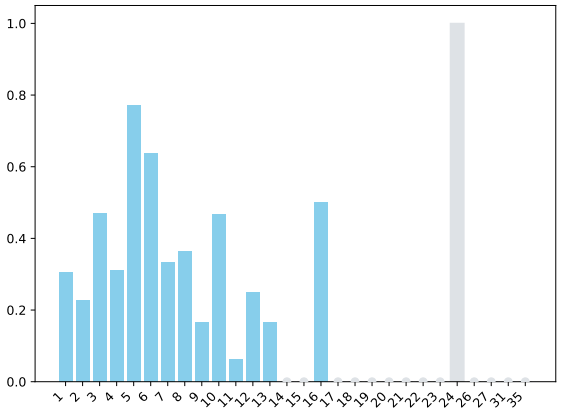


(d) Senseval2

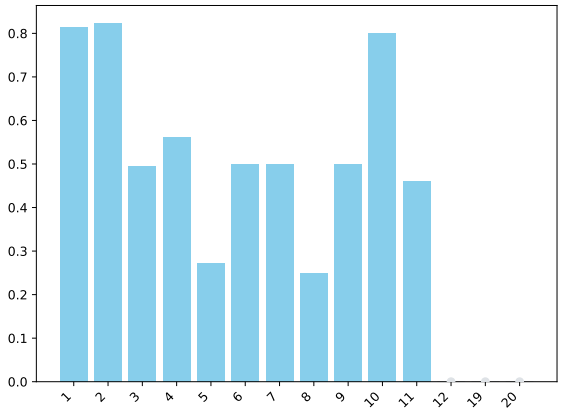


(e) Senseval3

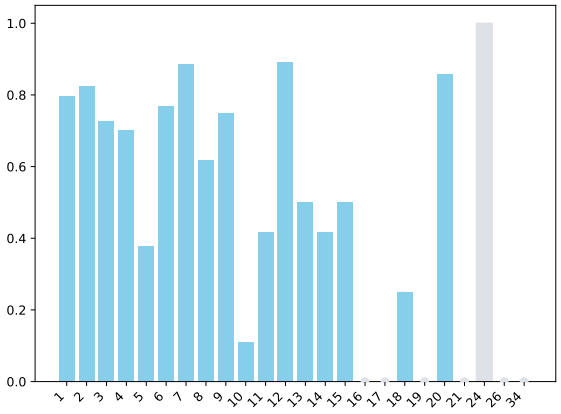
Figure 5: Recall of Each Option of LLaMA2-7b. Grey denotes options that appear only once.



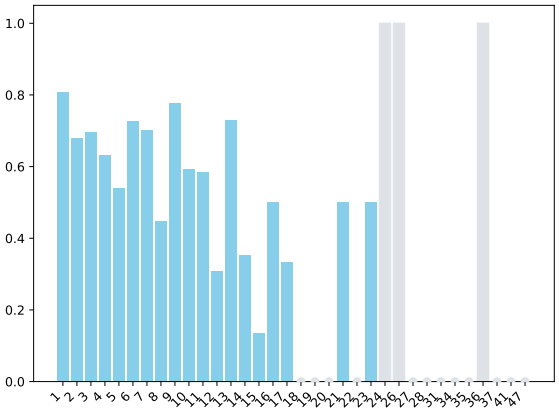
(a) SemEval2007



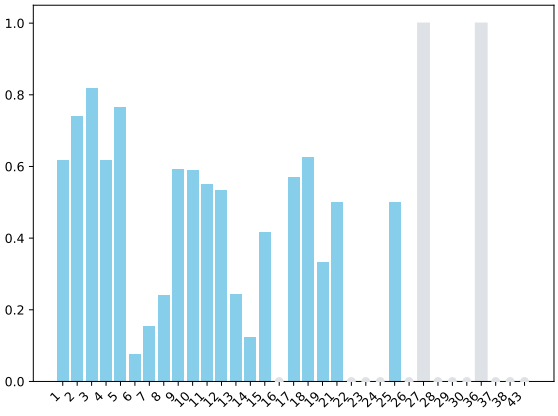
(b) SemEval2013



(c) SemEval2015

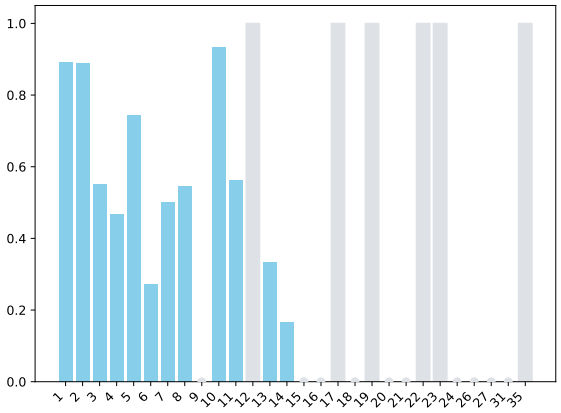


(d) Senseval2

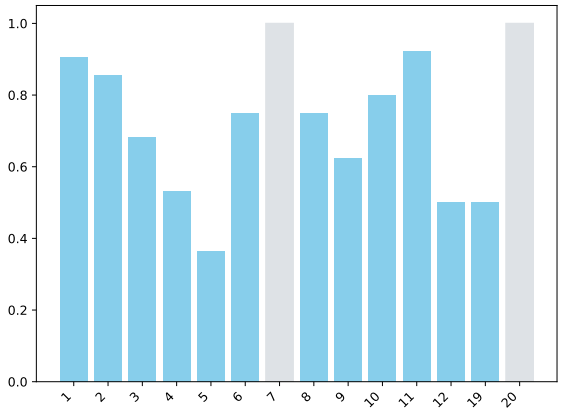


(e) Senseval3

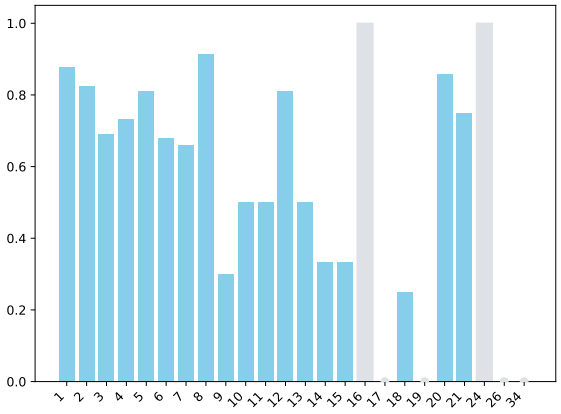
Figure 6: Recall of Each Option of LLaMA3.1-8b. Grey denotes options that appear only once.



