Revisiting *a Pain in the Neck*: A Semantic Reasoning Benchmark for Language Models

Anonymous ACL submission

Abstract

We present SEMANTICQA, a comprehensive benchmark suite designed to assess language models (LMs) across ten semantic phrase (SP) processing tasks. Unlike prior benchmarks, it provides a unified evaluation setting encompassing both general SPs, such as lexical collocations (LC), and three fine-grained categories: idiomatic expressions (IE), noun compounds (NC), and verbal constructions (VC). We systematically evaluate LMs of diverse architectures and scales on classification, extraction, and interpretation tasks. Our results reveal substantial performance variation, particularly on tasks requiring compositional semantic reasoning, highlighting differences in LMs' reasoning capabilities and semantic understanding. These findings provide actionable insights for advancing the development of LMs with stronger SP comprehension. **The code, data, and models will be made publicly available upon completion of the review process.**

1 Introduction

Semantic phrases (SPs), also referred to as multiword expressions (MWE), are lexical combinations whose meanings or usages cannot be fully derived from their individual components (Pasquer et al., 2020). They exhibit varying degrees of compositionality, idiomaticity, and fixedness (Sailer and Markantonatou, 2018; Ramisch, 2023). Despite extensive research across supervised and unsupervised paradigms, robust SP processing remains a long-standing challenge in NLP (Sag et al., 2002; Constant et al., 2017a; Shwartz and Dagan, 2019; Ramisch et al., 2023a; Wada et al., 2023; Tanner and Hoffman, 2023).

Large language models (LLMs) are typically evaluated using benchmarks that emphasize mathematical reasoning (An et al., 2025; Balunović et al., 2025), code generation (Austin et al., 2021; Li et al., 2024), or symbolic reasoning (Xu et al., 2025).

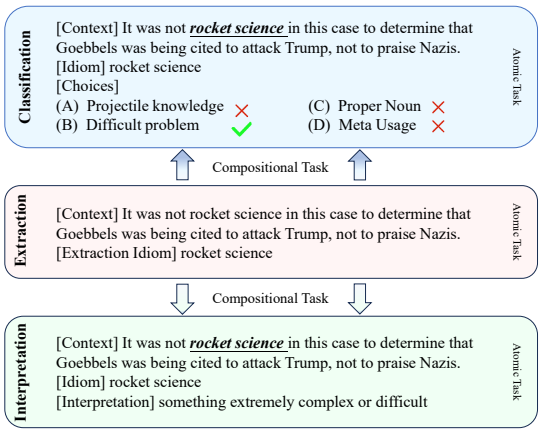

Figure 1: Examples of semantic operations for the Idiomatic Expressions (IE) task are provided, with additional details available in Appendix B.

While these benchmarks effectively assess logical reasoning and factual knowledge (Yu et al., 2024; Hu et al., 2024; Huang et al., 2025; Yu et al., 2025), they largely overlook fine-grained semantic-level reasoning that operates over sub-sentential units. In particular, phrase-level semantics, where meaning emerges from interactions between lexical components and context, remain underexplored, and when evaluated, are often assessed through isolated task formats that conflate multiple semantic operations. As a result, it is difficult to determine whether strong performance reflects stable phrase-level semantic representations or task-specific heuristics. Recent work has therefore called for diagnostic evaluation benchmarks that explicitly disentangle semantic operations and probe phrase-level semantic behavior beyond surface-level language understanding (Miletić and Walde, 2024).

We therefore ask: "How do language models (LMs) behave when evaluated on phrase-level semantics across distinct but structurally constrained semantic operations?" To address this question, we introduce SEMANTICQA, an operation-aligned evaluation benchmark for phrase-level semantic

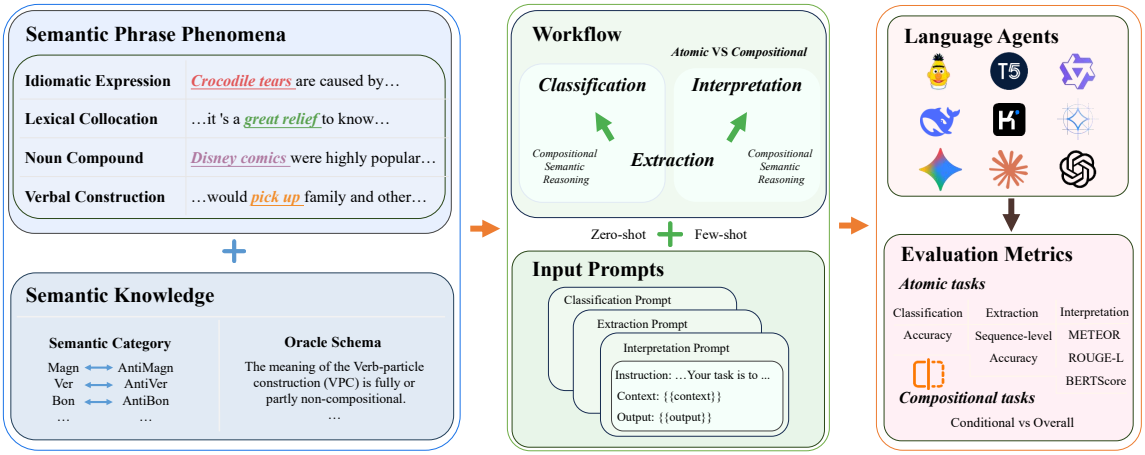

Figure 2: Overview of the Semantic Phrase Reasoning Benchmark.

processing.

In this work, we adopt a deliberately operationalized view of semantic reasoning at the benchmark level. Rather than requiring LMs to perform multiple semantic operations on the same instance, we examine whether phrase-level semantic understanding generalizes across datasets that instantiate different semantic operations. Specifically, we consider three atomic operations, including classification, extraction, and interpretation, which target the same underlying notion of phrase meaning while imposing distinct structural constraints on LM outputs. Under this formulation, semantic reasoning is assessed by a LM's ability to exhibit compatible performance patterns across operation-aligned tasks, reflecting whether learned phrase semantics transfer across different semantic interfaces rather than overfitting to a single task format.

Under this definition, high performance on any single task is insufficient evidence of semantic reasoning. Instead, semantic reasoning is evaluated through cross-operation consistency at the benchmark level, sensitivity to operation-specific structural constraints, and robustness under explicitly constructed compositional evaluation settings where applicable. Our contributions are threefold:

1. **Operation-Aligned Semantic Evaluation.** SEMANTICQA does not introduce new semantic theories but evaluates phrase-level competence through a set of controlled semantic operations with varying output structures and constraints. Its core contribution lies in **aligning existing SP tasks with the semantic operations they instantiate**, enabling systematic analysis of semantic behavior across structurally distinct yet related task families.

2. **Minimal and Controlled Benchmark Design.** SEMANTICQA employs fixed prompt templates to reduce prompt-induced variance across LMs. By holding prompt structure constant while varying semantic operations, the benchmark supports fair comparison under shared instructional conditions.

3. **Diagnostic Analysis of Compositional Sensitivity.** In explicitly designed compositional settings, we show that strong LLMs often fail to maintain semantic consistency across dependent operations, revealing phrase-level limitations that remain hidden in single-task evaluations.

2 Related Work

Complex Reasoning. Recent benchmarks have extensively evaluated LLMs across general language understanding (Hendrycks et al., 2020; Huang et al., 2023), mathematical reasoning (An et al., 2025; Balunović et al., 2025), code generation (Chen et al., 2021; Austin et al., 2021; Li et al., 2024), and symbolic reasoning(Xu et al., 2025). Many focus on structured inference over explicit representations, such as compositional procedures in math or rule-based symbolic tasks. While effective for formal reasoning, they largely overlook fine-grained semantic operations.While these benchmarks are effective at assessing formal reasoning and problem-solving skills, they primarily focus on structured inference over explicit representations (Liu et al., 2024; Balunović et al., 2025; An et al., 2025; Luong et al., 2025).

In contrast, natural language reasoning relies on phrase-level meaning composition, contextual disambiguation, implicit semantic-role inference, and paraphrase mapping. These operations require manipulating latent semantic representations rather than symbolic rules. Analyses show that even state-of-the-art LLMs often depend on shallow heuristics, emphasizing the need for benchmarks that directly test semantic-level reasoning (Yu et al., 2024; Huang et al., 2025).

Semantic Phrase Processing and Evaluation Resources. SP processing has long been studied in the context of MWEs, with early work focusing on unsupervised phrase representations and compositional modeling (Vacareanu et al., 2020; Arase and Tsujii, 2020). More recent studies have examined idiom identification, contextual paraphrasing, and noun compound interpretation using both fine-tuned models (e.g., T5) and large-scale transformers (e.g., GPT-5) (Klubička et al., 2023; Wada et al., 2023). In parallel, a wide range of datasets have been developed to evaluate phrase semantics, covering idiomatic expressions (Tedeschi et al., 2022; Zhou et al., 2021), collocations (Espinosa-Anke et al., 2019, 2021; Fisas et al., 2020; Espinosa-Anke et al., 2022), and verbal constructions (Savary et al., 2023; Ramisch et al., 2020).

Despite this progress, existing work and datasets typically isolate specific phrase types, task formats, or semantic phenomena, such as compositionality or literal, idiomatic distinctions, without explicitly modeling the atomic semantic operations underlying phrase understanding (Pham et al., 2023; Buijtelaar and Pezzelle, 2023; Zeng et al., 2023). As a result, evaluations are often conducted in isolation, limiting cross-task and cross-phenomenon analysis and obscuring how phrase-level semantic competence generalizes across structurally distinct settings. This fragmentation motivates SEMANTICQA, which provides a unified, operation-aligned evaluation benchmark for analyzing phrase-level semantic processing in LLMs.

3 SEMANTICQA: Semantic Phrase Processing Benchmark

3.1 Preliminaries

SPs exhibit diverse degrees of compositionality and conventionalization. We consider four representative phrase types that capture major sources of phrase-level semantic variation.

Idiomatic Expressions (IE) are prototypical non-compositional phrases whose meanings cannot be derived from their constituent words (e.g., *kick the bucket*). Processing such expressions requires LMs to recover conventionalized meanings beyond literal composition. (Zhou et al., 2022; Zeng and Bhat, 2022; Haviv et al., 2023).

Lexical Collocations (LC) form a broad class of SPs with varying degrees of compositionality. They are characterized by conventionalized lexical relations between a *base word* and a *collocate*, ranging from largely compositional combinations to idiom-like usages (Espinosa-Anke et al., 2021; Shvets and Wanner, 2022).

Noun Compounds (NC) are often compositional, but their interpretation frequently depends on implicit semantic relations, contextual cues, or world knowledge (e.g., *baby oil* vs. *olive oil*) (Kolluru et al., 2022; Coil and Shwartz, 2023).

Verbal Constructions (VC), or verbal multiword expressions (VMWE), including light-verb constructions (LVC), verb–particle constructions (VPC), and verbal idioms (VID), are typically semi-compositional (Tanner and Hoffman, 2023; Savary et al., 2023; Ramisch et al., 2023b). Their meanings arise from an interaction between literal composition and conventional usage.

Together, these categories provide a unified foundation for evaluating phrase-level semantic processing. Our benchmark assesses LMs' ability to perform semantic operations over these phrase types, both in isolation and in composition.

3.2 Task Definitions

We organize tasks by both phrase type and atomic semantic operation, where each operation targets a distinct aspect of semantic processing. This organization allows tasks operating on the same underlying phrase meaning to differ systematically in their output structure and constraints.

