Evaluating LLMs' Capability to Identify Lexical Semantic Equivalence: Probing with the Word-in-Context Task

Yoshihiko Hayashi
Perceptual Computing Laboratory, Waseda University
Wasedamachi 27, Shinjuku, Tokyo, 162-0042 Japan
yoshihiko.hayashi@gmail.com

Abstract

This study proposes a method to evaluate the capability of large language models (LLMs) in identifying lexical semantic equivalence. The Word-in-Context (WiC) task, a benchmark designed to determine whether the meanings of a target word remain identical across different contexts, is employed as a probing task. Experiments are conducted with several LLMs, including proprietary GPT models and open-source models, using zero-shot prompting with adjectives that represent varying levels of semantic equivalence (e.g., "the same") or inequivalence (e.g., "different"). The fundamental capability to identify lexical semantic equivalence in context is measured using standard accuracy metrics. Consistency across different levels of semantic equivalence is assessed via rank correlation with the expected canonical ranking of precision and recall, reflecting anticipated trends in performance across prompts. The proposed method demonstrates its effectiveness, highlighting the superior capability of GPT-4o, as it consistently outperforms other explored LLMs. Analysis of the WiC dataset, the discriminative properties of adjectives (i.e., their ability to differentiate between levels of semantic equivalence), and linguistic patterns in erroneous cases offer insights into the LLM's capability and sensitivity. These findings could inform improvements in WiC task performance, although performance enhancement is not the primary focus of this study.

1 Introduction

Polysemy, the phenomenon where a single word has multiple meanings, has been a significant concern across various academic disciplines (Ravin and Leacock, 2000). In NLP, this issue is particularly relevant to Word Sense Disambiguation (WSD) (Agirre and Edmonds, 2006; Navigli, 2009). Despite advancements in the field, particularly with the development of Transformer-based text encoders, accurately identifying the intended meaning of a word in context and mapping it to a predefined sense from a fixed sense inventory remains challenging (Bevilacqua et al., 2021). A major difficulty arises from the lack of clearly or rigorously defined sense boundaries (Ide and Wilks, 2006; Panchenko et al., 2017).

The Word-in-Context (WiC) task (Pilehvar and Camacho-Collados, 2019), which involves determining whether a target word's meanings are identical across different contexts, offers an alternative approach that bypasses the need for strict sense division. Despite the contextual clues provided in the WiC dataset[1], the task remains nuanced, with human accuracy reported at only 0.80. While recent large language models (LLMs) have achieved significantly higher accuracy, reaching 0.843 with advanced prompting techniques (Wang and Zhao, 2024), their specific semantic capabilities and characteristics from a lexical semantics perspective still remain elusive.

This study aims to develop a solid method for evaluating an LLM's capability to identify lexical semantic equivalence using the WiC task, prioritizing methodological development overachieving state-of-the-art task performance. Figure 1 illustrates the overall framework of our approach, highlighting the integration of zero-shot prompts and evaluation metrics. The key idea is to guide LLM predictions using zero-shot prompts featuring adjectives that represent different levels of semantic equivalence (e.g., "the same," "similar"). We evaluate the LLM's capability from two perspectives: fundamental capability, which focuses on baseline performance, and consistency with the level of equivalence, which examines the model's sensitivity to varying degrees of semantic similarity. For the former, we use the standard accuracy metric in the WiC dataset. For the latter, we assess precision

[1] https://pilehvar.github.io/wic/

Proceedings of the 31st International Conference on Computational Linguistics, pages 6985–6998
January 19–24, 2025.

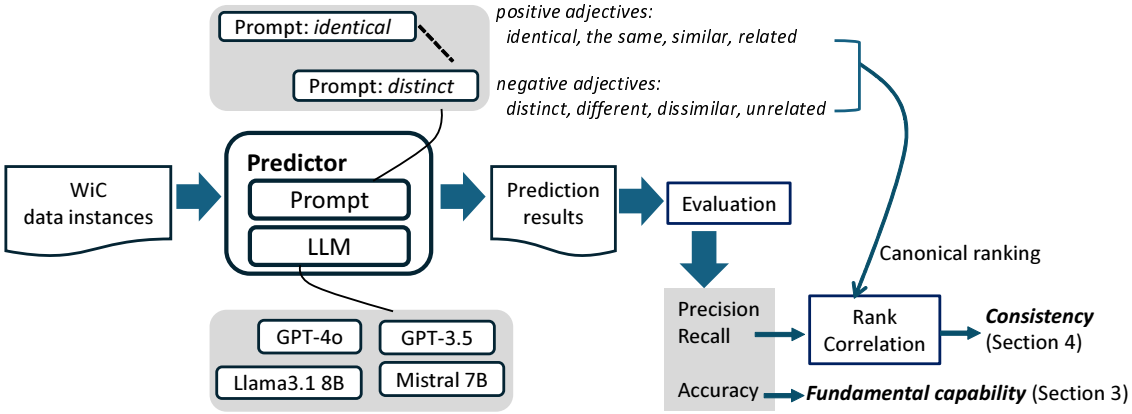

Figure 1: Overall framework of this study.

and recall figures obtained through these prompts, as these metrics effectively capture the trade-offs imposed by different levels of equivalence. For example, the adjective "the same" imposes a stricter equivalence criterion than "similar," resulting in higher precision but lower recall.

In summary, our evaluation framework is based on three key assumptions:

- Semantic equivalence exists on a spectrum, and adjectives such as “identical” or “related” represent various levels of equivalence. LLMs are expected to differentiate between these levels.
- The zero-shot prompts that feature these adjectives provide useful data to assess the semantic capabilities of an LLM.
- Although some individual cases in the WiC dataset may pose challenges, it remains a suitable and consistent tool for evaluating an LLM’s capability to identify semantic equivalence.