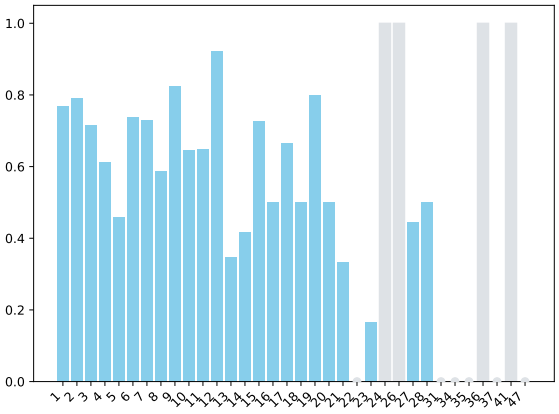
(a) SemEval2007



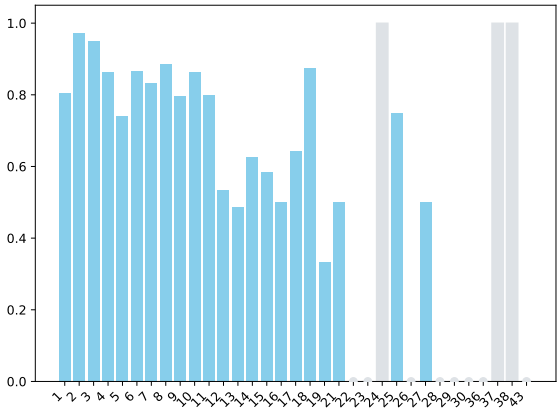
(b) SemEval2013



(c) SemEval2015

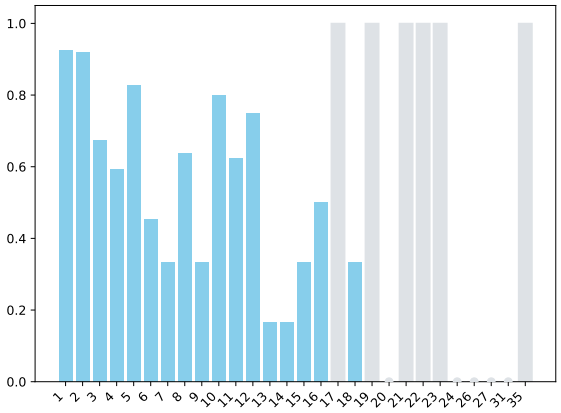


(d) Senseval2

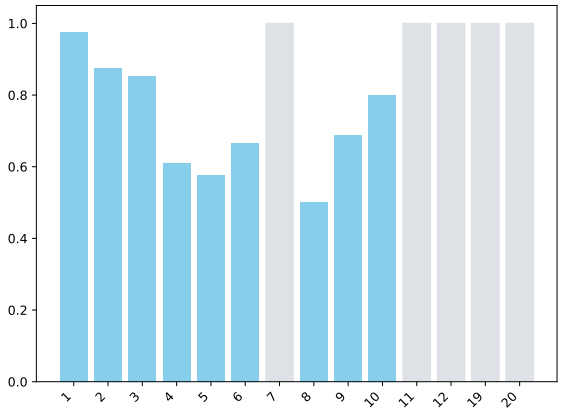


(e) Senseval3

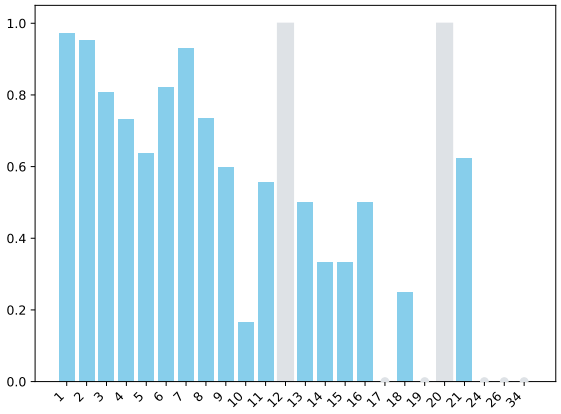
Figure 7: Recall of Each Option of LLaMA3.1-70b. Grey denotes options that appear only once.



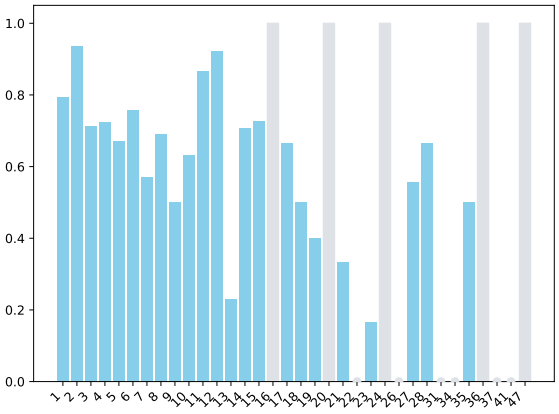
(a) SemEval2007



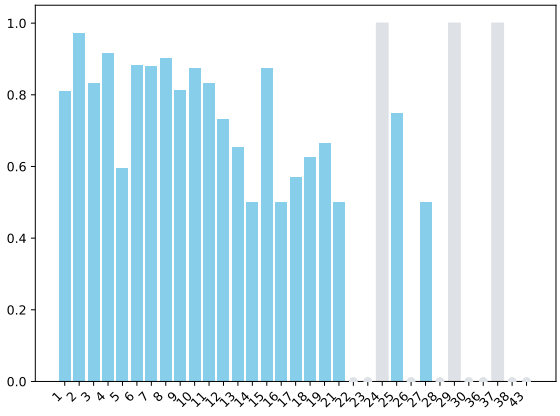
(b) SemEval2013



(c) SemEval2015

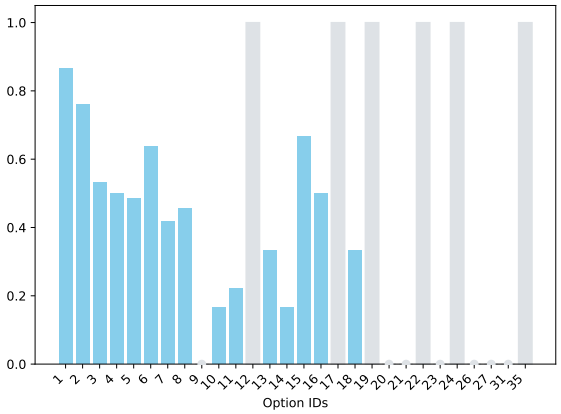


(d) Senseval2

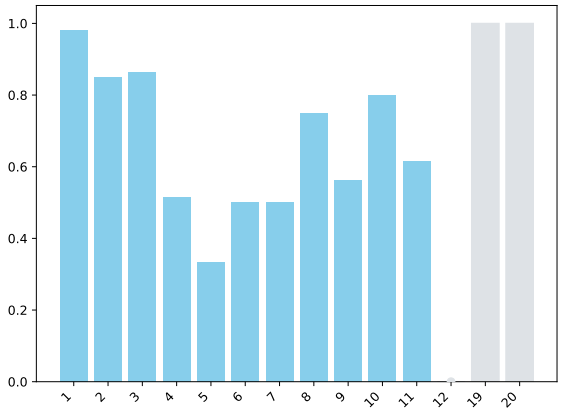


(e) Senseval3

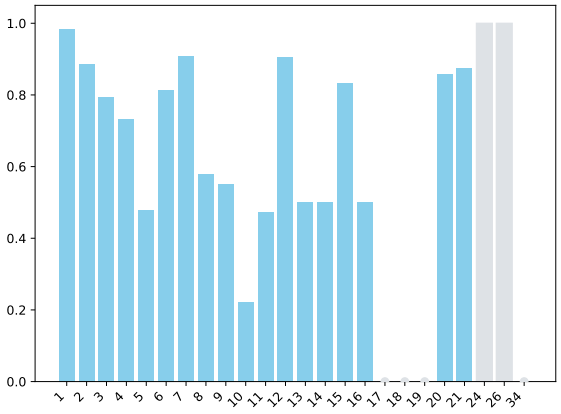
Figure 8: Recall of Each Option of GPT-4o. Grey denotes options that appear only once.