For IE, we include detection (IED), extraction (IEE), and interpretation (IEI). Detection is formulated as a multiple-choice classification task, extraction requires exact span identification, and interpretation evaluates contextualized paraphrase generation. All datasets are adapted from existing annotated resources (Harish et al., 2021; Tedeschi et al., 2022; Zhou et al., 2021), with overlapping instances deduplicated and reformatted to ensure consistency across operations.

Task	Data Source	Input ($\mathcal{I}$)	Output ($\mathcal{O}$)	Metrics	# Test Size	Phrase Type
IE Detection	Harish et al., 2021	$\mathcal{P} \oplus \mathcal{S} \oplus \mathcal{IE}$	Choice from *Options*	Acc	273	Idiomacity
IE Extraction	Tedeschi et al., 2022	$\mathcal{P} \oplus \mathcal{S}$	Extracted $\mathcal{IE}$	Acc_s	447	Idiomacity
IE Interpretation	Zhou et al., 2021; Chakrabarty et al., 2022	$\mathcal{P} \oplus \mathcal{S} \oplus \mathcal{IE}$	Interpretation of $\mathcal{IE}$	METEOR,Rouge-L, BERTScore	818	Idiomacity
LC Categorization	Espinosa-Anke et al., 2021	$\mathcal{P} \oplus \mathcal{T} \oplus \mathcal{S}$	Choice from *Options*	Acc	305	Collocation
LC Extraction	Fisas et al., 2020	$\mathcal{P} \oplus \mathcal{T} \oplus \mathcal{S}$	Extracted $\mathcal{LC}$	Acc_s	305	Collocation
LC Interpretation	Espinosa-Anke et al., 2019, 2021	$\mathcal{P} \oplus \mathcal{S} \oplus \mathcal{LC}$	Interpretation of $\mathcal{LC}$	METEOR,Rouge-L, BERTScore	305	Collocation
NC Compositionality	Garcia et al., 2021	$\mathcal{P} \oplus \mathcal{S} \oplus \mathcal{NC}$	Choice from *Options*	Acc	242	Noun Compound
NC Extraction	Garcia et al., 2021; Kolluru et al., 2022	$\mathcal{P} \oplus \mathcal{S}$	Extracted $\mathcal{NC}$	Acc_s	720	Noun Compound
NC Interpretation	Coil and Shwartz, 2023	$\mathcal{P} \oplus \mathcal{S} \oplus \mathcal{NC}$	Interpretation of $\mathcal{NC}$	METEOR,Rouge-L, BERTScore	110	Noun Compound
VMwE Extraction	Savary et al., 2023	$\mathcal{P} \oplus \mathcal{S}$	Extracted $\mathcal{VC}$	Acc_s	475	Verbal MwE

Table 1: A summary of the dataset statistics in SemanticQA. $\mathcal{P}$ refers to the prompt template, $\mathcal{S}$ the sentence context, $\mathcal{T}$ the semantic taxonomy narrative, $\mathcal{IE}$ idiomatic expressions, $\mathcal{LC}$ lexical collocations, and $\mathcal{NC}$ noun compounds.

Lexical Function	Example	Semantic Relation
Magn	Magn(*rain*) = *heavy*	"intense", "strong"
AntiMagn	AntiMagn(*accent*) = *slight*	"little", "weak"
Ver	Ver(*message*) = *clear*	"real", "genuine"
AntiVer	AntiVer(*accusation*) = *groundless*	"non-genuine"
Bon	Bon(*bread*) = *fresh*	"positive"
AntiBon	AntiBon(*advantage*) = *undue*	"negative"
Son	Son(alarm clock) = *ring(s)*	"sound", "voice"
Oper1	Oper1(*advice*) = *give*	"perform"

Table 2: Partial semantic relations involved in this paper, with their exemplars. More relations in lexical functions (LFs) can be referred to the Table 8.

For LC, we design categorization (LCC), extraction (LCE), and interpretation (LCI) tasks. Categorization requires predicting the semantic relation of a collocation under a lexical-function taxonomy (cf. Table 2). Extraction identifies both the *base word* and *collocate word* in context, while interpretation evaluates paraphrase generation conditioned on sentence context. Datasets are balanced across semantic relation categories to support controlled multi-class evaluation (Espinosa-Anke et al., 2021; Fisas et al., 2020; Espinosa-Anke et al., 2022, 2021).

For NC, we include compositionality classification (NCC), extraction (NCE), and interpretation (NCI), which respectively evaluate compositionality judgment, structural identification, and literal meaning reconstruction under context (Garcia et al., 2021; Kolluru et al., 2022; Coil and Shwartz, 2023; Hendrickx et al., 2013).

Finally, we include VMwE extraction task , which requires identifying a single verbal construction in context, covering VPC (VPE), LVC (LVE), and VID (VIE) (Savary et al., 2023).

We formalize SP processing as a conditional generation problem under operation-aligned constraints. Given a prompt template $\mathcal{P}$ that specifies a target semantic operation and a SP embedded in its context $\mathcal{S}$, a LM is required to generate an output $\mathcal{O}$ that satisfies the instruction induced by $\mathcal{P}$.

Concretely, the model input is constructed as $\mathcal{I} := \mathcal{P} \oplus \mathcal{S}$, where $\oplus$ denotes a task-specific composition of instruction and contextualized phrase. The output $\mathcal{O}$ varies according to the semantic operation being evaluated. For example, in extraction tasks, $\mathcal{O}$ corresponds to the target phrase span identified from $\mathcal{S}$ under the constraints specified by $\mathcal{P}$, whereas in classification or interpretation tasks, $\mathcal{O}$ represents a semantic decision or reconstruction aligned with the given instruction.

For each task, the configuration of the tuple $(\mathcal{P}, \mathcal{S}, \mathcal{O})$ is instantiated according to a fixed template, as summarized in Table 1. Each dataset is defined as $\mathcal{D} := \{(p_i, s_i, o_i)\}_{i=1}^{N}$, where each example pairs a prompt, a contextualized SP, and a gold-standard output corresponding to the target semantic operation.

3.3 Evaluation Metrics

We adopt task-appropriate automatic metrics aligned with the output space of each semantic operation. Classification tasks are evaluated using exact match[1] (EM) or accuracy (Acc). Extraction

[1]We apply heuristic rules to normalize and parse LLM outputs according to their response formats.

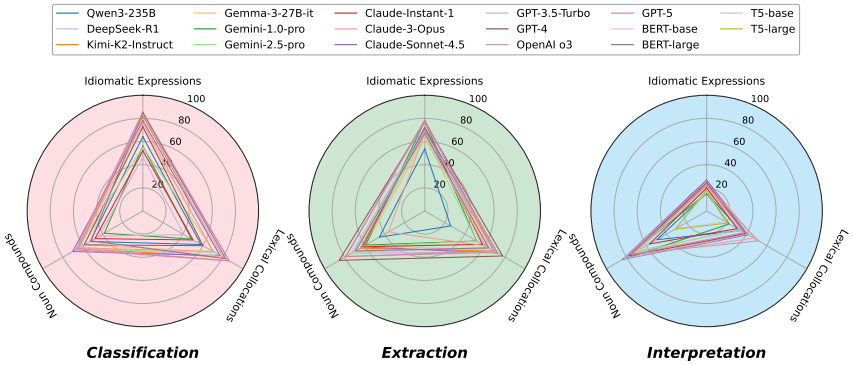

Figure 3: Overall the best performance (i.e., capacity triangle △) of models on SEMANTICQA

tasks are evaluated using sequence-level accuracy (ACC_s), which requires exact recovery of the target phrase and avoids score inflation from partial matches. Interpretation tasks are evaluated using METEOR (MTR) (Denkowski and Lavie, 2014) as the primary metric, with ROUGE-L (R-L) (Lin, 2004), and BERTScore (B-S) (Zhang et al., 2019) reported for complementary analysis.

4 Experimental Setup

Datasets. SEMANTICQA is built upon datasets drawn from multiple prior resources (Harish et al., 2021; Espinosa-Anke et al., 2022; Garcia et al., 2021; Savary et al., 2023), which vary in annotation protocols, difficulty distributions, and semantic granularity . Rather than enforcing uniform difficulty or annotation consistency across sources, the benchmark is designed to reflect this variation and is not intended for absolute performance comparisons across phrase types or tasks. Our analysis focuses on within-task and within-dataset trends, as well as relative changes induced by different semantic operations and compositional evaluation settings. Accordingly, claims about semantic reasoning are grounded in performance patterns that are stable across multiple tasks and datasets, rather than in absolute score levels. All experiments use the datasets described in Table 1 and §B.

Models. We evaluate a diverse set of LMs spanning different architectures, scales, and reasoning capabilities, including GPT-5 (OpenAI, 2025a), OpenAI o3 (OpenAI, 2025b), GPT-4 (OpenAI, 2023), Claude-Sonnet-4.5 (Anthropic, 2025), Gemini-2.5-Pro (Google, 2025), Claude-3-Opus (Anthropic, 2024), DeepSeek-R1 (DeepSeek-AI, 2025), Qwen3-235B (Qwen Team, 2025), Gemma-3-27B-it (Gemma Team, 2025), and Kimi-K2-Instruct (Kimi Team, 2025), BERT-base/large (Devlin et al., 2019), and T5-base/large (Raffel et al., 2020), as summarized in Figure 3 and Table 12.

5 Results

5.1 Benchmarking Results

Overall Performance Patterns. Table 3 and Figure 3 show substantial variation across semantic operations and phrase types. Even within the same semantic phenomenon (e.g., IE or LC), models behave differently on classification, extraction, and interpretation, indicating that these operations impose distinct structural and semantic constraints. No model performs uniformly well across all settings, suggesting operation-specific strengths and weaknesses rather than a single transferable notion of phrase-level competence. Moreover, SEMANTICQA is neither saturated nor uniformly difficult: different tasks expose complementary failure modes, supporting its use as a diagnostic benchmark rather than a leaderboard driven by aggregate scores.

Effect of In-Context Learning (ICL). The impact of ICL varies by task type (cf. Tables 3 and 4). Interpretation tasks benefit most consistently from few-shot prompting. Across IEI, LCI, and NCI, three- or five-shot demonstrations yield clear gains in MTR. However, as shown in Table 4, complementary metrics reveal that these improvements primarily reflect exemplar-guided reconstruction rather than strict semantic grounding, as embedding-based similarity can be high even when lexical overlap remains limited. Few-shot ICL improves both R-L and B-S, but gains vary by phrase type, reflecting the inherent underspecification of interpretation outputs.