The rest of this paper is organized as follows: Section 2 presents our methodology, along with an introduction to key concepts. Sections 3 and 4 present the evaluation results and insights, demonstrating the effectiveness of our approach. Our results suggest that GPT-4o outperforms other LLMs tested. Section 5 discusses these findings from various perspectives, including the WiC dataset, the discriminative properties of adjectives, and linguistic patterns in erroneous cases. Finally, Section 6 explores potential improvements through ensemble methods, although performance enhancement is not the primary focus of this study. The relevant codes and data are available at this URL[2].

2 Methodology

This section introduces the WiC task, defines the predictors used in the experiments, and outlines the experimental settings. The evaluation method is proposed alongside a description of the experimental results in the following sections.

2.1 WiC Task and the Dataset

In the WiC task (Pilehvar and Camacho-Collados, 2019), the system is required to determine whether a target word (w), either a verb or noun, exhibits semantic equivalence across two contextual sentences (c_1 and c_2). Each instance in the WiC dataset is annotated to indicate whether the meanings of the target word in the two contexts are identical (labeled "T", referred to as **T-instance**) or different (labeled "F", **F-instance**). Since the dataset maintains a balanced ratio of T-instances and F-instances, task performance is basically evaluated using the accuracy metric.

Figure 2 exemplifies a T-instance and an F-instance from the WiC dataset. In the first example, the target noun "plane" in both c_1 and c_2 refers to the process of operating machinery, leading to a positive label (T). In contrast, the second example features the target verb "excite," which refers to a physiological activity in c_1 and a mental reaction in c_2, justifying the negative label (F). These instances demonstrate how the WiC task distinguishes between identical and distinct senses of target words across different contexts.

[2] https://github.com/yoshihikohayashi/wic_llm_coling2025

```
w/POS: operation/N
c1: The plane's operation in high
winds.
c2: The power of its engine
determines its operation.
label: T

w/POS: excite/V
c1: Excite the neurons.
c2: The fireworks which opened the
festivities excited anyone present.
label: F
```

Figure 2: Examples of T- and F-instances from the WiC dataset.

```
Your task is to identify if the
meanings of the target word "{word}"
in the following c1 and c2 sentences
correspond to "{adj}" meanings or
not.
That is, it is the Word-in-Context
task.

Please simply answer T,
if the meanings correspond to
identical meanings.
Otherwise, simply answer F.
[Question]
Target word: {word}
c1: {c1}
c2: {c2}
Answer:
```

Figure 3: Template for zero-shot prompting with a positive adjective.

2.2 Predictor

The WiC task is a binary classification problem, where the classifier (referred to as a "Predictor" in Figure 1) is defined by a combination of the LLM used and the adjective instantiated in the prompt. We utilize four LLMs: GPT-3.5 (Brown et al., 2020), GPT-4o (OpenAI, 2023), Llama3.1 8B (LlamaTeam-AI@Meta, 2024), and Mistral 7B (Jiang et al., 2023) in the experiments: The first two models are proprietary, while the last two are open-source.

2.3 Experimental Settings

Under the two experimental settings described below, we collect standard evaluation metrics for classification, such as accuracy, precision, recall, and F1 score. Of these, accuracy is used to assess the fundamental capability of a predictor, while precision and recall are employed to calculate the consistency level, as detailed in Section 4.

(1) Regular WiC Task Setting: In this setting, the task adheres to the original formulation: the system must determine whether the meanings of a target word are identical in two given contextual sentences. We use zero-shot prompts with adjectives, referred to as **positive adjectives**, which denote different levels of semantic equivalence. This approach allows us to assess LLM's sensitivity to these varying levels of equivalence. Figure 3 illustrates the zero-shot prompt template used for this task setting[3].

[3]We used the exact same template for GPT-3.5 and GPT-4o. The template was slightly modified for Llama3.1 8B and Mistral 7B to control the models' output format.

Positive Adjectives: In this study, we consider four positive adjectives: **identical**, **same**, **similar**, and **related**. These adjectives are ordered according to the levels of semantic equivalence they represent, with "identical" indicating the highest level and "related" indicating the lowest.

(2) Reversed WiC Task Setting: To further evaluate the LLM's capability to identify lexical semantic equivalence, we employ a *reversed* task setting. This approach utilizes a zero-shot prompt template parallel to the one shown in Figure 3 but incorporates a **negative adjective** to represent a degree of semantic inequivalence. Consequently, the final label is flipped to align with the regular task evaluation setting. That is, if a model predicts "Yes" for a negative prompt (e.g., "the meanings of the target word in the contextual sentences are distinct"), the predicted label is assigned as "F."

Negative Adjectives: We focus on four negative adjectives that indicate semantic inequivalence: **distinct**, **different**, **dissimilar**, and **unrelated**. These adjectives are arranged according to the levels of semantic inequivalence they represent, with "distinct" indicating the highest level and "unrelated" indicating the lowest.

3 Fundamental Capability for Identifying Lexical Semantic Equivalence

We use accuracy from the WiC task as the primary metric to evaluate an LLM's fundamental capability to identify lexical semantic equivalence in context.