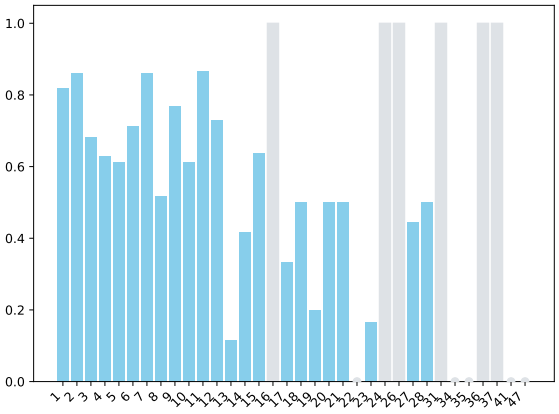
(a) SemEval2007



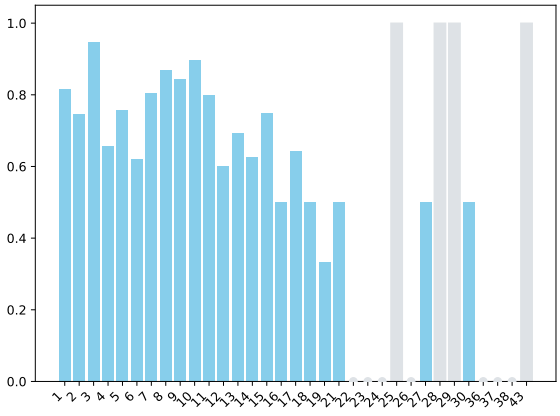
(b) SemEval2013



(c) SemEval2015

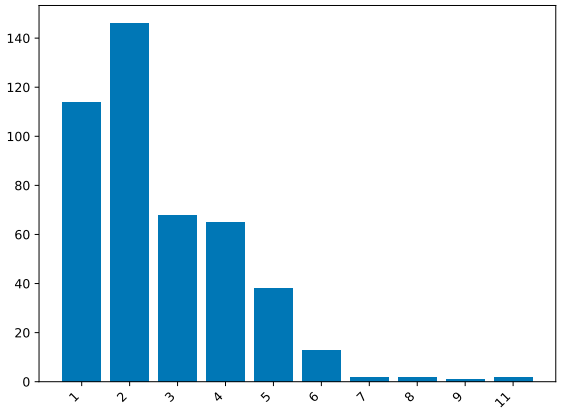


(d) Senseval2

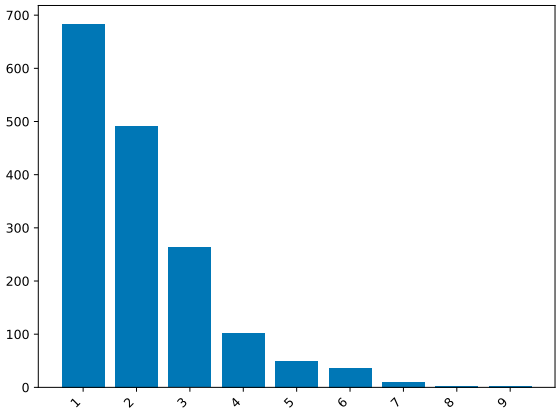


(e) Senseval3

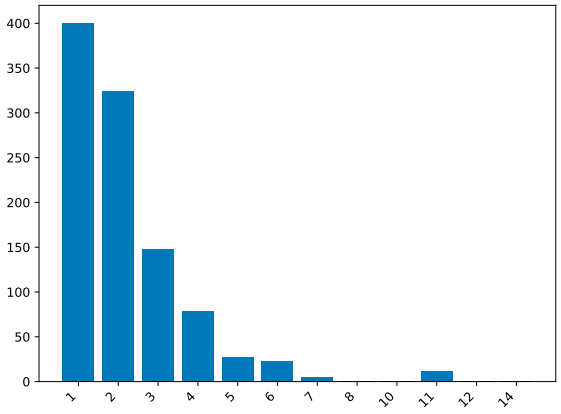
Figure 9: Recall of Each Option of DeepSeek-v3. Grey denotes options that appear only once.