For classification tasks, ICL exhibits mixed effects. Models with weaker zero-shot performance

Model	Idiom			Collocation			Noun Compound			VMWE		
	IED	IEE	IEI	LCC	LCE	LCI	NCC	NCE	NCI	VPE	LVE	VIE
Metric (%)	Acc	Acc_s	MTR	Acc	Acc_s	MTR	Acc	Acc_s	MTR	Acc_s	Acc_s	Acc_s
Human	71.0	**87.0**	20.5	47.0	50.0	16.7	**71.0**	73.0	17.2	**85.0**	**55.0**	**78.0**
DeepSeek-R1: *zero-shot*	71.1	69.4	**12.4**	66.6	**31.5**	**31.8**	60.2	51.3	**31.4**	76.8	26.7	50.5
↪ + *three-shot*	79.1	70.6	19.4	76.4	55.6	33.6	62.7	66.3	68.3	74.7	26.7	**59.1**
↪ + *five-shot*	84.3	72.3	19.2	76.1	64.3	32.9	60.6	70.7	68.7	81.6	35.8	57.1
Kimi-K2-Instruct: *zero-shot*	68.5	63.1	13.9	68.5	34.4	33.7	60.6	45.4	65.4	**55.8**	28.9	46.7
↪ + *three-shot*	77.7	68.9	23.5	79.0	67.9	39.1	59.3	64.4	71.4	79.5	39.4	43.8
↪ + *five-shot*	81.7	69.6	21.7	79.7	69.2	36.9	64.7	63.6	76.7	81.1	**43.3**	46.7
Gemma-3-27B-it: *zero-shot*	**55.0**	**57.3**	13.5	**58.0**	38.4	35.0	58.3	**39.9**	43.8	66.8	19.4	38.1
↪ + *three-shot*	69.6	62.0	19.9	70.1	63.7	37.3	56.7	57.2	68.3	74.1	28.3	45.7
↪ + *five-shot*	72.1	61.6	19.2	70.8	68.2	38.7	56.2	59.2	70.5	70.5	35.0	52.4
Claude-Sonnet-4.5: *zero-shot*	72.5	68.5	17.0	67.5	40.1	34.8	**51.0**	45.1	77.2	69.8	**16.1**	**41.9**
↪ + *three-shot*	77.7	72.0	25.8	77.1	70.5	41.2	61.4	59.3	81.2	76.8	30.6	42.9
↪ + *five-shot*	78.0	72.0	**26.7**	76.1	**72.7**	40.8	**70.1**	62.1	**83.8**	**82.0**	37.2	47.6
OpenAI o3 : *zero-shot*	57.1	65.1	12.6	72.1	37.7	35.9	65.2	62.9	45.7	67.9	25.6	51.4
↪ + *three-shot*	79.5	77.4	21.3	85.9	65.3	**41.6**	58.9	77.5	68.2	76.3	29.1	52.4
↪ + *five-shot*	83.5	74.7	21.9	83.6	71.5	35.9	63.5	78.6	74.5	77.3	36.9	50.0
GPT-5: *zero-shot*	82.8	67.6	13.9	75.4	36.7	33.7	66.8	64.3	57.3	74.2	28.9	56.2
↪ + *three-shot*	82.1	78.3	22.6	**86.2**	67.2	35.4	61.8	77.1	70.1	74.7	33.3	51.4
↪ + *five-shot*	**85.4**	**78.7**	22.5	84.3	68.9	37.4	67.2	**79.0**	75.3	74.7	38.3	50.5

Table 3: Major experimental results in SemanticQA. **Digits** highlight cases in which human scores are higher than those of all evaluated models, serving as a coarse reference. Light Green and Light Blue parts present open-source models and proprietary models.

System	IEI		LCI		NCI	
	R-L	B-S	R-L	B-S	R-L	B-S
DeepSeek-R1	**14.7**	**85.1**	42.0	90.2	**37.6**	**91.3**
↪ 3-shot	25.2	88.1	44.9	91.6	73.0	96.3
↪ 5-shot	25.0	88.0	44.6	91.8	75.5	96.6
Kimi-K2-Instruct	18.8	86.7	**40.2**	90.3	68.9	95.6
↪ 3-shot	**27.9**	88.4	**52.2**	92.8	77.8	96.7
↪ 5-shot	26.4	88.2	48.3	**97.2**	**83.7**	**97.2**
OpenAI o3	17.3	86.5	41.5	**89.8**	49.9	93.8
↪ 3-shot	26.2	88.5	51.2	92.6	71.2	96.0
↪ 5-shot	26.8	88.6	44.8	91.6	76.5	96.5
GPT-5	19.2	86.6	40.6	89.9	56.4	93.3
↪ 3-shot	27.5	**88.7**	46.9	92.3	70.9	96.4
↪ 5-shot	27.1	88.6	47.7	92.3	77.7	96.8

Table 4: Interpretation task results on IEI, LCI, and NCI. We report ROUGE-L (R-L) and BERTScore (B-S) scores. Green indicates the best performance and red indicates the worst performance within each column.

often improve, whereas others plateau or regress, such as OpenAI o3 on LCC and NCC, indicating sensitivity to exemplar selection and task formulation. Extraction tasks are the most unstable under ICL. While demonstrations can substantially improve performance when span structure is clearly illustrated, performance may degrade when test instances diverge from the demonstrated patterns. Overall, ICL is consistently beneficial for interpretation, variably effective for classification, and highly task-dependent for extraction.

5.2 Human Performance

We estimate human performance using annotations from three linguistics graduate students, each labeling 100 randomly sampled examples per task in SemanticQA, following a two-stage protocol inspired by SuperGLUE (Sarlin et al., 2020). Human scores are reported as a coarse reference rather than an upper bound on performance (cf. Table 3). Differences between human and model results may arise from metric properties, task ambiguity, or output normalization effects, especially for interpretation tasks. Accordingly, we avoid strong claims based on absolute human–model comparisons. Instead, human performance is used to contextualize task difficulty and to illustrate evaluation challenges under varying output constraints.

5.3 Semantic Category Scaling with In-Context Learning

To examine how LMs encode semantic distinctions among lexical relations, we evaluate performance on the LCC task under an increasing number of target categories. We construct a controlled scaling setup by varying the category size from 1 to 16, and evaluate supervised baseline models and four representative LLMs under zero-shot and few-shot settings. Overall results are shown in Figures 5 and 18.

Across all settings, models consistently outperform random and majority baselines, indicating non-trivial semantic reasoning even without demonstrations. Accuracy decreases as the number of categories grows, but the degradation rate varies substantially across model families. Supervised

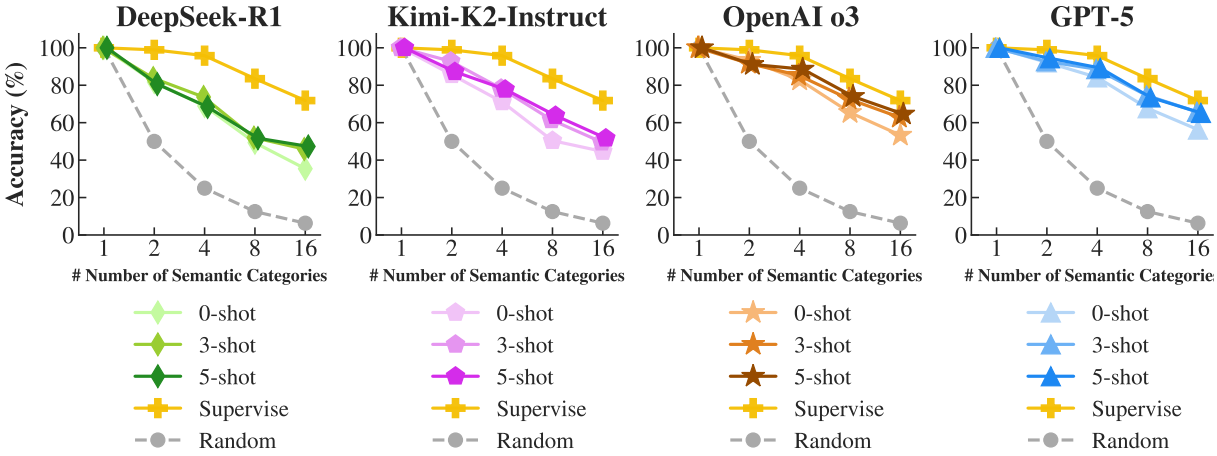

Figure 5: The ability of semantic relation categorization of $\mathcal{LC}$ with different numbers of in-context exemplars and semantic category scale. The number n of classes is chosen from $N := \{1, 2, 4, 8, 16\}$. Each model is prompted with the k-shot settings, where $k \in \{0, 3, 5\}$, respectively. Accuracy scores are calculated by the mean values based on 30 examples sampled per class from the test split of (Espinosa-Anke et al., 2021), partial categories $(n \leq 8)$ are run with three-class combinations in random selection, finally result in the mean value as the average.

baselines remain relatively stable, whereas LLMs exhibit sharper drops at larger scales. For example, DeepSeek-R1 decreases from 81.7% to 35.4% as category size increases, suggesting that in-context semantic reasoning alone does not fully substitute for explicit supervised signals when fine-grained relational distinctions are required.

5.4 Compositional Semantic Reasoning

To approximate realistic semantic processing workflows, we evaluate **Compositional Semantic Reasoning**, where models must perform multiple *dependent* semantic operations in sequence, such as extraction followed by interpretation or categorization. Tables 5 and 6 report results for compositional interpretation and classification setting (Ram et al., 2024; Alazraki et al., 2025).

For interpretation, conditional performance (Cond MTR) on correctly extracted phrases is consistently lower than overall scores (Overall MTR) and shows only limited gains from few-shot prompting across both IE and LC settings. This gap indicates that accurate extraction remains a primary bottleneck for downstream interpretation, and that fluent semantic reconstruction does not reliably compensate for upstream structural errors. Compositional classification degrades more sharply as task complexity increases. While leading models perform well in four-class LC settings, accuracy drops substantially in eight- and sixteen-class scenarios, with similar trends observed for IE and NC. Few-shot prompting partially mitigates this degradation but does not remove the strong dependence

Type	Model	Setting	Cond MTR	Overall MTR
LC	DeepSeek -R1	*0-shot*	**35.8**	10.0
		3-shot	38.8	13.4
		5-shot	**42.3**	14.3
	GPT-5	*0-shot*	37.6	**9.9**
		3-shot	40.1	15.9
		5-shot	41.8	**17.3**
IE	DeepSeek -R1	*0-shot*	**12.0**	**6.2**
		3-shot	13.0	7.4
		5-shot	13.4	7.6
	GPT-5	*0-shot*	**17.4**	8.4
		3-shot	17.2	9.6
		5-shot	17.1	**10.1**

Table 5: Performance comparison on combined extraction-interpretation tasks. Cond MTR evaluates interpretation of correctly extracted phrases; Overall MTR reflects end-to-end performance.

on extraction quality. Overall, performance drops in compositional settings should be viewed as a diagnostic signal rather than evidence of complex error propagation. They indicate that current models struggle to robustly integrate intermediate semantic outputs, even when individual operations perform well in isolation. By separating atomic semantic operations from their compositions, SEMANTICQA exposes a persistent gap between performance on isolated atomic tasks and the stability of end-to-end semantic pipelines.

5.5 VMWE Extraction via ORACLE SCHEMA

We analyze prompting strategies for VMWE extraction under zero-shot and few-shot ICL settings and introduce ORACLE SCHEMA, which augments task instructions with the target type and its definition (cf. Appendix §D). Table 7 shows that this strategy consistently improves performance across

Type	Model	0-shot		3-shot		5-shot	
		Cond	Overall	Cond	Overall	Cond	Overall
LC	DeepSeek-R1						
	4-class	73.4	36.4	74.9	44.2	80.5	44.4
	8-class	56.1	26.7	79.7	39.2	71.7	38.8
	16-class	**34.7**	**16.0**	51.0	25.6	54.5	27.7
	GPT-5						
	4-class	**91.3**	45.7	89.9	**58.1**	89.9	55.0
	8-class	76.2	38.8	83.3	50.0	80.3	49.2
	16-class	63.4	33.1	69.4	43.4	73.4	44.8
IE	DeepSeek-R1						
	4-class	63.8	46.5	65.0	46.9	**61.9**	**45.8**
	GPT-5						
	4-class	79.3	65.9	77.7	**66.3**	**79.7**	65.9
NC	DeepSeek-R1						
	4-class	**63.5**	**33.2**	**71.2**	36.9	71.1	37.8
	GPT-5						
	4-class	68.8	36.5	64.7	37.3	66.9	**38.6**

Table 6: Classification performance comparison. Cond: accuracy given correct extraction; Overall: end-to-end accuracy.