Adjective	GPT-3.5			GPT-4o			Llama3.1 8B			Mistral 7B		
	train	val	test	train	val	test	train	val	test	train	val	test
identical	0.61	0.605	0.616	0.75	0.749	0.755	**0.633**	0.616	**0.633**	0.644	0.632	0.648
the same	0.657	0.636	**0.661**	0.761	0.765	0.769	0.627	**0.63**	0.611	**0.681**	**0.665**	**0.661**
similar	0.669	**0.644**	0.651	0.778	**0.768**	**0.776**	0.616	0.574	0.586	**0.681**	0.663	0.653
related	0.62	0.594	0.592	**0.786**	0.762	0.759	0.538	0.519	0.527	0.566	0.553	0.559
pos. avg.	0.639	0.62	0.63	0.769	0.761	0.765	0.604	0.585	0.589	0.643	0.628	0.63
distinct	0.504	0.505	0.499	0.772	0.76	0.767	0.505	0.508	0.514	0.501	0.5	0.5
different	0.592	0.594	0.603	0.779	0.77	0.765	0.51	0.514	0.519	0.509	0.511	0.515
dissimilar	**0.673**	**0.644**	0.647	0.778	0.751	0.756	0.546	0.544	0.552	0.521	0.525	0.532
unrelated	0.658	0.603	0.619	0.774	0.738	0.731	0.505	0.506	0.507	0.624	0.6	0.611
neg. avg.	0.607	0.587	0.592	0.776	0.755	0.755	0.516	0.518	0.523	0.539	0.534	0.539
all avg.	0.623	0.603	0.611	0.772	0.758	0.76	0.56	0.551	0.556	0.591	0.581	0.585

Table 1: LLMs' zero-shot prediction accuracies.

3.1 Zero-shot Prediction Results

Table 1 presents the zero-shot prediction accuracies of the LLMs across the training, validation, and test data splits[4]. The highest accuracy achieved by an LLM on each data split is highlighted in bold. The table also includes averaged accuracies across the adjective groups, highlighted in the shaded rows.

From the results in Table 1, we can observe several notable trends:

- Stable high performance of GPT-4o: GPT-4o consistently achieves the highest accuracies across both positive and negative adjectives. This suggests that GPT-4o has a more refined capability to discern nuanced semantic equivalences and differences compared to the other models, likely due to its advanced training and larger parameter size.
- Effectiveness of positive adjectives: The accuracy results for positive adjectives are generally higher than those for negative adjectives across all models including GPT-4o. This suggests that the models are better at understanding and processing semantic equivalence compared to semantic inequivalence. This may be because LLMs were more frequently exposed to positive adjectives during pre-training.

3.2 Agreement Analysis: among Adjectives and across LLMs

Another trend we would like to investigate is the variability among models. While GPT-4o consistently demonstrates high performance across all adjectives, other models exhibit greater variability in their results. Notably, models like GPT-3.5 and Mistral 7B show unusually low performance with specific adjectives, such as "distinct." These findings emphasize the need for further investigation into the consistency of predictions made by each LLM across different adjectives.

LLM	κ_1	κ_2	Diff.
GPT-3.5	0.262	0.508	0.246
GPT-4o	**0.774**	**0.835**	**0.061**
Llama3.1 8B	0.046	0.362	0.316
Mistral 7B	0.142	0.454	0.312

Table 2: Fleiss's κ coefficients for LLMs.

To achieve this objective, we examine the agreement among the prediction results made with the adjectives for each LLM. Fleiss's κ is adopted instead of Cohen's, as we assess the agreement among eight adjectives across four LLMs.

More specifically, we calculated two types of Fleiss's κ coefficients (Fleiss et al., 1971). The first, κ_1, measures the agreement between gold labels and predictions across two categories, T and F. The second, κ_2, assesses the agreement across four categories, representing combinations of correct labels and predictions. For instance, the category "TF" denotes cases where the gold label is T, but the prediction is F. The κ_2 coefficient is introduced to measure the agreement among the compared items while accounting for tendencies toward mispredictions. Consequently, κ_2 coefficients are generally higher than κ_1 coefficients, and the difference between them may be inversely correlated with prediction accuracy.

[4]The actual models we employed are: GPT-3.5 (`gpt-3.5-turbo-1106`), GPT-4o (`GPT-4o-2024-0513`), Llama3.1 8B (`Meta-Llama-3.1-8B-Instruct`), and Mistral 7B (`Mistral-7B-Instruct-v0.3"`). The GPT models were accessed through the OpenAI API. The open models were used via Hugging Face `transformers` library: https://huggingface.co/docs/transformers/index.

Adjective	κ_1	κ_2	Diff.
identical	0.429	0.624	0.195
the same	**0.464**	**0.636**	**0.172**
similar	0.400	0.594	0.194
related	0.225	0.618	0.392
distinct	-0.118	0.533	0.650
different	0.028	0.525	0.497
dissimilar	0.113	0.422	0.310
unrelated	0.159	0.502	0.343

Table 3: Fleiss's kappa coefficients for adjectives.

Table 2 displays the statistical figures for the test data split, which will be the primary focus of discussion in the remainder of this paper. Each row presents Fleiss's κ_1, κ_2, and the difference between them for each LLM, based on predictions made with the used adjectives. The highest κ values and the smallest difference are shown in bold. The following trends are observed from the table:

- Consistent predictions made by GPT-4o: The high κ_1 values indicate *substantial agreement* (Landis and Koch, 1977) across the adjectives used in the prompts. Furthermore, the small difference between κ_1 and κ_2 supports the high accuracy levels.
- Inconsistent predictions made by other LLMs: Notably, Llama3.1 8B and Mistral 7B exhibit low κ_1 values, which may be related to their insufficient semantic capabilities, as also suggested by their accuracy scores.

These results further endorse the superiority of GPT-4o compared to other models.

We are also interested in investigating the *dual* of this analysis. Specifically, we investigate the agreement among prediction results made by the LLMs for each adjective. Table 3 presents Fleiss's κ_1 and κ_2 coefficients for each adjective, illustrating the level of agreement among the LLMs. It also shows the differences between these coefficients. These results suggest that predictions made with positive adjectives are generally more stable than those made with negative adjectives. Among the positive adjectives, "the same" appears to provide the most stable and accurate predictions.