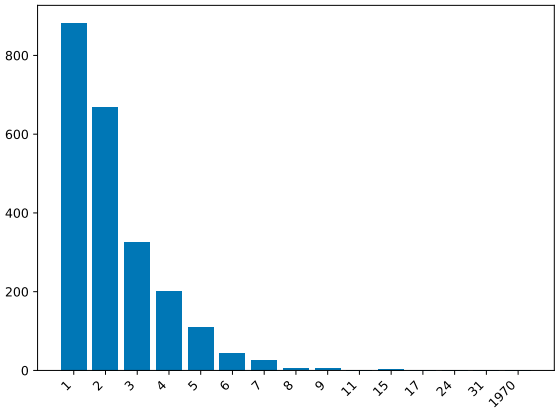
(a) SemEval2007



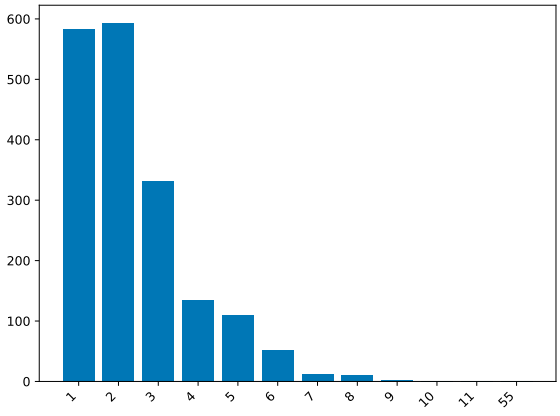
(b) SemEval2013



(c) SemEval2015

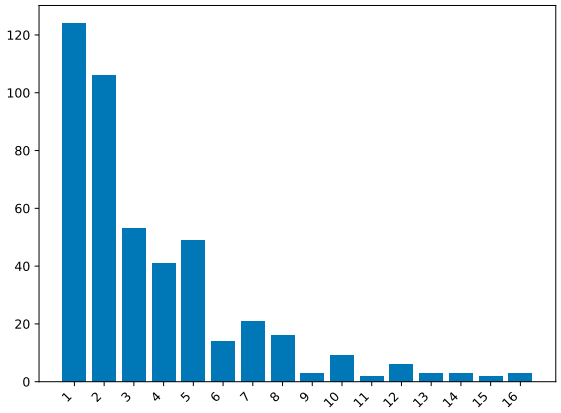


(d) Senseval2

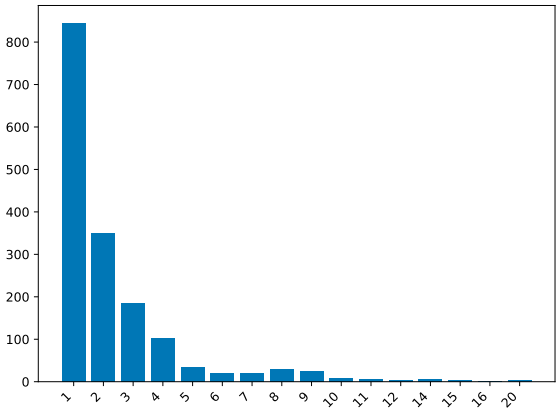


(e) Senseval3

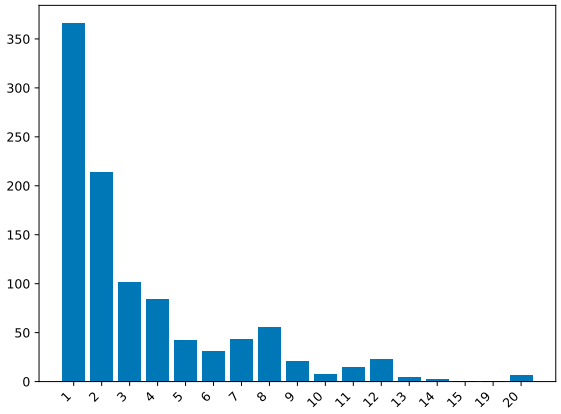
Figure 10: Number of Selected Option of LLaMA2-7b



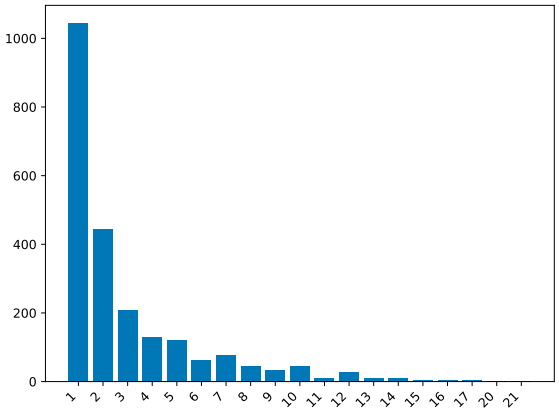
(a) SemEval2007



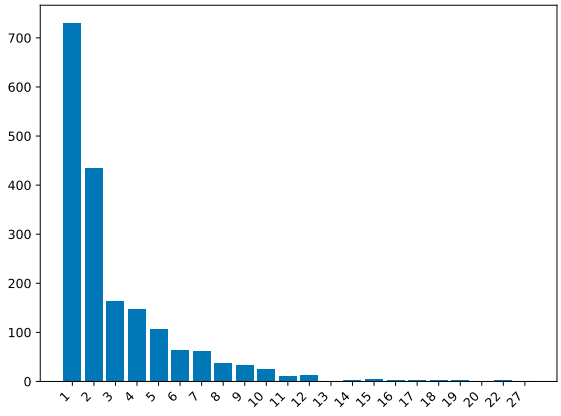
(b) SemEval2013



(c) SemEval2015

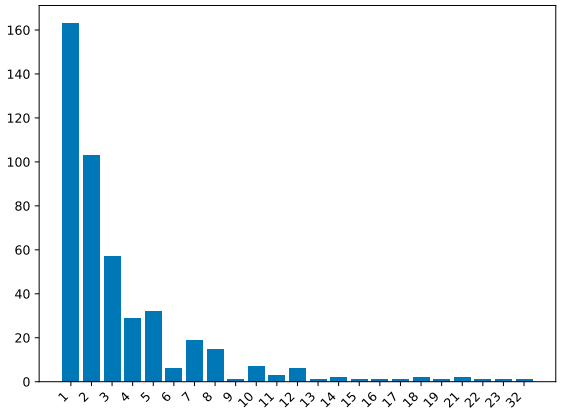


(d) Senseval2

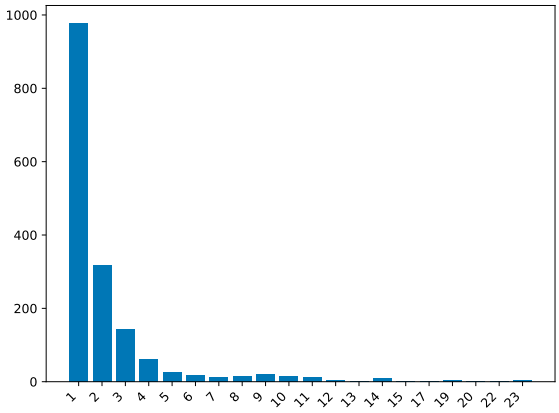


(e) Senseval3

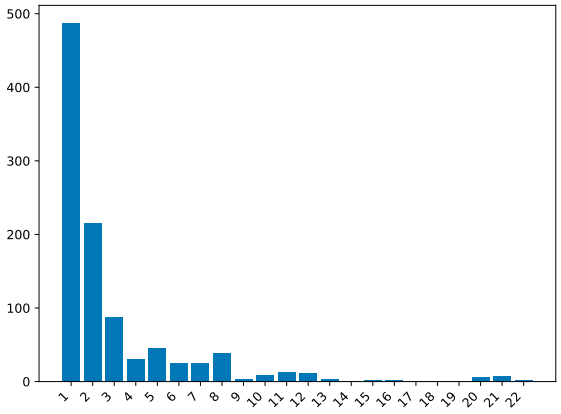
Figure 11: Number of Selected Option of LLaMA3.1-8b