System	Setting	w/ ORACLE		w/o ORACLE
		ACC(Δ) ↑		ACC ↑
Deepseek-R1	*0-shot*	64.1 (+12.5)	⟷	51.6
	3-shot	**72.3 (+8.9)**	⟷	63.4
	5-shot	70.5 (+1.2)	⟷	69.3
Kimi-K2-Instruct	*0-shot*	53.3 (+9.1)	⟷	44.2
	3-shot	67.6(+1.1)	⟷	66.5
	5-shot	**69.5 (+3.8)**	⟷	65.7
OpenAI o3	*0-shot*	54.8 (+6.9)	⟷	47.9
	3-shot	67.3 (+5.1)	⟷	62.2
	5-shot	**70.7 (+3.6)**	⟷	67.1
GPT-5	*0-shot*	59.6 (+7.6)	⟷	52.0
	3-shot	66.8 (+5.1)	⟷	61.7
	5-shot	**72.6(+6.9)**	⟷	65.7

Table 7: We report the accuracy of four representative LLMs on the VMWE extraction task under different ICL settings, both with and without ORACLE SCHEMA.

models. For example, DeepSeek-R1 increases from 51.6% to 64.1%, demonstrating that providing explicit semantic descriptions of the target expression substantially enhances VMWE extraction.

6 Discussions and Takeaways

Rather than restating performance trends, we distill what operation-aligned evaluation reveals about the assessment and modeling of semantics.

Phrase Semantics Requires Multi-Dimensional Evaluation. Our results show that phrase-level semantic competence cannot be captured by any single task or metric. Interpretation, extraction, and categorization probe distinct aspects of semantic processing and differ substantially in structural constraint. While extraction and categorization require explicit grounding in linguistic structure or semantic relations, interpretation operates in a weakly constrained output space. Consequently, performance on open-ended interpretation alone risks conflating fluent semantic generation with structurally grounded understanding.

Metric Sensitivity Shapes Apparent Model Strengths. The contrast between strong interpretation scores (B-S, cf. Table 12) and weaker extraction performance highlights how evaluation metrics shape perceived model capabilities. Flexible similarity-based metrics used for interpretation primarily reward paraphrasing ability and instruction-following behavior, whereas strict span-based evaluations expose brittleness in structural grounding. As a result, high interpretation scores should be interpreted as evidence of improved exemplar-guided semantic reconstruction rather than conclusive semantic correctness. This discrepancy suggests that current evaluation practices may overestimate semantic robustness when structural constraints are not explicitly enforced.

Compositional Robustness Remains Limited. Compositional evaluations further reveal that semantic pipelines are highly sensitive to upstream errors. Interpretation does not reliably compensate for failures in extraction or categorization; instead, structural errors propagate and often remain undetected under flexible metrics. This lack of robustness under error accumulation remains a key challenge for realistic semantic applications and is largely obscured by atomic benchmarks.

7 Conclusions

We introduce SEMANTICQA, a benchmark for evaluating phrase-level semantic processing across diverse LMs. We conduct evaluations on a broad range of models using automated metrics, complemented by targeted human comparisons across ten tasks. The results show that, despite strong performance on general benchmarks, LMs continue to face substantial challenges on SEMANTICQA, revealing persistent limitations in SP understanding. Our analysis further characterizes model behavior across task types and highlights directions for future research on more robust and structurally grounded semantic processing.

Limitations

This work has several limitations that suggest directions for future research. First, although SEMANTICQA covers four common phrase phenomena, it is restricted to English and does not capture the

long tail of SP types, such as multiword named entities or complex function words (Constant et al., 2017b; Miletić and Walde, 2024). Second, while multiple task formats are included, future benchmarks should incorporate more complex compositional semantic reasoning and additional evaluation paradigms, such as semantic retrieval (Espinosa-Anke et al., 2021; Pham et al., 2023). Finally, although we evaluate lots representative models, rapid progress in LLM architectures calls for continual updates and broader coverage. We encourage future work to extend SEMANTICQA toward more comprehensive and multilingual resources (Espinosa-Anke et al., 2019).

Ethical Considerations

This research uses publicly available datasets in accordance with their original licenses and does not include any private, sensitive, or personally identifiable information. The benchmark is intended solely for research and diagnostic purposes, and known limitations are explicitly documented to avoid overgeneralization. Computational resources were used responsibly, and potential risks related to data misuse and model evaluation were considered. Where human annotations were involved, annotators were recruited under fair labor practices and received appropriate compensation.

References

Lisa Alazraki, Lihu Chen, Ana Brassard, Joe Stacey, Hossein A. Rahmani, and Marek Rei. 2025. Agentcoma: A compositional benchmark mixing commonsense and mathematical reasoning in real-world scenarios. *Preprint*, arXiv:2508.19988.

Shengnan An, Xunliang Cai, Xuezhi Cao, Xiaoyu Li, Yehao Lin, Junlin Liu, Xinxuan Lv, Dan Ma, Xuanlin Wang, Ziwen Wang, and Shuang Zhou. 2025. Amo-bench: Large language models still struggle in high school math competitions. *Preprint*, arXiv:2510.26768.

Anthropic. 2024. The claude 3 model family: Opus, sonnet, haiku. In *Anthropic Blog*.

Anthropic. 2025. Anthropic. `https://www.anthropic.com/news/claude-sonnet-4-5`. September 30, 2025.

Yuki Arase and Jun'ichi Tsujii. 2020. Compositional phrase alignment and beyond. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 1611–1623, Online. Association for Computational Linguistics.

Jacob Austin, Augustus Odena, Maxwell Nye, Maarten Bosma, Henryk Michalewski, David Dohan, Ellen Jiang, Carrie J. Cai, Michael Terry, Quoc V. Le, and Charles Sutton. 2021. Program synthesis with large language models. *ArXiv*, abs/2108.07732.

Mislav Balunović, Jasper Dekoninck, Ivo Petrov, Nikola Jovanović, and Martin Vechev. 2025. Matharena: Evaluating llms on uncontaminated math competitions.

Ekaba Bisong. 2019. Google colaboratory. *Building machine learning and deep learning models on google cloud platform: a comprehensive guide for beginners*, pages 59–64.

Lars Buijtelaar and Sandro Pezzelle. 2023. A psycholinguistic analysis of BERT's representations of compounds. In *Proceedings of the 17th Conference of the European Chapter of the Association for Computational Linguistics*, pages 2230–2241, Dubrovnik, Croatia. Association for Computational Linguistics.

Tuhin Chakrabarty, Yejin Choi, and Vered Shwartz. 2022. It's not rocket science: Interpreting figurative language in narratives. *Transactions of the Association for Computational Linguistics*, 10:589–606.

Mark Chen, Jerry Tworek, Heewoo Jun, Qiming Yuan, Henrique Ponde, Jared Kaplan, Harrison Edwards, Yura Burda, Nicholas Joseph, Greg Brockman, Alex Ray, Raul Puri, Gretchen Krueger, Michael Petrov, Heidy Khlaaf, Girish Sastry, Pamela Mishkin, Brooke Chan, Scott Gray, and 34 others. 2021. Evaluating large language models trained on code. *ArXiv*, abs/2107.03374.

Albert Coil and Vered Shwartz. 2023. From chocolate bunny to chocolate crocodile: Do language models understand noun compounds? In *Findings of the Association for Computational Linguistics: ACL 2023*, pages 2698–2710, Toronto, Canada. Association for Computational Linguistics.

Mathieu Constant, Gülşen Eryiğit, Johanna Monti, Lonneke Van Der Plas, Carlos Ramisch, Michael Rosner, and Amalia Todirascu. 2017a. Multiword expression processing: A survey. *Computational Linguistics*, 43(4):837–892.

Mathieu Constant, Gülşen Eryiğit, Johanna Monti, Lonneke van der Plas, Carlos Ramisch, Michael Rosner, and Amalia Todirascu. 2017b. Survey: Multiword expression processing: A Survey. *Computational Linguistics*, 43(4):837–892.

DeepSeek-AI. 2025. Deepseek-r1 incentivizes reasoning in llms through reinforcement learning. *Nature*, 645:633–638.

Michael Denkowski and Alon Lavie. 2014. Meteor universal: Language specific translation evaluation for any target language. In *Proceedings of the Ninth Workshop on Statistical Machine Translation*, pages 376–380, Baltimore, Maryland, USA. Association for Computational Linguistics.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. Bert: Pre-training of deep bidirectional transformers for language understanding. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 4171–4186.

Luis Espinosa-Anke, Joan Codina-Filba, and Leo Wanner. 2021. Evaluating language models for the retrieval and categorization of lexical collocations. In *Proceedings of the 16th Conference of the European Chapter of the Association for Computational Linguistics: Main Volume*, pages 1406–1417, Online. Association for Computational Linguistics.

Luis Espinosa-Anke, Steven Schockaert, and Leo Wanner. 2019. Collocation classification with unsupervised relation vectors. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 5765–5772, Florence, Italy. Association for Computational Linguistics.

Luis Espinosa-Anke, Alexander Shvets, Alireza Mohammadshahi, James Henderson, and Leo Wanner. 2022. Multilingual extraction and categorization of lexical collocations with graph-aware transformers. In *Proceedings of the 11th Joint Conference on Lexical and Computational Semantics*, pages 89–100, Seattle, Washington. Association for Computational Linguistics.

Beatriz Fisas, Luis Espinosa-Anke, Joan Codina-Filbá, and Leo Wanner. 2020. CollFrEn: Rich bilingual English–French collocation resource. In *Proceedings of the Joint Workshop on Multiword Expressions and Electronic Lexicons*, pages 1–12, online. Association for Computational Linguistics.

Thierry Fontenelle. 1997. *Turning a bilingual dictionary into a lexical-semantic database*. De Gruyter.

Marcos Garcia, Tiago Kramer Vieira, Carolina Scarton, Marco Idiart, and Aline Villavicencio. 2021. Assessing the representations of idiomaticity in vector models with a noun compound dataset labeled at type and token levels. In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, pages 2730–2741, Online. Association for Computational Linguistics.

Alexander Gelbukh and 1 others. 2012. *Semantic analysis of verbal collocations with lexical functions*, volume 414. Springer.

Gemma Team. 2025. Gemma 3.

Google. 2025. Google. `https://blog.google/technology/google-deepmind/gemini-model-thinking-updates-march-2025`. Mar 25, 2025.

Tayyar Madabushi Harish, Gow-Smith Edward, Scarton Carolina, and Villavicencio Aline. 2021. AStitchInLanguageModels: Dataset and methods for the exploration of idiomaticity in pre-trained language models. In *Findings of the Association for Computational Linguistics: EMNLP 2021*, pages 3464–3477, Punta Cana, Dominican Republic. Association for Computational Linguistics.

Adi Haviv, Ido Cohen, Jacob Gidron, Roei Schuster, Yoav Goldberg, and Mor Geva. 2023. Understanding transformer memorization recall through idioms. In *Proceedings of the 17th Conference of the European Chapter of the Association for Computational Linguistics*, pages 248–264, Dubrovnik, Croatia. Association for Computational Linguistics.