To conclude this section based on these empirical results, we can affirm that GPT-4o stands out as the best LLM, and among the adjectives, "the same" would be the optimal choice if only one adjective is to be used.

4 Consistency with Canonical Semantic Equivalence Levels

The positive and negative adjectives used in zero-shot prompts are carefully selected to impose varying degrees of semantic equivalence (for positive adjectives) or inequivalence (for negative adjectives). For instance, the positive adjective "identical" demands the strictest interpretation of sameness, whereas "related" permits a broader and more flexible interpretation, capturing a wider range of relatedness.

For a *consistent* LLM, we anticipate that prompts employing "identical" will yield high precision but low recall for instances with identical meanings, reflecting their stricter criteria. In contrast, prompts utilizing "related" are expected to exhibit the opposite trend in the WiC task, with higher recall but lower precision, aligning with their more inclusive interpretation.

We define the term "canonical semantic equivalence levels" to represent the inherent equivalence levels imposed by these adjectives and refer to this concept as the **canonicity assumption** throughout the paper. Furthermore, the term **canonical ranking** denotes the expected ranking of precision and recall metrics associated with these semantic equivalence levels, providing a basis for evaluating the LLMs' consistency across prompts.

4.1 Precision and Recall as Indicators of Consistency

Table 4 summarizes the results for GPT-4o across the adjectives in the test data split. In the table, the "F/P" and "F/R" columns display the precision (P) and recall (R) for the F-instances, respectively. Similarly, the "T/P" and "T/R" columns show the precision and recall for the T-instances. For convenience, the table also includes the F1 scores for both F-instances (F/F1) and T-instances (T/F1), along with the overall accuracy. Figures 4 and 5 provide a visual representation of these results, separated by positive and negative adjectives.

These results clearly demonstrate that GPT-4o aligns perfectly with the canonicity assumption. Specifically, F/R and T/P decrease with the order of positive adjectives, while F/P and T/R increase with the order of positive adjectives. Conversely, F/R and T/P decrease with the order of negative adjectives, while F/P and T/R increase with the order of negative adjectives. However, the results for other LLMs, as summarized in the tables in the

Adjective	F/P	F/R	F/F1	T/P	T/R	T/F1	Acc
identical	0.724	**0.824**	0.771	**0.796**	0.686	0.737	0.755
the same	0.748	0.813	**0.779**	0.795	0.726	0.759	0.769
similar	0.783	0.764	0.774	0.770	0.789	0.779	**0.776**
related	**0.828**	0.654	0.731	0.714	0.864	**0.782**	0.759
distinct	0.783	0.739	0.760	0.753	0.796	0.774	0.767
different	0.792	0.719	0.754	0.742	0.811	0.775	0.765
dissimilar	0.796	0.687	0.738	0.725	0.824	0.771	0.756
unrelated	0.843	0.567	0.678	0.674	**0.894**	0.769	0.731

Table 4: Performance metrics of GPT-4o on the test split.

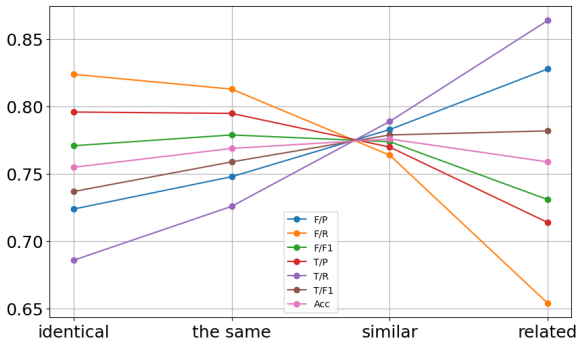

Figure 4: Performance of GPT-4o (positive adjectives).

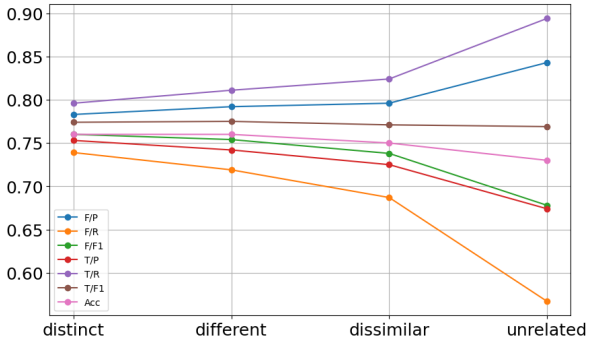

Figure 5: Performance of GPT-4o (negative adjectives).

Appendix B, reveal some flaws, particularly with negative adjectives.

4.2 Rank Correlations as Measures of Semantic Consistency

To quantify an LLM's consistency with the canonical rankings, we use Kendall's rank correlation coefficient (Kendall, 1938). This coefficient measures the agreement between the canonical rank of semantic equivalence levels (derived from the expected order) and the rank actually obtained by the LLM. For example, the canonical rank for T/R with negative adjectives is [4, 3, 2, 1]. GPT-4o achieves this exact rank, resulting in a rank correlation coefficient of 1.0, indicating perfect alignment. In contrast, Llama3.1 8B produces the rank [3, 4, 2, 1] (refer to Table 11), yielding a coefficient of 0.667, reflecting partial inconsistency.

Table 5 aggregates these rank coefficients and presents averaged figures for positive, negative, and all adjectives in the highlighted rows. As shown, GPT-4o aligns fully with the canonicity assumption for both positive and negative adjectives. In contrast, other LLMs exhibit consistency with the assumption for positive adjectives but show inconsistencies for negative adjectives.