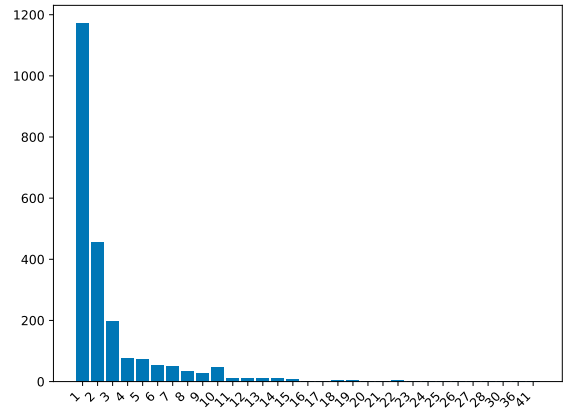
(a) SemEval2007



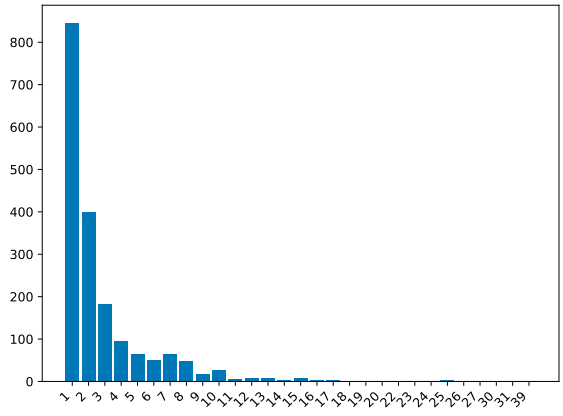
(b) SemEval2013



(c) SemEval2015

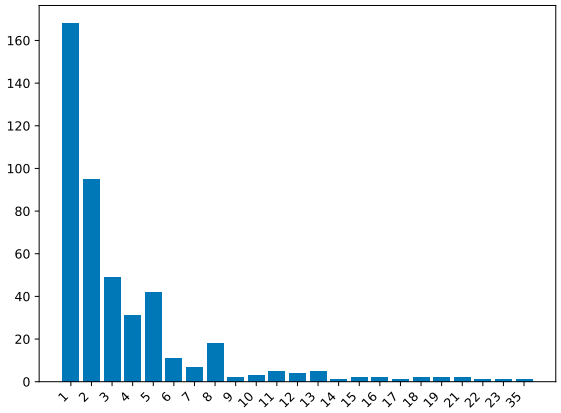


(d) Senseval2

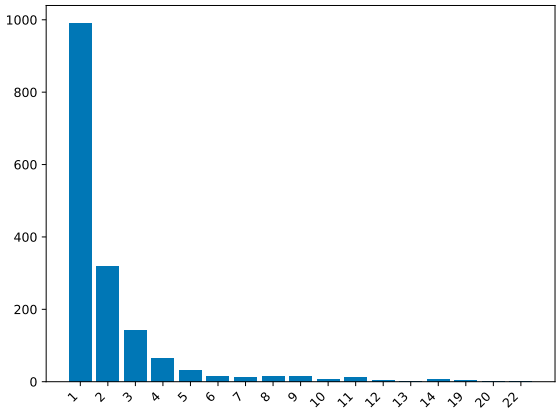


(e) Senseval3

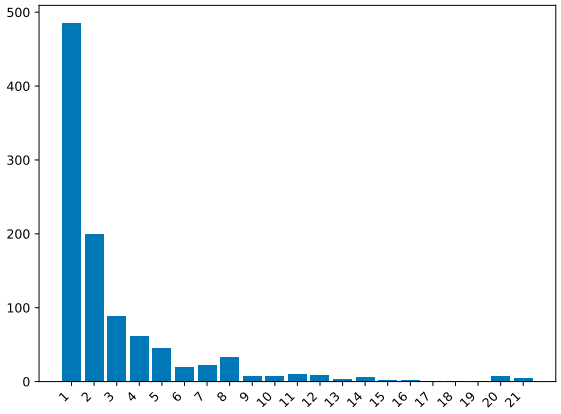
Figure 12: Number of Selected Option of LLaMA3.1-70b



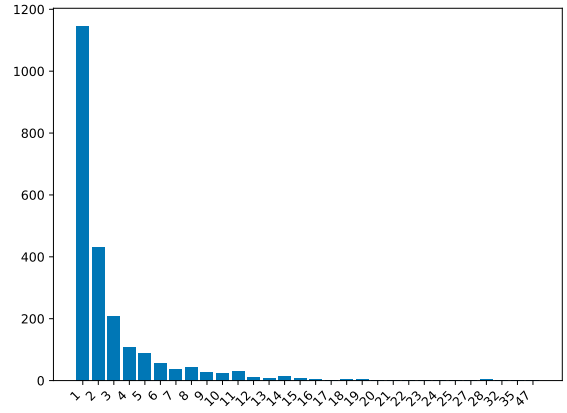
(a) SemEval2007



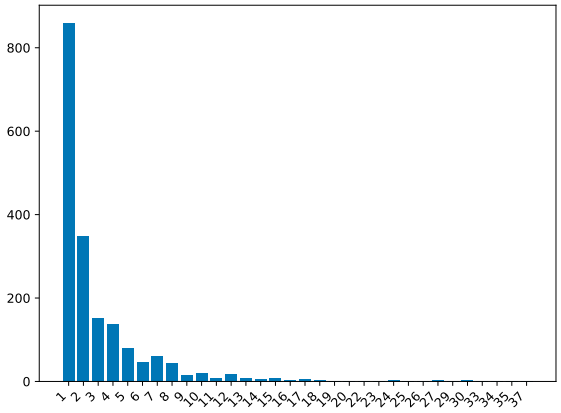
(b) SemEval2013



(c) SemEval2015

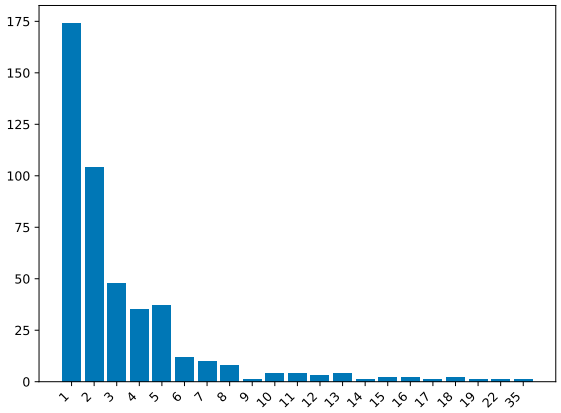


(d) Senseval2

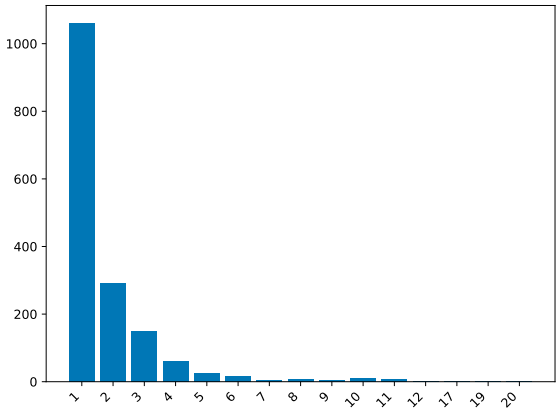


(e) Senseval3

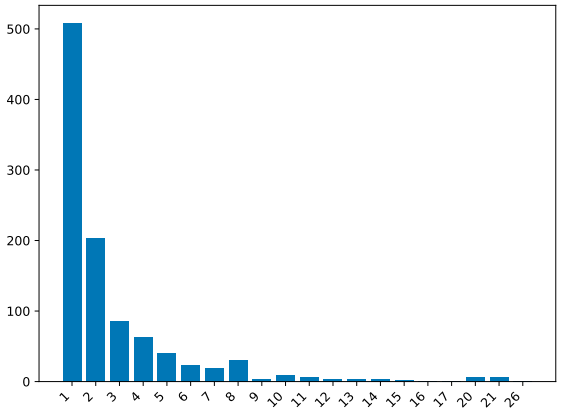
Figure 13: Number of Selected Option of GPT-4o