Iris Hendrickx, Zornitsa Kozareva, Preslav Nakov, Diarmuid Ó Séaghdha, Stan Szpakowicz, and Tony Veale. 2013. SemEval-2013 task 4: Free paraphrases of noun compounds. In *Second Joint Conference on Lexical and Computational Semantics (*SEM), Volume 2: Proceedings of the Seventh International Workshop on Semantic Evaluation (SemEval 2013)*, pages 138–143, Atlanta, Georgia, USA. Association for Computational Linguistics.

Dan Hendrycks, Collin Burns, Steven Basart, Andy Zou, Mantas Mazeika, Dawn Song, and Jacob Steinhardt. 2020. Measuring massive multitask language understanding. In *International Conference on Learning Representations*.

Ari Holtzman, Jan Buys, Li Du, Maxwell Forbes, and Yejin Choi. 2019. The curious case of neural text degeneration. In *International Conference on Learning Representations*.

Xuming Hu, Junzhe Chen, Xiaochuan Li, Yufei Guo, Lijie Wen, Philip Yu, and Zhijiang Guo. 2024. Towards understanding factual knowledge of large language models. In *International Conference on Representation Learning*, volume 2024, pages 28680–28715.

Sirui Huang, Yanggan Gu, Zhonghao Li, Xuming Hu, Li Qing, and Guandong Xu. 2025. StructFact: Reasoning factual knowledge from structured data with large language models. In *Findings of the Association for Computational Linguistics: ACL 2025*, pages 7521–7552, Vienna, Austria. Association for Computational Linguistics.

Yuzhen Huang, Yuzhuo Bai, Zhihao Zhu, Junlei Zhang, Jinghan Zhang, Tangjun Su, Junteng Liu, Chuancheng Lv, Yikai Zhang, Jiayi Lei, Yao Fu, Maosong Sun, and Junxian He. 2023. C-eval: A multi-level multi-discipline chinese evaluation suite for foundation models. In *Advances in Neural Information Processing Systems*.

Kimi Team. 2025. Kimi k2: Open agentic intelligence. *Preprint*, arXiv:2507.20534.

Filip Klubička, Vasudevan Nedumpozhimana, and John Kelleher. 2023. Idioms, probing and dangerous things: Towards structural probing for idiomaticity in

vector space. In *Proceedings of the 19th Workshop on Multiword Expressions (MWE 2023)*, pages 45–57, Dubrovnik, Croatia. Association for Computational Linguistics.

Olga Kolesnikova. 2020. Automatic detection of lexical functions in context. *Computación y sistemas*, 24(3):1337–1352.

Keshav Kolluru, Gabriel Stanovsky, and Mausam. 2022. "covid vaccine is against covid but Oxford vaccine is made at Oxford!" semantic interpretation of proper noun compounds. In *Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing*, pages 10407–10420, Abu Dhabi, United Arab Emirates. Association for Computational Linguistics.

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

Jia Li, Ge Li, Xuanming Zhang, Yunfei Zhao, Yihong Dong, Zhi Jin, Binhua Li, Fei Huang, and Yongbin Li. 2024. Evocodebench: An evolving code generation benchmark with domain-specific evaluations. In *Advances in Neural Information Processing Systems*, volume 37, pages 57619–57641. Curran Associates, Inc.

Chin-Yew Lin. 2004. ROUGE: A package for automatic evaluation of summaries. In *Text Summarization Branches Out*, pages 74–81, Barcelona, Spain. Association for Computational Linguistics.

Hongwei Liu, Zilong Zheng, Yuxuan Qiao, Haodong Duan, Zhiwei Fei, Fengzhe Zhou, Wenwei Zhang, Songyang Zhang, Dahua Lin, and Kai Chen. 2024. Mathbench: Evaluating the theory and application proficiency of llms with a hierarchical mathematics benchmark. *Preprint*, arXiv:2405.12209.

Thang Luong, Dawsen Hwang, Hoang H Nguyen, Golnaz Ghiasi, Yuri Chervonyi, Insuk Seo, Junsu Kim, Garrett Bingham, Jonathan Lee, Swaroop Mishra, Alex Zhai, Huiyi Hu, Henryk Michalewski, Jimin Kim, Jeonghyun Ahn, Junhwi Bae, Xingyou Song, Trieu Hoang Trinh, Quoc V Le, and Junehyuk Jung. 2025. Towards robust mathematical reasoning. In *Proceedings of the 2025 Conference on Empirical Methods in Natural Language Processing*, pages 35406–35430, Suzhou, China. Association for Computational Linguistics.

Igor A. Mel'čuk. 1998. Collocations and lexical functions. *Phraseology. Theory, analysis, and applications*, pages 23–53.

Igor A. Mel'čuk. 2023. *General phraseology: Theory and practice*. John Benjamins.

Filip Miletić and Sabine Schulte im Walde. 2024. Semantics of multiword expressions in transformer-based models: A survey. *arXiv preprint arXiv:2401.15393*.

OpenAI. 2023. Gpt-4 technical report. https://arxiv.org/pdf/2303.08774.pdf. *Preprint*, arXiv:2303.08774.

OpenAI. 2025a. Openai. https://openai.com/index/introducing-gpt-5. Accessed: August 7, 2025.

OpenAI. 2025b. Openai. https://openai.com/index/introducing-o3-and-o4-mini. April 16, 2025.

Caroline Pasquer, Agata Savary, Carlos Ramisch, and Jean-Yves Antoine. 2020. Verbal multiword expression identification: Do we need a sledgehammer to crack a nut? In *Proceedings of the 28th International Conference on Computational Linguistics*, pages 3333–3345, Barcelona, Spain (Online). International Committee on Computational Linguistics.

Adam Paszke, Sam Gross, Francisco Massa, Adam Lerer, James Bradbury, Gregory Chanan, Trevor Killeen, Zeming Lin, Natalia Gimelshein, Luca Antiga, and 1 others. 2019. Pytorch: An imperative style, high-performance deep learning library. *Advances in neural information processing systems*, 32.

Thang Pham, Seunghyun Yoon, Trung Bui, and Anh Nguyen. 2023. PiC: A phrase-in-context dataset for phrase understanding and semantic search. In *Proceedings of the 17th Conference of the European Chapter of the Association for Computational Linguistics*, pages 1–26, Dubrovnik, Croatia. Association for Computational Linguistics.

Qwen Team. 2025. Qwen3 technical report. *Preprint*, arXiv:2505.09388.

Alec Radford, Jeffrey Wu, Rewon Child, David Luan, Dario Amodei, Ilya Sutskever, and 1 others. 2019. Language models are unsupervised multitask learners. *OpenAI blog*, 1(8):9.

Colin Raffel, Noam Shazeer, Adam Roberts, Katherine Lee, Sharan Narang, Michael Matena, Yanqi Zhou, Wei Li, and Peter J Liu. 2020. Exploring the limits of transfer learning with a unified text-to-text transformer. *The Journal of Machine Learning Research*, 21(1):5485–5551.

Parikshit Ram, Tim Klinger, and Alexander G. Gray. 2024. What makes models compositional? a theoretical view. In *Proceedings of the Thirty-Third International Joint Conference on Artificial Intelligence*, IJCAI '24.

Carlos Ramisch. 2023. *Multiword expressions in computational linguistics*. Habilitation à diriger des recherches, Aix Marseille Université (AMU).

Carlos Ramisch, Agata Savary, Bruno Guillaume, Jakub Waszczuk, Marie Candito, Ashwini Vaidya, Verginica Barbu Mititelu, Archna Bhatia, Uxoa Iñurrieta, Voula Giouli, Tunga Güngör, Menghan Jiang, Timm Lichte, Chaya Liebeskind, Johanna Monti, Renata Ramisch, Sara Stymne, Abigail Walsh, and Hongzhi Xu. 2020. Edition 1.2 of the PARSEME shared task on semi-supervised identification of verbal multiword expressions. In *Proceedings of the Joint Workshop on Multiword Expressions and Electronic Lexicons*, pages 107–118, online. Association for Computational Linguistics.

Carlos Ramisch, Abigail Walsh, Thomas Blanchard, and Shiva Taslimipoor. 2023a. A survey of mwe identification experiments: The devil is in the details. In *Proceedings of the 19th Workshop on Multiword Expressions (MWE 2023)*, pages 106–120.

Carlos Ramisch, Abigail Walsh, Thomas Blanchard, and Shiva Taslimipoor. 2023b. A survey of MWE identification experiments: The devil is in the details. In *Proceedings of the 19th Workshop on Multiword Expressions (MWE 2023)*, pages 106–120, Dubrovnik, Croatia. Association for Computational Linguistics.

María A Barrios Rodríguez. 2003. The domain of the lexical functions fact0, causfact0 and real1. *learning*, page 64.

Ivan A Sag, Timothy Baldwin, Francis Bond, Ann Copestake, and Dan Flickinger. 2002. Multiword expressions: A pain in the neck for nlp. In *Computational Linguistics and Intelligent Text Processing: Third International Conference, CICLing 2002 Mexico City, Mexico, February 17–23, 2002 Proceedings 3*, pages 1–15. Springer.

Manfred Sailer and Stella Markantonatou. 2018. *Multiword expressions: Insights from a multi-lingual perspective*. Language Science Press.

Paul-Edouard Sarlin, Daniel DeTone, Tomasz Malisiewicz, and Andrew Rabinovich. 2020. Superglue: Learning feature matching with graph neural networks. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 4938–4947.

Agata Savary, Cherifa Ben Khelil, Carlos Ramisch, Voula Giouli, Verginica Barbu Mititelu, Najet Hadj Mohamed, Cvetana Krstev, Chaya Liebeskind, Hongzhi Xu, Sara Stymne, Tunga Güngör, Thomas Pickard, Bruno Guillaume, Eduard Bejček, Archna Bhatia, Marie Candito, Polona Gantar, Uxoa Iñurrieta, Albert Gatt, and 9 others. 2023. PARSEME corpus release 1.3. In *Proceedings of the 19th Workshop on Multiword Expressions (MWE 2023)*, pages 24–35, Dubrovnik, Croatia. Association for Computational Linguistics.

Alexander Shvets and Leo Wanner. 2022. The relation dimension in the identification and classification of lexically restricted word co-occurrences in text corpora. *Mathematics*, 10(20).

Vered Shwartz and Ido Dagan. 2019. Still a pain in the neck: Evaluating text representations on lexical composition. *Transactions of the Association for Computational Linguistics*, 7:403–419.

Joshua Tanner and Jacob Hoffman. 2023. MWE as WSD: Solving multiword expression identification with word sense disambiguation. In *Findings of the Association for Computational Linguistics: EMNLP 2023*, pages 181–193, Singapore. Association for Computational Linguistics.

Simone Tedeschi, Federico Martelli, and Roberto Navigli. 2022. ID10M: Idiom identification in 10 languages. In *Findings of the Association for Computational Linguistics: NAACL 2022*, pages 2715–2726, Seattle, United States. Association for Computational Linguistics.

Robert Vacareanu, Marco A. Valenzuela-Escárcega, Rebecca Sharp, and Mihai Surdeanu. 2020. An unsupervised method for learning representations of multi-word expressions for semantic classification. In *Proceedings of the 28th International Conference on Computational Linguistics*, pages 3346–3356, Barcelona, Spain (Online). International Committee on Computational Linguistics.