In summary, the proposed method for measuring an LLM's consistency with canonical semantic equivalence levels effectively evaluates a key aspect of its semantic capability, with GPT-4o emerging as the most exemplary model. Additionally, the ordering of LLMs based on their overall consistency, as summarized in Table 5, well correlates with the zero-shot accuracy results presented in Table 1.

5 Analysis

In this section, we analyze overall trends in the WiC dataset, the discriminative properties of adjectives, and the linguistic patterns observed in error cases. We particularly focus on the validation split, which consists of 638 instances evenly divided between T- and F-instances, as our reference.

5.1 Overall Trends in the WiC Dataset

In this analysis, we examined the relationship between overall trends in the WiC dataset, including the part of speech of the target word (noun or verb), the number of senses it has, the *richness* of the contextual sentence, and prediction errors. The results showed no clear correlation between the part of speech or sense ambiguity of the target word

		GPT-3.5	GPT-4o	Llama3.1 8B	Mistral 7B
positives	F/P	1.0	1.0	0.667	1.0
	F/R	1.0	1.0	0.667	1.0
	T/P	1.0	1.0	0.667	1.0
	T/R	1.0	1.0	0.667	1.0
pos. avg.		1.0	1.0	0.667	1.0
negatives	F/P	1.0	1.0	1.0	1.0
	F/R	1.0	1.0	0.667	1.0
	T/P	0.0	1.0	0.667	-0.333
	T/R	1.0	1.0	0.667	1.0
neg. avg.		0.75	1.0	0.75	0.667
all avg.		0.875	1.0	0.709	0.833

Table 5: Kendall's rank correlations of model predictions with canonical rankings.

and prediction errors. However, a weak correlation was observed with the richness of the contextual sentence (using token count as the simplest proxy), suggesting that richer contextual clues could be provided by longer contextual sentences. Refer to Appendix C for further details of the results of these investigations.

5.2 Discriminative Property of Adjectives

The choice of adjectives in the prompt can influence prediction outcomes. Table 6 shows the changes in prediction results for each LLM when using positive adjectives. For example, since "identical" represents a higher level of semantic equivalence than "the same," some instances predicted as F with the former may be predicted as T with the latter. In the table, it is shown, for example, that for GPT-3.5, 52 T-instances and 32 F-instances exhibited such changes. In other words, 52 instances shifted from incorrect to correct predictions, while 32 instances changed from correct to incorrect predictions.

The following trends can be observed from this table:

- GPT-4o shows relatively few changes and provides stable predictions regardless of the adjective used.
- For all LLMs, the level of semantic equivalence imposed by "identical" is too high, while that imposed by "related" is too low.
- In models other than Llama3.1 8B, the shift from "the same" to "similar" results in minor changes, while Llama3.1 8B shows significant fluctuations (148 and 180), the causes of which remain unknown.

These observations suggest that "the same" or "similar" can be appropriately used for the WiC task, consistent with the accuracy results discussed in Section 3.2 and shown in Table 1.

5.3 Linguistic Patterns in Error Cases

This analysis seeks to identify trends and weaknesses in LLM predictions, which could guide improvements such as prompt design. It may also uncover issues with the WiC dataset, especially questionable gold labels, raising concerns about the data creation process.

We manually analyzed 150 instances from the validation split where GPT-4o's predictions using the adjective "the same" disagreed with the gold labels. These cases highlight discrepancies rather than outright errors. Table 7 presents the distribution of these instances, categorized by the gold label (T or F) and the part of speech of the target word. In the following, we discuss the analysis results from a high-level perspective with broadly categorized causes. Slightly more detailed descriptions are given in Appendix D.

Analysis of 91 T-instances: The question here is why the LLM predicted these instances as non-synonymous (F), in contrast to the gold label (T). Upon closer examination:

- For the 56 Noun instances, 26 had different meanings, suggesting potential issues with the gold labels. The remaining instances can be attributed to factors such as idiomatic or collocational distinctions (10 instances; Ex.1 in Table 8), metaphorical meanings (9 instances; Ex.2), relative or collective nouns (3 instances;

Predicted as		GPT-3.5			GPT-4o			Llama3.1 8B			Mistral 7B		
F	T	T-inst.	Sign	F-inst.	T-inst.	Sign	F-inst.	T-inst.	Sign	F-inst.	T-inst.	Sign	F-inst.
identical	the same	52	>	32	20	>	10	14	>	2	63	>	42
the same	similar	39	≈	33	19	>	13	146	<	180	41	=	41
similar	related	90	<	122	31	≈	35	30	<	64	51	<	121

Table 6: Comparison of prediction results across different LLMs with shifts in adjectives.

	Noun	Verb	Total
T-instances	56	35	91
F-instances	26	33	59
Total	82	68	150

Table 7: Breakdown of the disagreement cases.

Ex.3), and other various factors detailed in Appendix D.

- For the 35 Verb instances, the LLM appears to identify potential meaning differences based on the syntactic-semantic structure: differences in the meaning of the head noun of a particular case element (20 instances; Ex.4), mainly the object case, idiomatic or collocational expressions (7 instances; Ex.5), and distinctions between transitive and intransitive verbs (5 instances; Ex.6).

Analysis of 59 F-instances: The question here is why the LLM predicted these instances as synonymous (T) when the gold label indicated they were non-synonymous (F). The findings can be summarized as follows:

- Of the 26 noun instances, 12 were genuinely different in meaning, indicating that the LLM may have struggled with distinguishing their often subtle senses. Of the remaining instances, nine should be identified as synonymous, suggesting potential issues with the gold labels, while the remaining five can be attributed to various factors detailed in Appendix D.

- For the 33 verb instances, the distribution of disagreement reasons is relatively similar to that in the T-instance case described above. Specifically, 20 instances are attributed to differences in the meaning of the head noun of a particular case element, three instances to idiomatic or collocational expressions, and six of the remaining instances to distinctions between transitive and intransitive verbs.