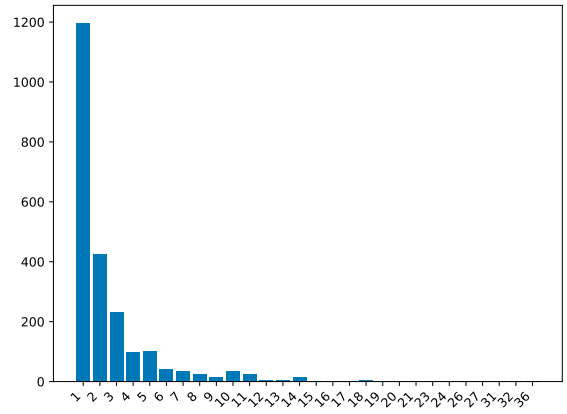
(a) SemEval2007



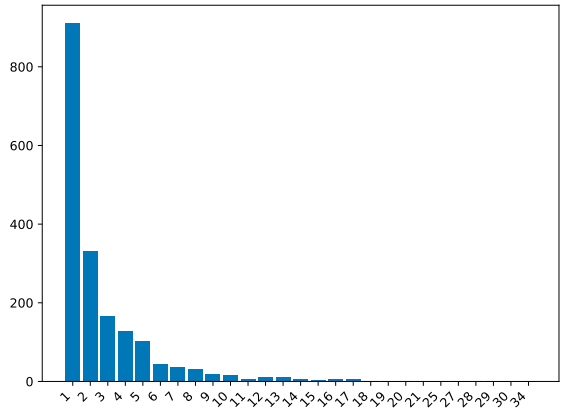
(b) SemEval2013



(c) SemEval2015



(d) Senseval2



(e) Senseval3

Figure 14: Number of Selected Option of DeepSeek-v3

Dataset	Normal	Lower	Upper	Title	No-punc	σ^2
Eval07	66.09	66.60	66.61	67.07	64.88	0.57
Eval13	80.57	80.12	80.58	80.56	80.70	0.04
Eval15	81.09	82.76	80.92	82.15	82.24	0.50
Eval2	74.09	76.52	75.65	76.27	75.24	0.74
Eval3	72.07	75.05	74.21	75.15	75.45	1.51

Table 14: F1 scores (%) of DeepSeek-v3 on DOCQ using four prompt templates.

A.5 Number of Different Correct Options for MCQs

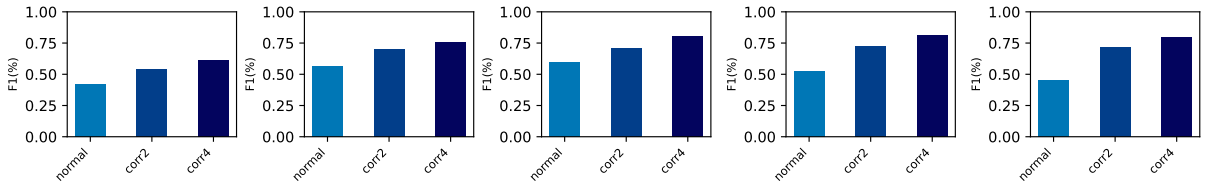
In this section, we investigate how the model's performance changes when increasing the number of correct options in MCQs. Considering that an excessive number of correct options might exceed the number of candidate meanings for some polysemous words, we evaluate F1 scores of MCQs by increasing the number of correct options to 2 and 4, respectively. The Figures 15-19 show the performance of all models under different numbers of correct options. The results indicate that the F1 scores of all models are positively correlated with the number of correct options. Furthermore, we calculated the increase in F1 scores and their variances for all models when the number of correct options increased, as shown in the table 15. It indicates that, despite the varying degrees of word sense disambiguation capabilities among the models, the increase in F1 scores is similar when the number of correct options is increased (with all variances being less than 0.002). This suggests that,increasing the number of correct options cannot help the model select the correct answer; the model's errors are not due to ignoring or forgetting the correct answer.

A.6 Generative Metrics

In this section, Figures 16-19 show the performance of five models on the DG tasks of SemEval2013, SemEval2015, Senseval2, and Senseval3, evaluated using various generative metrics. The results show that while relaxed matching F1 can differentiate the quality of generated content across models, it significantly underestimates the generation quality for all models. Similarly, ROUGE scores exhibit the same issue. BLEU not only show minimal differences among models with varying generation capabilities, making it difficult to distinguish their quality, but also exhibit significant bias in evaluating generation quality. For instance, LLaMA2-7b achieved the second-highest BLEU scores across all datasets. **We will make all model outputs publicly available in our code repository.**

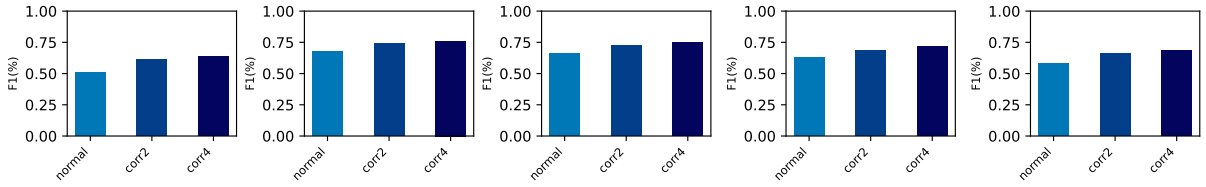
Dataset	DeepSeek	GPT-4o	LLaMA3.1-70b	LLaMA3.1-8b	LLaMA2-7b	σ^2
Eval07	0.0612	0.0084	0.0472	0.0613	0.0963	**0.0008**
Eval13	0.0226	0.0055	0.0212	0.0410	0.0972	**0.0010**
Eval15	0.0273	0.0218	0.0313	0.0441	0.1053	**0.0009**
Eval2	0.0397	0.0066	0.0324	0.0420	0.1430	**0.0022**
Eval3	0.0416	0.0110	0.0454	0.0522	0.1741	**0.0032**

Table 15: Increasing in F1 for all models when the number of correct options increased. The values in the table represent the average increase in F1 when the number of correct options increases from 1 to 2 and from 2 to 4, respectively.



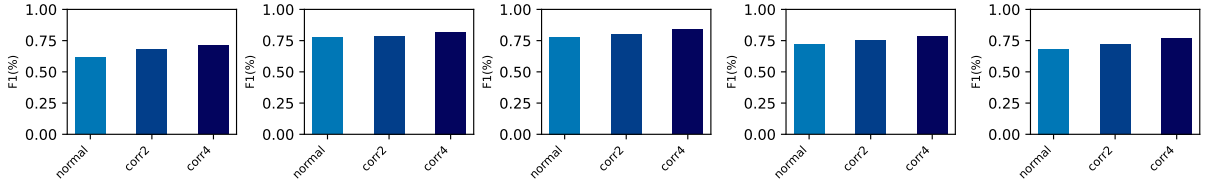
(a) SemEval2007 (b) SemEval2013 (c) SemEval2015 (d) Senseval2 (e) Senseval3

Figure 15: Number of Different Correct Options for MCQs Evaluated with LLaMA2-7b.



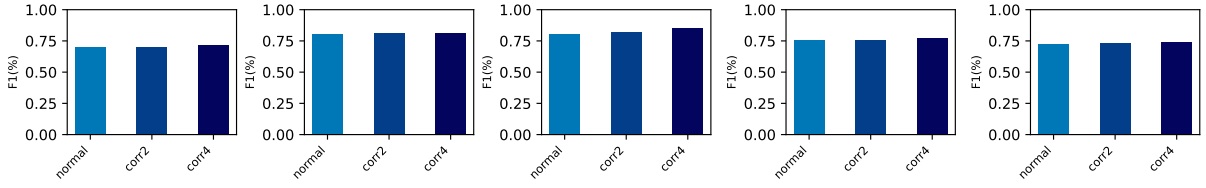
(a) SemEval2007 (b) SemEval2013 (c) SemEval2015 (d) Senseval2 (e) Senseval3

Figure 16: Number of Different Correct Options for MCQs Evaluated with LLaMA3.1-8b.