Takashi Wada, Yuji Matsumoto, Timothy Baldwin, and Jey Han Lau. 2023. Unsupervised paraphrasing of multiword expressions. In *Findings of the Association for Computational Linguistics: ACL 2023*, pages 4732–4746, Toronto, Canada. Association for Computational Linguistics.

Thomas Wolf, Lysandre Debut, Victor Sanh, Julien Chaumond, Clement Delangue, Anthony Moi, Pierric Cistac, Tim Rault, Remi Louf, Morgan Funtowicz, Joe Davison, Sam Shleifer, Patrick von Platen, Clara Ma, Yacine Jernite, Julien Plu, Canwen Xu, Teven Le Scao, Sylvain Gugger, and 3 others. 2020. Transformers: State-of-the-art natural language processing. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing: System Demonstrations*, pages 38–45, Online. Association for Computational Linguistics.

Xinnuo Xu, Rachel Lawrence, Kshitij Dubey, Atharva Pandey, Risa Ueno, Fabian Falck, Aditya V. Nori, Rahul Sharma, Amit Sharma, and Javier Gonzalez. 2025. Re-imagine: Symbolic benchmark synthesis for reasoning evaluation. *Preprint*, arXiv:2506.15455.

Fei Yu, Hongbo Zhang, Prayag Tiwari, and Benyou Wang. 2024. Natural language reasoning, a survey. *ACM Comput. Surv.*, 56(12).

Zhouliang Yu, Ruotian Peng, Keyi Ding, Yizhe Li, Zhongyuan Peng, Minghao Liu, Yifan Zhang, Yuan Zheng, Huajian Xin, Wenhao Huang, Yandong Wen, and Weiyang Liu. 2025. Formalmath: Benchmarking formal mathematical reasoning of large language models. *arXiv preprint arXiv:2505.02735*.

Ziheng Zeng and Suma Bhat. 2022. Getting BART to ride the idiomatic train: Learning to represent idiomatic expressions. *Transactions of the Association for Computational Linguistics*, 10:1120–1137.

Ziheng Zeng, Kellen Cheng, Srihari Nanniyur, Jianing Zhou, and Suma Bhat. 2023. IEKG: A commonsense knowledge graph for idiomatic expressions. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pages 14243–14264, Singapore. Association for Computational Linguistics.

Tianyi Zhang, Varsha Kishore, Felix Wu, Kilian Q Weinberger, and Yoav Artzi. 2019. Bertscore: Evaluating text generation with bert. *arXiv preprint arXiv:1904.09675*.

Jianing Zhou, Hongyu Gong, and Suma Bhat. 2021. PIE: A parallel idiomatic expression corpus for idiomatic sentence generation and paraphrasing. In *Proceedings of the 17th Workshop on Multiword Expressions (MWE 2021)*, pages 33–48, Online. Association for Computational Linguistics.

Jianing Zhou, Ziheng Zeng, Hongyu Gong, and Suma Bhat. 2022. Idiomatic expression paraphrasing without strong supervision. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 36, pages 11774–11782.

A Semantic Gloss for Lexical Functions

In recent years, there has been an increasing interest in assigning lexical functions as labels to annotated MWE in the sense of the meaning-text theory (Mel'čuk, 2023). The lexical function is a multi-valued function, which f associates a lexical unit L with a set $f(L)$ of lexical expressions.

As seen in Table 8, we constructed a collection of the representative lexical functions with their semantic glosses from the existing work. We compiled the prompts with the task descriptions.

B Additional Details of Datasets

B.1 Idiomacity Detection

In the initial dataset[2] proposed by (Harish et al., 2021), there exists three or four possible meanings (i.e., interpretations) for each instance. For instances with only three interpretations, we add the option "None of the above" to keep consistency to the four-choices form. We deduplicate according to the unique "(idiom, choice)" pair for all instances. As a result, we collate 273 examples (cf. Table 1). Figure 6 shows an example of data.

[2] https://github.com/H-TayyarMadabushi/AStitchInLanguageModels

[Context] There is also a covered pavilion. It is located next to *Silver Lining* Tire Recycling. The hours are 6:00 am to 10:00 pm, year round.

[Choices]
(A) grey lining ✘
(B) unexpected advantage ✘
(C) Proper Noun ✔
(D) Meta Usage ✘

Figure 6: A data example of idiomacity detection (IED).

B.2 Idiom Extraction

The original dataset[3] consists of instances with or without idiom $\mathcal{IE}$. Since the inference-only experiments comprise most of our work, we filter out all the examples without the $\mathcal{IE}$ existing to increase the coverage diversity of idioms; then, we deduplicate according to the unique item of the occurred $\mathcal{IE}$. The final prepared test set consists of 447 examples with a unique item of $\mathcal{IE}$ existing in each. Figure 7 shows an example of data.

[Context] In the screenplay by Lorenzo Semple Jr. , and David Rayfiel , Turner very early on stumbles upon the existence of a kind of super - C.I.A. within the C.I.A. , after which his life is *not worth a plug nickel* .

[Idiom] **"not worth a plug nickel"**

Figure 7: A data example of idiom extraction (IEE).

B.3 Idiom Interpretation

We collected 916 instances in total from the PIE (Zhou et al., 2021)[4] and (Chakrabarty et al., 2022)[5], after deduplication by occurred items of idiom $\mathcal{IE}$. Figure 8 shows an example of data.

B.4 Lexical Collocation Categorization

We collect the collocation data with the annotated labels from the expanded LEXFUNC[6] (Espinosa-Anke et al., 2021). We inherited the training and validation sets of the initial data and sampled 50 examples per semantic category from the test set

[3] https://github.com/Babelscape/ID10M
[4] https://github.com/zhjjn/MWE_PIE
[5] https://github.com/tuhinjubcse/FigurativeNarrativeBenchmark
[6] https://github.com/luisespinosaanke/lexicalcollocations

Lexical Function	Semantic Gloss	Complete Description
Magn (Mel'čuk, 1998)	Intense, strong degree, an intensifier of semantic relation for base lexeme.	Intensify the base lexeme to a high level, strengthening its semantic relation with the associated concept via the collocate lexeme.
AntiMagn (Mel'čuk, 1998)	Slight and weak degree, a de-intensifier	Weaken meaning intensity, diminishing the semantic relationship between the base lexeme and its associated concept.
Ver (Gelbukh et al., 2012)	Lat. verus, real, genuine	"As it should be", "Meet the intended requirements of *K*".
AntiVer (Mel'čuk, 1998)	Non-genuine	Characterize something as non-genuine, not authentic, not in its intended or proper state, and not meeting the required standards or expectations.
Bon (Espinosa-Anke et al., 2021)	Positive	Something is good or in a positive situation.
AntiBon (Espinosa-Anke et al., 2021)	Negative	Something is bad or in a negative situation.
IncepPredPlus (Fontenelle, 1997)	Start to increase.	Denote initiating a process or action that leads to an increase or enhancement of something.
FinFunc0 (Kolesnikova, 2020)	End.existence	The value means "the *K* of FinFunc0 ceases to be experienced".
Fact0 (Mel'čuk, 1998)	Lat. factum, fact. To fulfil the requirement of *K*, and the argument of this function fulfills its own requirement.	Fulfill the base requirement, do something with the base, and do what you are supposed to do with the base.
CausFunc0 (Gelbukh et al., 2012)	The agent does something so that the event denoted by the noun occurs	Do something so that *K* begins occurring.
Caus1Func0 (Espinosa-Anke et al., 2021)	Cause the existence. 1st argument.	Bring about something's presence or creation, with the first argument indicating the responsible agent or entity.
CausFact0 (Rodríguez, 2003)	To cause something to function according to its destination.	Denote causing something to function according to its intended purpose or destination.
CausPredMinus (Fontenelle, 1997)	Cause to decrease.	Describe the act of causing a decrease or reduction in something.
CausFunc1 (Gelbukh et al., 2012)	The non-agentive participant does something such that the event denoted by the noun occurs.	A person/object, different from the agent of *K*, does something so that *K* occurs and has an effect on the agent of *K*.
LiquFunc0 (Espinosa-Anke et al., 2021)	Cause termination of the existence	Cause termination of the existence.
Son (Kolesnikova, 2020)	Lat. *sonare*: sound.	The *K* is usually a noun, and the value means "emit a characteristic sound".
Oper1 (Kolesnikova, 2020)	Lat. *operari*: perform, do, act something. The subject is as the 1st argument.	Represent a light verb linking the event's first participant (subject) with the event's name (direct object).
Oper2 (Espinosa-Anke et al., 2021)	Lat. *operari*: perform, do, act something. The subject is as the 2nd argument.	Represent a light verb linking the event's first participant (subject) with the event's name (indirect object).
IncepOper1 (Gelbukh et al., 2012)	Incep is from Lat. *incipere*: begin. Begin to do, perform, experience, carry out *K*.	Signify the start of an action or event, linking the event's subject with its name using a light verb.
FinOper1 (Kolesnikova, 2020)	Fin is from Lat. *finire*: cease.	Terminate doing something.
Real1 (Rodríguez, 2003)	Fulfill a requirement imposed by the noun or performing an action typical for the noun.	To fulfill the requirement of *K*, to act according to *K*.
Real2 (Kolesnikova, 2020)	Acting as expected. Something be realized as expected	*K* that is normally expected of the second participant
AntiReal2 (Kolesnikova, 2020)	Not acting as expected. Something not be realized as expected.	The *V* is the negation of an internal element of the argument of this function.

Table 8: All lexical functions with their semantic gloss in this paper. The column "semantic gloss" provides the definition for each LF, and we use a sentence to describe the complete meaning of LF in column "Complete Description". *K* denotes the keyword/base word of a LF, and *V* denotes the value/collocate word of a LF.

[Context] The remission at this stage of having cancer was truly the *turning point* of her life .

[Idiom] "turning point"

[Interpretation] **"the time of significant change (mostly positive) in situation"**

Figure 8: A data example of idiom interpretation (IEI).

in classification concerning the computation efficiency. Figure 9 shows an example of data.

[Context] In genoa, the *violent storm* knocked down power lines, blacking out the homes of 5,000 residents.

[Category] **Magn** (strong semantic).

Figure 9: A data example is the lexical collocation categorization (LCC) by semantic relations. Note that the taxonomy included in the prompt is omitted here.

B.5 Lexical Collocation Extraction

The initial dataset is collected from (Fisas et al., 2020)[7]. We select the English part of the data and perform deduplication to filter out overlap collocations. We downsample 50 instances randomly for each semantic category to form our test set and reuse the training and validation sets of the original data. Figure 10 shows an example of data. We conduct $\mathcal{LC}$ extraction but not identification task, and not query models to distinguish the base and the collocate to simplify the task in this work.

[Context] He still gets up the moment the *alarm clock rings* .

[Semantic relation] Strong or intense degree in the lexical semantic relation.

[Collocation]
"alarm clock rings"

Figure 10: A data example of collocation extraction (LCE).

[7] https://github.com/TalnUPF/CollFrEn

B.6 Lexical Collocation Interpretation

The data[8] we used is proposed in (Espinosa-Anke et al., 2021). We perform random sampling from the original data and get the 400 examples (50 per class) as our test set. We manually annotated and revised the test examples, and finally get the Cohen's kappa coefficient $\kappa = 0.718$, to confirm the quality. An example of data is shown in Figure 11.

[Context] Through robert bennett, his lawyer, the president continued friday to call mrs. jones' *baseless accusation*.

[Collocation] "baseless accusation"

[Interpretation] **"Groundless claim made without substantiation"**

Figure 11: A data example of collocation interpretation (LCI).