In summary, the analysis indicates that a significant number of instances may reflect issues with the gold labels, suggesting that revising the dataset could be beneficial. The distribution of causes behind correct and incorrect predictions for Verb instances was similar across both T-instances and F-instances. This trend suggests that enhancing the prompt to better incorporate relevant linguistic reasoning steps for checking the transitivity of a verb and the semantic category of the object case may help improve overall task performance. Furthermore, identifying figurative meanings and idiomatic expressions warrants consideration, although these may represent areas for further research.

6 Discussion: Ensembling of Predictors

Each predictor, defined by its LLM and prompt adjective, exhibits unique traits, suggesting that combining complementary predictors could improve classification accuracy. In this study, we have 32 unique predictors (4 LLMs paired with 8 adjectives), requiring us to identify the optimal combination from a vast number of possibilities[5]. To address this computational issue, we apply the greedy algorithm outlined in Algorithm 1, using each of the 32 predictors as a seed and selecting the best combination from the resulting combinations. In each iterative step, a meta-classifier is trained on the training split of the WiC dataset using the predictions made by the individual predictors from the current combination.

The results of the ensembling experiment, detailed in Appendix E, revealed that a combination of three GPT-4o-based predictors, using the adjectives "identical" (as the seed), "the same," and "similar" in the prompts, achieved the best performance. This combination yielded a testset accuracy of 0.781, slightly surpassing the 0.776 testset accuracy of the best single predictor driven by "similar" (see Table 4). The resulting meta-classifier demonstrated a modest increase in precision for F-

[5] $\sum_{k=1}^{24} {}_{24}C_k$=16,777,215

	$c1$	$c2$
Ex.1	Always a step behind. [unidiomatic]	Keep in step with the fashions. [idiomatic]
Ex.2	He alone gives me such heartbeats. [literal]	The policeman waited for a heartbeat in vain. [metaphorical]
Ex.3	He is about average in height. [relative]	The snowfall this month is below average. [non-relative]
Ex.4	To liberate the mind from prejudice. [obj:abstract]	To liberate gases. [obj:physical]
Ex.5	Brush aside the objections. [metaphoric]	Brush the dust from the jacket. [literal]
Ex.6	We must not proliferate nuclear arms. [transitive]	The flowers proliferated rapidly all spring. [intransitive]

Table 8: Examples of contextual sentences. In Examples 1 through 3, the target words (underlined) are nouns, while in Examples 4 through 6, they are verbs.

instances (from 0.783 to 0.792), although there was a slight decrease in recall (from 0.764 to 0.763). For T-instances, both precision (from 0.77 to 0.771) and recall (from 0.789 to 0.8) showed slight improvements. Given that the adjective "similar" indicates a lower level of semantic equivalence, while "identical" and "the same" suggest a higher level, these changes in performance are consistent with expectations.

While the overall improvement in these experiments is marginal, the results support the potential effectiveness of ensembling predictors, albeit at the cost of increased computational resources.

7 Related Work

The WiC task is a component of the SuperGLUE benchmark (Wang et al., 2019), which serves as a comprehensive evaluation suite for a range of natural language understanding tasks. Current leaderboard[6] results for the task report top scores of approximately 0.78. While high-performing systems, such as those discussed in (Zhong et al., 2022), leverage advanced machine learning techniques, these studies provide limited linguistic insights and lack detailed analyses specific to the WiC task. Note also that GPT-4o has nearly matched this level of accuracy with the use of baseline zero-shot prompts, as shown in Table 1.

Relatively few studies have systematically evaluated LLMs on the WiC task. Brown et al. (2020) reported an accuracy of 0.494 using an early version of GPT-3 in a few-shot setting, highlighting challenges in comparing two items. Subsequently, Laskar et al. (2023) achieved an accuracy of 0.621 with GPT-3.5-turbo in a zero-shot setting. However, these results may now be outdated. Moreover, neither study provides an in-depth analysis of the WiC task or explores broader semantic issues.

More recently, Wang and Zhao (2024) achieved a significantly higher accuracy of 0.843 on the validation split with GPT-4, surpassing human performance. Their work introduced a metacognitive prompting method, incorporating sophisticated prompts with detailed steps and explanations through few-shot demonstrations. Despite its remarkable performance, this study seems unconcerned with examining the semantic capabilities of LLMs in the context of the WiC task.

Similar to the present work, Hayashi (2024) evaluated LLMs' ability to identify semantic equivalence in context using the WiC task. They employed GPT models to generate textual descriptions explaining the semantic usage of target words, which were then used to train a binary classifier. The results demonstrated GPT-4's strong performance, with its descriptions being both compact and precise. However, their approach relied solely on the adjective "identical," overlooking the potential impact of other adjectives on LLM behavior.

8 Conclusion

This study introduced a method for evaluating LLMs' capability to identify lexical semantic equivalence using the WiC task, with a focus on both overall capability and consistency across levels of semantic equivalence. The method leverages zero-shot prompts incorporating adjectives that express varying degrees of semantic equivalence.

We evaluated several LLMs, including large proprietary models and smaller open-source ones, highlighting the superior semantic capabilities of GPT-4o. Manual analysis revealed potential issues in both the LLMs driven by zero-shot prompts and the WiC dataset, pointing to opportunities for refining prompt design and enhancing dataset quality.

Our future research will build on the proposed framework to investigate graded semantic similarity (Erk et al., 2013; Armendariz et al., 2020; Schlechtweg et al., 2021), potentially offering deeper insights into the sensitivity and semantic capabilities of LLMs.