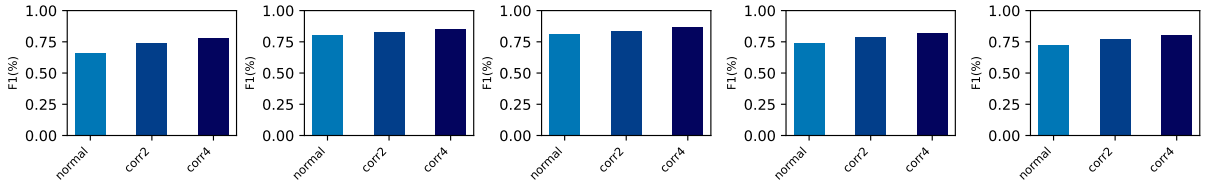
(a) SemEval2007 (b) SemEval2013 (c) SemEval2015 (d) Senseval2 (e) Senseval3

Figure 17: Number of Different Correct Options for MCQs Evaluated with LLaMA3.1-70b.



(a) SemEval2007 (b) SemEval2013 (c) SemEval2015 (d) Senseval2 (e) Senseval3

Figure 18: Number of Different Correct Options for MCQs Evaluated with GPT-4o.



(a) SemEval2007 (b) SemEval2013 (c) SemEval2015 (d) Senseval2 (e) Senseval3

Figure 19: Number of Different Correct Options for MCQs Evaluated with DeepSeek-v3.

	Typical Generative Metrics					
Model	F1	BLEU	R-1.	R-2.	R-l.	**DGQS**
LLaMA2-7b	0.2388	**0.0149**	0.1437	0.0281	0.1226	0.5187
LLaMA3.1-8b	0.2731	0.0128	0.1830	0.0463	0.1640	0.6175
LLaMA3.1-70b	0.3293	0.0141	0.2488	0.0717	0.2161	0.6799
GPT-4o	0.3033	0.0142	0.2460	0.0695	0.2125	0.6794
DeepSeek-v3	**0.3779**	0.0145	**0.2832**	**0.0886**	**0.2524**	**0.6832**
*LLaMA2-Dictionary	0.2685	0.0133	0.2106	0.0609	0.1875	0.6459
*LLaMA3-Dictionary	0.2874	0.0133	0.2038	0.0512	0.1785	0.6263
*T5-definition	0.2588	0.0097	0.1432	0.0441	0.1357	0.5836

Table 16: DGQS and typical Generative metrics of five models on SemEval2013. F1 represents the F1 score for relaxed Matching, while R-1, R-2, and R-L are ROUGE-1, ROUGE-2, and ROUGE-L, respectively. Bold numbers indicate the highest score, while underlined numbers indicate the second-highest. * indicates fine-tuned models.

	Typical Generative Metrics					
Model	F1	BLEU	R-1.	R-2.	R-l.	**DGQS**
LLaMA2-7b	0.2157	0.0133	0.1206	0.0215	0.1043	0.4703
LLaMA3.1-8b	0.1930	0.0101	0.1330	0.0338	0.1223	0.5247
LLaMA3.1-70b	0.3288	**0.0140**	0.2599	0.0722	0.2280	0.6595
GPT-4o	0.3219	0.0130	0.2649	**0.0920**	0.2390	**0.6766**
DeepSeek-v3	**0.3581**	0.0128	**0.2770**	0.0869	**0.2521**	0.6628
*LLaMA2-Dictionary	0.2572	0.0136	0.2050	0.0537	0.1861	0.6127
*LLaMA3-Dictionary	0.2966	0.0132	0.2012	0.0512	0.1819	0.5958
*T5-definition	0.3045	0.0090	0.1159	0.0296	0.1103	0.5206

Table 17: DGQS and typical Generative metrics of five models on SemEval2015. F1 represents the F1 score for relaxed matching, while R-1, R-2, and R-L are ROUGE-1, ROUGE-2, and ROUGE-L, respectively. Bold numbers indicate the highest score, while underlined numbers indicate the second-highest. * indicates fine-tuned models.

Model	Typical Generative Metrics					DGQS
	F1	BLEU	R-1.	R-2.	R-l.	
LLaMA2-7b	0.2195	**0.0112**	0.1207	0.0208	0.1026	0.4753
LLaMA3.1-8b	0.2143	0.0075	0.1277	0.0368	0.1183	0.5511
LLaMA3.1-70b	0.3067	0.0110	0.2316	0.0633	0.2059	0.6587
GPT-4o	0.3056	0.0104	0.2470	0.0797	0.2214	**0.6759**
DeepSeek-v3	**0.3792**	0.0105	**0.2865**	**0.0979**	**0.2608**	0.6737
*LLaMA2-Dictionary	0.2557	0.0097	0.2033	0.0588	0.1825	0.6318
*LLaMA3-Dictionary	0.2925	0.0101	0.1933	0.0541	0.1748	0.6128
*T5-definition	0.3103	0.0055	0.1103	0.0344	0.1064	0.5293

Table 18: DGQS and typical Generative metrics of five models on Senseval2. F1 represents the F1 score for relaxed matching, while R-1, R-2, and R-L are ROUGE-1, ROUGE-2, and ROUGE-L, respectively. Bold numbers indicate the highest score, while underlined numbers indicate the second-highest. * indicates fine-tuned models.

Model	Typical Generative Metrics					DGQS
	F1	BLEU	R-1.	R-2.	R-l.	
LLaMA2-7b	0.2012	**0.0129**	0.1136	0.0195	0.0983	0.4888
LLaMA3.1-8b	0.1815	0.0082	0.1114	0.0225	0.1028	0.5407
LLaMA3.1-70b	0.2944	0.0120	0.2292	0.0571	0.2006	0.6541
GPT-4o	0.2881	0.0120	0.2447	0.0740	0.2207	**0.6778**
DeepSeek-v3	**0.3583**	0.0124	**0.2753**	**0.0887**	**0.2515**	0.6729
*LLaMA2-Dictionary	0.2411	0.0110	0.2021	0.0659	0.1850	0.6260
*LLaMA3-Dictionary	0.2742	0.0116	0.1924	0.0591	0.1762	0.6135
*T5-definition	0.2831	0.0068	0.1283	0.0390	0.1239	0.5291

Table 19: DGQS and typical Generative metrics of five models on Senseval3. F1 represents the F1 score for relaxed matching, while R-1, R-2, and R-L are ROUGE-1, ROUGE-2, and ROUGE-L, respectively. Bold numbers indicate the highest score, while underlined numbers indicate the second-highest. * indicates fine-tuned models.

A.7 Fine-tuned WSD Models

When testing these fine-tuned models on other tasks, such as selecting correct options or determining whether disambiguation was possible, we observe that they struggle to properly comprehend different task descriptions. Specifically, the models either generate their own answers instead of selecting from the provided options, repeat the prompt verbatim, or produce null outputs. To quantify this, we evaluate the models on the DOCQ task using SemEval-2007 and categorize their output behaviors, as shown in table 21. Although T5-definition can correctly output option indices, its F1-score is only 0.385.