B.7 Noun Compound Compositionality

The annotated noun compound data is collected from the NCTTI[9] (Garcia et al., 2021). After data processing, we filtered out the compound without reference context, collated 237 examples, and split them into training, validation, and test sets. Figure 12 shows an example of data.

[Context] *Fair play* incorporates the concepts of friendship, respect for others and always playing in the right spirit.

[Noun compound] "Fair play"

[Choices]
(A) Compositional ✘
(B) Partly compositional ✔
(C) None of the above ✘
(D) Non-compositional ✘

Figure 12: A data example of noun compound compositionality (NCC).

B.8 Noun Compound Extraction

As our beginning, we sampled the test set from the PRONCI[10] (Kolluru et al., 2022). We used the

[8] https://github.com/luisespinosaanke/lexicalcollocations

[9] https://github.com/marcospln/nctti

[10] https://github.com/dair-iitd/pronci

training and validation sets to leverage the compositional part of noun compounds in the original dataset. We randomly sampled from the test set to form the new test set with 720 examples. We demonstrate a data example in Figure 13.

[Context] The rhombus shape of the patches arose by adaptation to the *Paris fashion* of the 17th century by Biancolelli.

[Noun compound] **"Paris fashion"**

Figure 13: A data example of noun compound extraction (NCE).

B.9 Noun Compound Interpretation

We leverage the initial training, validation, and test data splits from (Coil and Shwartz, 2023)[11]. To provide a context for each noun compound, we use ChatGPT to generate a reference sentence. To verify the quality of synthetic data, we performed a manual inspection, which resulted in $acc > 98\%$. A data example is shown in the figure 14.

[Context] She used a straightedge to draw a *ruler line* across the paper, ensuring her graph was perfectly aligned.

[Noun compound] "ruler line"

[Interpretation] **"line drawn with a ruler"**

Figure 14: A data example of noun compound interpretation (NCI).

B.10 VMwE Extraction

We used the English corpus of PARSEME v1.3[12] (Savary et al., 2023), the existing largest annotated corpora of VMwE. The initial data is used to conduct extraction instead of identification tasks. Figure 15 shows an example of the data.

[Context] Harry tore back across the room as the landing light *clicked on*.

[VMwE] **"clicked on"**

Figure 15: A data example of VMwE Extraction.

[11] https://github.com/jordancoil/noun-compound-interpretation

[12] https://gitlab.com/parseme/parseme_corpus_en

C Example Prompt

We manually create a unified prompt template for all tasks that can be adapted to each task with specific filling arguments. The prompt format is shown in the Figure 16. The detailed prompt for each task can be accessed in our code base[13].

Unified Prompt Template

Assume that you are a linguist who researches {{semantic phrases}}.

You will be given a sentence that contains only an item of {{semantic phrase}}.

Your task is to ...

Please make sure you read and understand these instructions carefully.

Few-shot Examples:

Phrase: {{*an example of the phrase*}}

Context: {{*a context of the example*}}
Output: {{*an output of the example*}}
...

Phrase: {{*phrase*}}

Context: {{*context*}}

Output:

Figure 16: Unified prompt template used in the work.

D Oracle Prompt

Oracle Prompt Template

Assume that you are a linguist who conducts research on {{verbal multiword expressions (VMwEs)}}.

You will be given a context that includes only one {{verbal multiword expression}}.

Your task is to ...

Please make sure you read and understand these instructions carefully.

Few-shot Examples:
Context: {{*a context of the example*}}
Output: {{*an output of the example*}}
...

VMwE Definition: {{*Definition of VMwE*}}

VMwE Definition Example:
Definition of VMwE: {{*"Verb-particle construction (VPC) is sometimes called phrasal or phrasal-prepositional verb. The meaning of the VPC is fully or partly non-compositional."*}}

Context: {{*context*}}

Output:

[13] https://github.com/lexbench/LexBench/tree/main/lexbench/prompts

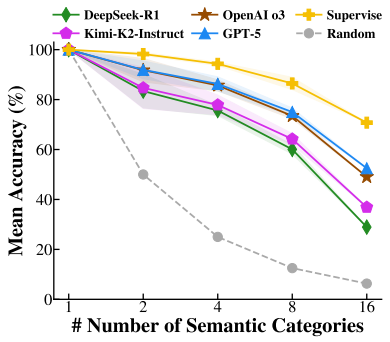

Figure (18) Each model is run with zero-shot prompting in the semantic relation classification with category scaling. Mean accuracy scores (%) of different models are average over runs in three sampled sets. For comparative reasons, we also plotted the level of random baseline.

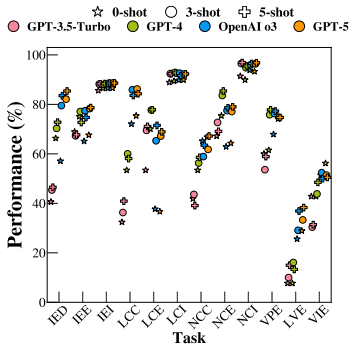

Figure (19) Model performance (GPT-3.5-Turbo, GPT-4, OpenAI o3, GPT-5) across all twelve tasks. Note that the y-axis denotes task-specific metrics, and thus absolute values should not be compared across different tasks.

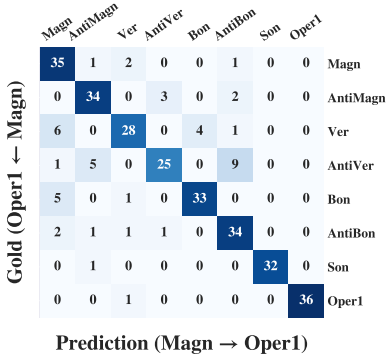

Figure (20) Confusion matrix for the best-performing model with ICL (GPT-5 in 5-shot setting) in categorizing eight semantic relations described by lexical functions (cf. Table 2). The x-axis denotes the prediction results, and the y-axis represents the gold standards.

Model	# Params	Arch.	Creator	Public	Post Training
BERT base†	110M	Enc.	Google	✔	SFT
BERT large†	340M	Enc.	Google	✔	SFT
T5 base†	220M	Enc.+Dec.	Google	✔	SFT
T5 large†	770M	Enc.+Dec.	Google	✔	SFT
Qwen3-235B‡	235B	Dec.(MoE)	Qwen Team	✔	SIFT
DeepSeek-R1‡	685B	Dec.(MoE)	DeepSeek-AI	✔	SIFT + RLHF
Kimi-K2-Instruct‡	1T	Dec.(MoE)	Kimi Team	✔	SIFT
Gemma-3-27B-it‡	27B	Dec.	Gemma Team	✔	SIFT
Gemini-1.0-pro‡	*	*	Google	✘	SIFT + RLHF
Gemini-2.5-pro‡	*	*	Google	✘	SIFT + RLHF
Claude-Instant-1‡	*	*	Anthropic	✘	SIFT + RLHF
Claude-3-Opus‡	*	*	Anthropic	✘	SIFT + RLHF
Claude-Sonnet-4.5‡	*	*	Anthropic	✘	SIFT + RLHF
GPT-3.5-Turbo‡	*	*	OpenAI	✘	SIFT + RLHF
GPT-4‡	*	*	OpenAI	✘	SIFT + RLHF
OpenAI o3‡	*	*	OpenAI	✘	SIFT + RLHF
GPT-5‡	*	*	OpenAI	✘	SIFT + RLHF

Table 9: A list of LMs tested in this paper: "Public" indicates whether the model weights are open. In detail, Light Pink text delineates the supervised fine-tuned models. Light Green and Light Blue parts present open-source models and proprietary models, respectively. "Post Training" indicates whether the model is trained further in some ways after pre-training. †We perform trivial full-set fine-tuning for the models. ‡We use the official API for the model inference.

Figure 20: Oracle prompt template used in the work.

E Additional Experiment Details

For models accessed via API endpoints, the evaluation probes both zero-shot and few-shot (three- and five-shot) performance. Throughout all experiments, we set the sampling temperature to $\tau = 0$ and employ top-p decoding (Holtzman et al., 2019) with $p = 1.0$. Inference is accelerated and deployed using vLLM (Kwon et al., 2023).

For non-API-based models, we apply the following configuration. For the sequence classification tasks such as LCC, we employ `bert-base/large-uncased` as our tuning initiation. Similarly, we construct primary baselines for extraction tasks that leverage the B-I-O scheme to conduct sequence labeling. The training is run with an NVIDIA A100-40GB on Google Colab (Bisong, 2019). For interpretation tasks, we use `t5-base/large` model to conduct vanilla fine-tuning. Additionally, We train all models for a specific number of epochs shown in Table 11 and perform early stopping over the validation set. Model checkpoints used in our experiment are implemented by PyTorch (Paszke et al., 2019), and Hugging Face Transformers (Wolf et al., 2020). The input format of the prompt and the few-shot demonstration settings we used during the experiment are shown in Figure 16. Since each model has different generation styles, we conduct a pre-run before each test. Then, we develop ad hoc heuristics based on the response generated by models to parse predictions accurately. The perplexity computing in the interpretation tasks is to feed the phrase and its interpretation into the template *"The meaning of phrase {{phrase}} in context is {{interpretation}}"*, and then we compute the token-level perplexity by GPT-2-XL (Radford et al., 2019).

F Annotation Guideline

We established the following criteria for compiling the dataset of collocation interpretation (§3.2).

1. **Objective:** Interpret each lexical collocation in five distinct narratives for comprehensive understanding according to the given context.

2. **Dataset Overview:** Contains context and col-

locations paired with base and collocate.

3. **Annotation Format:** Include collocation, five narratives (N1-N5), and rationale.

4. **Consistency and Accuracy:** Maintain consistent and accurate interpretations across the five narratives in the same semantic meaning.

VMwE	BERT-base			BERT-large			# Support
	P	R	F1	P	R	F1	
IAV	$60.7_{5.6}$	$38.0_{4.3}$	$46.5_{3.3}$	$46.5_{3.6}$	$38.9_{5.6}$	$42.3_{4.8}$	36.0
LVC.cause	$46.4_{12.2}$	$18.4_{4.0}$	$26.2_{5.6}$	$26.4_{12.4}$	$20.7_{9.1}$	$23.2_{10.5}$	29.0
LVC.full	$52.1_{4.4}$	$61.1_{2.0}$	$56.1_{2.2}$	$55.2_{2.5}$	$56.8_{8.4}$	$55.9_{5.4}$	172.0
MVC	$95.9_{4.0}$	$80.5_{2.0}$	$87.5_{2.8}$	$100.0_{0.0}$	$80.5_{2.0}$	$89.2_{1.2}$	29.0
VID	$52.4_{5.3}$	$36.1_{0.9}$	$42.7_{1.8}$	$63.8_{5.0}$	$36.1_{1.9}$	$46.1_{2.0}$	108.0
VPC.full	$64.3_{3.3}$	$78.4_{1.6}$	$70.6_{1.5}$	$64.4_{0.1}$	$79.4_{0.5}$	$71.1_{0.2}$	194.0
VPC.semi	$55.9_{38.7}$	$8.9_{6.9}$	$12.9_{7.3}$	$38.8_{6.4}$	$35.6_{3.9}$	$37.1_{5.0}$	30.0
Micro Avg.	$63.2_{1.9}$	$61.2_{0.6}$	$62.3_{1.3}$	$64.2_{0.5}$	$62.4_{2.7}$	$63.3_{1.6}$	85.4

Table 10: We report the full results of VMwE extraction reproduced on MTLB-STRUCT. The performance of all categories are defined in the corpora PARSEME 1.3. The corresponding standard deviation is calculated by the results of three runnings with the selected seeds $\{21, 42, 84\}$.