[6] https://super.gluebenchmark.com/leaderboard

Limitations

We are aware of the following limitations in this study:

- We evaluated only four LLMs: two proprietary models and two open-source models with relatively smaller parameter sizes. Including open-source models with larger parameter sizes could potentially result in significantly improved performance, potentially making them competitive with the proprietary models.
- We did not perform an exhaustive search for classifiers or extensive hyperparameter tuning during the training of the meta-classifier. Different settings and tuning strategies could lead to substantially better ensemble results.
- Since the primary focus of this study was not on achieving state-of-the-art performance, we did not optimize the prompts used in our experiments. Our error analysis, as discussed in the paper, may help in developing better prompts.
- Our manual inspection for the error analysis was based on a small sample size. A more extensive examination with a larger dataset would provide deeper insights.
- We did not conduct a cost analysis of using OpenAI's API.

Acknowledgments

This work was supported by JSPS KAKENHI Grant Number 22K12723.

References

Eneko Agirre and Philip Edmonds, editors. 2006. *Word Sense Disambiguation: Algorithms and Applications*. Springer, Dordrecht, the Netherlands.

Carlos Santos Armendariz, Matthew Purver, Matej Ulčar, Senja Pollak, Nikola Ljubešić, and Mark Granroth-Wilding. 2020. CoSimLex: A resource for evaluating graded word similarity in context. In *Proceedings of the Twelfth Language Resources and Evaluation Conference*, pages 5878–5886, Marseille, France. European Language Resources Association.

Michele Bevilacqua, Tommaso Pasini, Alessandro Raganato, and Roberto Navigli. 2021. Recent trends in word sense disambiguation: A survey. In *Proceedings of the Thirtieth International Joint Conference on Artificial Intelligence, IJCAI-21*, pages 4330–4338. International Joint Conferences on Artificial Intelligence Organization. Survey Track.

Tom B. Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, Sandhini Agarwal, Ariel Herbert-Voss, Gretchen Krueger, Tom Henighan, Rewon Child, Aditya Ramesh, Daniel M. Ziegler, Jeffrey Wu, Clemens Winter, Christopher Hesse, Mark Chen, Eric Sigler, Mateusz Litwin, Scott Gray, Benjamin Chess, Jack Clark, Christopher Berner, Sam McCandlish, Alec Radford, Ilya Sutskever, and Dario Amodei. 2020. Language models are few-shot learners. *CoRR*, abs/2005.14165.

Sašo Džeroski and Bernard Ženko. 2004. Is combining classifiers with stacking better than selecting the best one? *Machine learning*, 54(3):255–273.

Katrin Erk, Diana McCarthy, and Nicholas Gaylord. 2013. Measuring word meaning in context. *Computational Linguistics*, 39(3):511–554.

J.L. Fleiss et al. 1971. Measuring nominal scale agreement among many raters. *Psychological Bulletin*, 76(5):378–382.

Yoshihiko Hayashi. 2024. Reassessing semantic knowledge encoded in large language models through the word-in-context task. In *Proceedings of the 2024 Joint International Conference on Computational Linguistics, Language Resources and Evaluation (LREC-COLING 2024)*, pages 13610–13620, Torino, Italia. ELRA and ICCL.

Nancy Ide and Yorick Wilks. 2006. Making sense about sense. In Edmonds Philip Agirre, Eneko, editor, *Word Sense Disambiguation: Algorithms and Applications*, pages 47–73. Springer Netherlands, Dordrecht.

Albert Q. Jiang, Alexandre Sablayrolles, Arthur Mensch, Chris Bamford, Devendra Singh Chaplot, Diego de las Casas, Florian Bressand, Gianna Lengyel, Guillaume Lample, Lucile Saulnier, Lélio Renard Lavaud, Marie-Anne Lachaux, Pierre Stock, Teven Le Scao, Thibaut Lavril, Thomas Wang, Timothée Lacroix, and William El Sayed. 2023. Mistral 7b. *Preprint*, arXiv:2310.06825.

M. G. Kendall. 1938. A new measure of rank correlation. *Biometrika*, 30(1/2):81–93.

J Richard Landis and Gary G. Koch. 1977. The measurement of observer agreement for categorical data. *Biometrics*, 33 1:159–74.

Md Tahmid Rahman Laskar, M Saiful Bari, Mizanur Rahman, Md Amran Hossen Bhuiyan, Shafiq Joty, and Jimmy Huang. 2023. A systematic study and comprehensive evaluation of ChatGPT on benchmark

datasets. In *Findings of the Association for Computational Linguistics: ACL 2023*, pages 431–469, Toronto, Canada. Association for Computational Linguistics.

LlamaTeam-AI@Meta. 2024. The llama 3 herd of models. *Preprint*, arXiv:2407.21783.

George A. Miller, Richard Beckwith, Christiane Fellbaum, Derek Gross, and Katherine J. Miller. 1990. Introduction to WordNet: An On-line Lexical Database*. *International Journal of Lexicography*, 3(4):235–244.

Roberto Navigli. 2009. Word sense disambiguation: A survey. *ACM Comput. Surv.*, 41(2).

OpenAI. 2023. Gpt-4 technical report. *Preprint*, arXiv:2303.08774.

Alexander Panchenko, Eugen Ruppert, Stefano Faralli, Simone Paolo Ponzetto, and Chris Biemann. 2017. Unsupervised does not mean uninterpretable: The case for word sense induction and disambiguation. In *Proceedings of the 15th Conference of the European Chapter of the Association for Computational Linguistics: Volume 1, Long Papers*, pages 86–98, Valencia, Spain. Association for Computational Linguistics.

Mohammad Taher Pilehvar and Jose Camacho-Collados. 2019. WiC: the word-in-context dataset for evaluating context-sensitive meaning representations. 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 1267–1273, Minneapolis, Minnesota. Association for Computational Linguistics.