Due to the model's highly anomalous outputs, we discontinue further testing on additional datasets and tasks, shifting our focus to investigating the underlying causes of this phenomenon. Given that the model underwent fine-tuning specifically for definition generation tasks, its diminish generalization capability likely manifests in two key aspects: (1) Degradation in inherent disambiguation ability; (2) Severe escalation of prompt bias, where the model only comprehends the fine-tuning prompts and treats all tasks as definition-generation tasks. To examine these hypotheses, we first isolate instances where LLaMA3-Dictionary produce option outputs, calculate their accuracy, and compare them with the base model. Table 22 demonstrates that fine-tuning leads to a significant degradation in WSD capability. The definition generation task comprises two stages: (1) **Disambiguation:** Identifying the correct word sense; (2) **Generation:** Producing the corresponding definition. While fine-tuning enhances Stage 2 performance, this improvement masks the model's declining Stage 1 ability, a critical issue requiring attention.

Additionally, we observe that LLaMA2-Dictionary exclusively generate definitions as outputs. This raise a critical question: Has the model internally complete the selection process but output the textual content of options rather than their indices? To test this hypothesis, we change the evaluation metrics for the DOCQ task to those used in definition generation task and compare with the base model. Table 20 indicates that the model do not output the content of the provided options, but instead generate its own answers. The fine-tuned model fails to correctly comprehend the task requirements specified in the prompt and when we modify the prompt to explicitly follow DOCQ task descriptions, it paradoxically lead to degrade generation quality.

Models	BLEU	Rougle-1	Rougle-2	Rougle-L	DGQS
LLaMA2-Dictionary	0.012	0.013	0.043	0.012	0.398
Base model	0.014	0.095	0.011	0.084	0.483

Table 20: Performance Comparison Using Definition Generation Metrics on DOCQ Task (SemEval-2007)

To systematically investigate how prompt modifications affect fine-tuned models' performance on definition generation tasks, we first divide the prompt into a system prompt and a task description. Our system prompt is unified as "You are an expert in word sense disambiguation." Then, we combine our system prompt and task description with those used in the original papers to have the models complete the definition generation task. All prompts follow the structure: *[System Prompt]* + *[Task Description]*. The results on SemEval2007 are shown in table 23, where Original denotes the prompt segment used in the baseline paper, Ours is the prompt segment from our framework and Empty denotes the empty prompt segment or only containing a context. Since the T5-definition do not employ system prompts, we only evaluate three prompt variants for this model.

A comparison between LLaMA2-Dictionary and LLaMA3-Dictionary reveals significant differences in prompt sensitivity. LLaMA2-Dictionary shows marked performance degradation when using prompts that differ from the original formulation, regardless of which component is modified. In contrast, LLaMA3-Dictionary demonstrates substantially greater robustness to prompt variations. This indicates that LLaMA2-Dictionary developed stronger prompt bias during fine-tuning, exhibiting higher dependency on specific prompt content. Further analysis shows that both models are more sensitive to changes in task descriptions than system prompts, with performance declining more significantly when task description segments are altered. Interestingly, even when provide only with contextual information without any additional content or task descriptions, the models can still finish definition generation tasks, albeit at reduced capacity. This suggests that while prompt engineering significantly impacts performance, the fine-tuned models retain some baseline capability to handle the core task without explicit instructions.

Model	Total Instances	Behaviors		
		Option Outputs	Definition Outputs	Prompt regurgitation
LLaMA2-Dictionary	455	0	455	0
LLaMA3-Dictionary	455	77	204	174
T5-definition	455	455	0	0

Table 21: Model Output Behavior Categorization on the DOCQ Task (SemEval-2007)

Model	Option Outputs	Correct Number	Accuracy	Base Accuracy
LLaMA3-Dictionary	77	34	0.4416	0.5099

Table 22: Accuracy Comparison Between LLaMA3-Dictionary and Base Model on Valid Option Outputs (SemEval-2007)

Models	[System Prompt] + [Task Description]	BLEU	ROUGE-1	ROUGE-2	ROUGE-L	DGQS
LLaMA2-Dictionary	Original+Original	0.012	0.172	0.047	0.162	0.600
	Original+Ours	0.009	0.126	0.019	0.120	0.439
	Ours+Ours	0.011	0.141	0.022	0.128	0.487
	Ours+Original	0.012	0.170	0.045	0.160	0.610
	Empty+Original	0.011	0.151	0.035	0.147	0.593
	Original+Empty	0.015	0.069	0.006	0.065	0.244
	Empty+Empty	0.014	0.086	0.003	0.073	0.248
LLaMA3-Dictionary	Original+Original	0.012	0.153	0.034	0.141	0.576
	Original+Ours	0.010	0.167	0.022	0.152	0.571
	Ours+Ours	0.010	0.216	0.089	0.200	0.608
	Ours+Original	0.012	0.153	0.037	0.141	0.589
	Empty+Original	0.009	0.134	0.026	0.120	0.569
	Original+Empty	0.012	0.101	0.001	0.089	0.200
	Empty+Empty	0.011	0.047	0	0.044	0.215
T5-definition	/ + Original	0.008	0.124	0.040	0.121	0.548
	/ + Ours	0.008	0.080	0.017	0.079	0.438
	/ + Empty	0.003	0.005	0	0.004	0.245

Table 23: Performance Comparison of Definition Generation with Different Prompt Strategies across Models.

B Templates and Examples

This appendix presents the prompt templates for three task types (DOCQ, PWCJ, DG) along with an illustrative example demonstrating different metric scores in the DG task.

Prompt for DOCQ

Please select the option corresponding to the definition of the target word in the context from the candidate definitions.

The target word is represented in the context.

You only need to output the option number corresponding to the definition of the target word. Any additional output will reduce the quality of your answer.

Context: {context}
Options: {options}
Correct option:

Figure 20: Prompt for DOCQ task.

Prompt for PWCJ

Please check if you can determine the definition of the target word in the given context.

You do not need to give the specific definition of the target word, just check if you know its definition.

You should carefully consider it.
If you are very confident that you know the specific definition of the target word in the context, output Yes. Your answer is likely to be incorrect. If you have any uncertainty about the specific definition of the target word, you should output No.

Context: {context}
Target word: {target word}
Determine (Yes/No):

Figure 21: Prompt for PWCJ task.

Prompt for DG

You are now an expert in word sense disambiguation. Please determine the correct definition of the target word in the context.

Then output the correct definition of the target word.

You only need to output the definition of the target word. Any additional output will reduce the quality of your answer.

Context: {context}
Target word: {target word}
Correct definition:

Figure 22: Prompt for DG task.

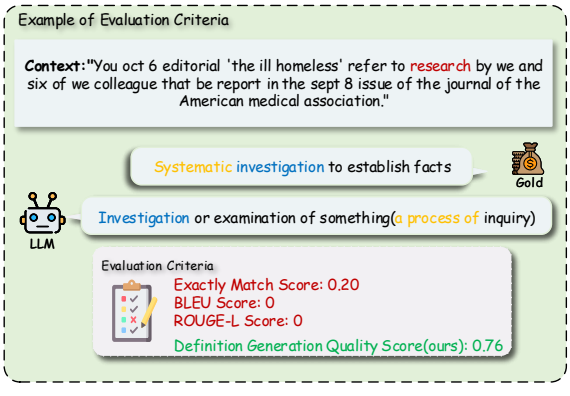


Figure 23: An example of DG task using different evaluation metrics.