Computing Infrastructure 1 × A100 40GB GPU (Google Colab)			
Hyperparameter	**Assignment**	**Hyperparameter**	**Assignment**
architecture	BERT-{base, large}	architecture	T5-{base, large}
tokens per sample	150	tokens per sample	128
batch size	$4,800$	batch size	$2,048$
number of workers	8	number of workers	4
learning rate	$3e^{-5}$	learning rate	$5e^{-5}$
number of epochs	10	number of epochs	5
save interval (epoch)	1	save interval (epoch)	1
validation interval (epoch)	1	validation interval (epoch)	1
ratio of warmup steps	3%	ratio of warmup steps	3%
learning rate scheduler	Polynomial decay	learning rate scheduler	Cosine decay
learning rate optimizer	Adam	learning rate optimizer	Adam
Adam beta weights	$(0.9, 0.99)$	Adam beta weights	$(0.9, 0.99)$
Adam epsilon	$1e^{-6}$	Adam epsilon	$1e^{-6}$
weight decay	0	weight decay	0
random seed	21, 42, 84	random seeds	21, 42, 84

Table 11: Hyperparameters for finetuning BERT-Taggers and T5 Generators.

Model	Idiom			Collocation			Noun Compound			VMwE		
	IED	IEE	IEI	LCC	LCE	LCI	NCC	NCE	NCI	VPE	LVE	VIE
Metric (%)	Acc	Acc_s	B-S	Acc	Acc_s	B-S	Acc	Acc_s	B-S	Acc_s	Acc_s	Acc_s
Human	71.0	**87.0**	87.6	47.0	50.0	86.8	**71.0**	73.0	80.3	**85.0**	**55.0**	**78.0**
Supervised Methods												
$BERT_B$: *fine-tuned*	85.0	66.8	-	78.8	63.1	-	53.6	68.5	-	68.7	**52.2**	36.1
$BERT_L$: *fine-tuned*	85.1	67.2	-	82.6	63.8	-	51.5	69.1	-	74.1	41.7	34.2
$T5_B$: *fine-tuned*	-	-	86.8	-	-	87.2	-	-	**89.7**	-	-	-
$T5_L$: *fine-tuned*	-	-	87.1	-	-	87.7	-	-	89.8	-	-	-
Prompt-based Methods												
Qwen3-235B: *zero-shot*	64.1	53.6	86.7	58.0	**25.9**	90.3	52.7	45.3	93.5	56.8	19.4	39.1
DeepSeek-R1: *zero-shot*	71.1	69.4	**85.1**	66.6	31.5	90.2	60.2	51.3	91.3	76.8	26.7	50.5
↪ + *three-shot*	79.1	70.6	88.1	76.4	55.6	91.6	62.7	66.3	96.3	74.7	26.7	**59.1**
↪ + *five-shot*	84.3	72.3	88.0	76.1	64.3	91.8	60.6	70.7	96.6	81.6	35.8	57.1
Kimi-K2-Instruct: *zero-shot*	68.5	63.1	86.7	68.5	34.4	90.3	60.6	45.4	95.6	55.8	28.9	46.7
↪ + *three-shot*	77.7	68.9	88.4	79.0	67.9	92.8	59.3	64.4	96.7	79.5	39.4	43.8
↪ + *five-shot*	81.7	69.6	88.2	79.7	69.2	92.3	64.7	63.6	97.2	81.1	43.3	46.7
Gemma-3-27B-it: *zero-shot*	55.0	57.3	86.4	58.0	38.4	89.5	58.3	39.9	92.1	66.8	19.4	38.1
↪ + *three-shot*	69.6	62.0	88.1	70.1	63.7	91.1	56.7	57.2	95.3	74.1	28.3	45.7
↪ + *five-shot*	72.1	61.6	87.9	70.8	68.2	90.7	56.2	59.2	95.9	70.5	35.0	52.4
Gemini-1.0-pro: *zero-shot*	56.0	77.8	86.9	48.5	51.8	89.5	**38.5**	59.0	91.8	43.8	**6.7**	43.8
Gemini-2.5-pro: *zero-shot*	55.0	65.6	87.4	71.5	52.1	89.4	65.6	61.2	93.7	**42.6**	27.4	42.9
Claude-Instant-1: *zero-shot*	51.2	72.2	85.7	40.5	42.6	89.7	43.2	50.9	91.9	59.2	11.6	39.0
↪ + *three-shot*	47.9	60.8	86.5	49.8	54.7	**87.0**	47.8	59.1	94.1	48.9	18.8	35.5
↪ + *five-shot*	52.0	**47.4**	87.0	50.1	57.7	87.1	44.9	61.8	94.5	53.1	15.0	38.4
Claude-3-Opus: *zero-shot*	66.3	62.8	87.1	61.3	34.7	88.5	50.4	36.3	91.7	67.3	28.3	42.8
↪ + *three-shot*	75.8	64.8	88.1	69.5	56.7	92.8	56.7	33.6	93.1	74.7	37.2	47.6
↪ + *five-shot*	72.8	67.1	88.2	69.8	60.0	92.8	63.9	**30.9**	96.0	75.7	35.5	43.2
Claude-Sonnet-4.5: *zero-shot*	72.5	68.5	87.2	67.5	40.1	88.9	51.0	45.1	94.4	69.8	16.1	41.9
↪ + *three-shot*	77.7	72.0	88.4	77.1	70.5	91.8	61.4	59.3	96.8	76.8	30.6	42.9
↪ + *five-shot*	78.0	72.0	88.5	76.1	72.7	90.9	**70.1**	62.1	**97.6**	**82.0**	37.2	47.6
GPT-3.5-Turbo: *zero-shot*	**40.6**	68.9	85.6	**32.4**	53.4	88.9	41.9	67.2	91.4	60.0	7.7	42.8
↪ + *three-shot*	45.4	67.3	88.2	36.3	69.5	92.4	43.6	72.7	96.5	53.6	10.0	**30.4**
↪ + *five-shot*	46.5	67.7	88.3	40.9	71.1	92.4	39.1	69.1	96.9	58.9	15.0	31.4
GPT-4: *zero-shot*	66.3	75.1	86.5	53.4	70.1	89.4	53.4	75.4	89.9	61.5	7.7	42.8
↪ + *three-shot*	70.3	77.1	88.1	60.0	77.7	**92.9**	56.3	83.6	94.8	75.8	16.1	43.8
↪ + *five-shot*	72.8	72.7	88.4	58.1	**77.8**	92.7	58.6	**85.4**	95.5	77.8	13.3	48.5
OpenAI o3 : *zero-shot*	57.1	65.1	86.5	72.1	37.7	89.8	65.2	62.9	93.8	67.9	25.6	51.4
↪ + *three-shot*	79.5	77.4	88.5	85.9	65.3	92.6	58.9	77.5	96.0	76.3	29.1	52.4
↪ + *five-shot*	83.5	74.7	88.6	83.6	71.5	91.6	63.5	78.6	96.5	77.3	36.9	50.0
GPT-5: *zero-shot*	82.8	67.6	86.6	75.4	36.7	89.9	66.8	64.3	93.3	74.2	28.9	56.2
↪ + *three-shot*	82.1	78.3	**88.7**	**86.2**	67.2	92.3	61.8	77.1	96.4	74.7	33.3	51.4
↪ + *five-shot*	**85.4**	**78.7**	88.6	84.3	68.9	92.3	67.2	79.0	96.8	74.7	38.3	50.5

Table 12: Complete Experimental Results in SemanticQA. "-" denotes the model that is unavailable or inappropriate for the task. **Digits** highlight cases in which human scores are higher than those of all evaluated models, serving as a coarse reference. Light Pink text delineates the baselines with supervised fine-tuning. Light Green and Light Blue parts present open-source models and proprietary models.

System	Acc@1	Acc@2	Acc@4	Acc@8	Acc@16
Baselines					
Random	100.0	50.0	25.0	12.5	6.3
Majority	100.0	50.0	25.0	12.5	6.3
Small language models					
BERT_{B}	$100.0_{0.0}$	$98.9_{1.9}$	$89.4_{4.9}$	$79.9_{6.4}$	$69.9_{0.0}$
BERT_{L}	$100.0_{0.0}$	$\mathbf{98.9_{1.0}}$	$\mathbf{95.8_{1.4}}$	$\mathbf{83.5_{5.2}}$	$\mathbf{71.8_{0.0}}$
Large language models					
DeepSeek-R1					
$\hookrightarrow$ + *0-shot*	$100.0_{0.0}$	81.7 $_{11.9}$	68.9 $_{4.2}$	49.3 $_{4.0}$	35.4 $_{0.0}$
$\hookrightarrow$ + *3-shot*	$100.0_{0.0}$	83.8 $_{7.5}$	73.7 $_{2.4}$	52.0 $_{6.2}$	45.9 $_{0.0}$
$\hookrightarrow$ + *5-shot*	$100.0_{0.0}$	80.6 $_{11.8}$	68.4 $_{7.8}$	51.7 $_{7.7}$	47.3 $_{0.0}$
Kimi-K2-Instruct					
$\hookrightarrow$ + *0-shot*	$100.0_{0.0}$	85.6 $_{14.6}$	71.1 $_{8.2}$	50.4 $_{4.6}$	44.6 $_{0.0}$
$\hookrightarrow$ + *3-shot*	$100.0_{0.0}$	92.8 $_{6.7}$	78.3 $_{3.8}$	61.5 $_{7.9}$	49.6 $_{0.0}$
$\hookrightarrow$ + *5-shot*	$100.0_{0.0}$	87.2 $_{4.2}$	77.8 $_{6.9}$	63.8 $_{5.2}$	51.7 $_{0.0}$
OpenAI o3					
$\hookrightarrow$ + *0-shot*	$100.0_{0.0}$	93.3 $_{8.3}$	82.8 $_{4.6}$	65.4 $_{2.5}$	53.3 $_{0.0}$
$\hookrightarrow$ + *3-shot*	$100.0_{0.0}$	91.7 $_{5.9}$	85.3 $_{3.1}$	72.6 $_{3.4}$	62.9 $_{0.0}$
$\hookrightarrow$ + *5-shot*	$100.0_{0.0}$	91.1 $_{5.5}$	88.6 $_{4.5}$	74.0 $_{1.6}$	64.6 $_{0.0}$
GPT-5					
$\hookrightarrow$ + *0-shot*	$100.0_{0.0}$	92.2 $_{6.3}$	84.2 $_{5.9}$	67.7 $_{3.4}$	56.3 $_{0.0}$
$\hookrightarrow$ + *3-shot*	$100.0_{0.0}$	92.8 $_{9.1}$	88.6 $_{3.8}$	<u>74.6 $_{2.8}$</u>	<u>65.8 $_{0.0}$</u>
$\hookrightarrow$ + *5-shot*	$100.0_{0.0}$	<u>94.4 $_{6.7}$</u>	<u>89.2 $_{4.1}$</u>	73.5 $_{3.6}$	65.2 $_{0.0}$

Table 13: Our best experimental results (avg_{std}). The mean accuracy scores with their standard deviation are computed by averaging the results of three independent runs with different random seeds. Results of baselines are also provided including random choice as well as the majority of class instances over each sub categorization tasks. The **Bold** and <u>underlined</u> texts denote the best and second-best performance in the specific category, respectively.