Yael Ravin and Claudia Leacock, editors. 2000. *Polysemy: Theoretical and Computational Approaches*. Oxford University Press, Oxford.

Dominik Schlechtweg, Nina Tahmasebi, Simon Hengchen, Haim Dubossarsky, and Barbara McGillivray. 2021. DWUG: A large resource of diachronic word usage graphs in four languages. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pages 7079–7091, Online and Punta Cana, Dominican Republic. Association for Computational Linguistics.

Alex Wang, Yada Pruksachatkun, Nikita Nangia, Amanpreet Singh, Julian Michael, Felix Hill, Omer Levy, and Samuel Bowman. 2019. Superglue: A stickier benchmark for general-purpose language understanding systems. In *Advances in Neural Information Processing Systems*, volume 32. Curran Associates, Inc.

Yuqing Wang and Yun Zhao. 2024. Metacognitive prompting improves understanding in large language models. In *Proceedings of the 2024 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (Volume 1: Long Papers)*, pages 1914–1926, Mexico City, Mexico. Association for Computational Linguistics.

Qihuang Zhong, Liang Ding, Yibing Zhan, Yu Qiao, Yonggang Wen, Li Shen, Juhua Liu, Baosheng Yu, Bo Du, Yixin Chen, Xinbo Gao, Chunyan Miao, Xiaoou Tang, and Dacheng Tao. 2022. Toward efficient language model pretraining and downstream adaptation via self-evolution: A case study on superglue. *Preprint*, arXiv:2212.01853.

A More Details on the WiC Dataset

Table 9 presents the configuration of the dataset, detailing its division into training, validation, and test splits, with corresponding instance counts. Notably, the dataset reveals an imbalance in the Noun/Verb ratio between the training split and the validation/test splits. This disparity may affect model performance, as the distribution of target word types differs across the splits, potentially influencing how well the model generalizes across different parts of the dataset.

Split	Instances	Nouns	Verbs
Training	5,428	49%	51%
Validation	638	62%	38%
Test	1,400	59%	41%

Table 9: Overview of the WiC dataset.

B Details of the Consistency Results

The test split results for GPT-3.5, Llama3.1 8B, and Mistral are presented in Tables 10, 11, and 12, respectively, mirroring the information shown for GPT-4o in Table 4. Similar to GPT-4o, these LLMs exhibit consistent behavior with positive adjectives. However, unlike GPT-4o, they display somewhat inconsistent trends with negative adjectives.

C More Details on the Overall Trends Analysis

There could be various causes for prediction errors, including the part-of-speech (noun or verb) of a target word, inherent semantic ambiguities of the target word, vague or insufficient contextual clues, and potentially incorrect gold labels in the dataset. Among these, the first three aspects are examined in the following. Note that the predictor made more F predictions than T predictions (342 versus 296). However, this difference is not statistically

adjective	F/P	F/R	F/F1	T/P	T/R	T/F1	Acc
identical	0.574	0.893	0.699	0.760	0.339	0.468	0.616
the same	0.626	0.800	**0.702**	0.723	0.521	0.606	**0.661**
similar	0.644	0.676	0.660	0.659	0.627	0.643	0.651
related	0.717	0.304	0.427	0.558	0.880	0.683	0.592
distinct	0.499	**0.991**	0.664	0.400	0.006	0.011	0.499
different	0.563	0.917	0.698	**0.777**	0.289	0.421	0.603
dissimilar	0.661	0.603	0.631	0.635	0.691	0.662	0.647
unrelated	**0.765**	0.344	0.475	0.577	**0.894**	**0.701**	0.619

Table 10: Performance metrics of GPT-3.5.

adjective	F/P	F/R	F/F1	T/P	T/R	T/F1	Acc
identical	0.626	0.661	0.643	0.641	0.604	0.622	**0.633**
the same	0.580	0.801	0.673	0.679	0.420	0.519	0.611
similar	0.687	0.317	0.434	0.556	0.856	**0.674**	0.586
related	**0.702**	0.094	0.166	0.515	**0.960**	0.670	0.527
distinct	0.507	**0.930**	0.657	0.581	0.097	0.166	0.514
different	0.510	0.983	**0.671**	**0.760**	0.054	0.101	0.519
dissimilar	0.540	0.710	0.613	0.576	0.394	0.468	0.552
unrelated	0.557	0.070	0.124	0.504	0.944	0.657	0.507

Table 11: Performance metrics of Llama3.1 8B.

adjective	F/P	F/R	F/F1	T/P	T/R	T/F1	Acc
identical	0.612	0.807	**0.696**	0.717	0.489	0.581	0.648
the same	0.660	0.663	0.661	0.661	0.659	0.660	**0.661**
similar	0.700	0.536	0.607	0.624	0.770	**0.689**	0.653
related	**0.733**	0.184	0.295	0.533	**0.933**	0.679	0.559
distinct	0.500	**1.000**	0.667	0.000	0.000	0.000	0.500
different	0.508	0.994	0.672	0.862	0.036	0.069	0.515
dissimilar	0.517	0.991	0.679	**0.895**	0.073	0.135	0.532
unrelated	0.605	0.636	0.620	0.617	0.586	0.601	0.611

Table 12: Performance metrics of Mistral 7B.

significant, as indicated by the chi-square test p-value of 0.069.

Part-of-speech: Table 13 presents the accuracy scores for nouns and verbs across T-instances, F-instances, and all instances, along with the p-values from the chi-square tests. The table suggests that while there is a statistically significant difference in accuracy between nouns and verbs in F-instances, with nouns being more accurate, there is no significant difference in T-instances or when considering all instances together. Therefore, it can be said that there is no clear trend indicating whether this LLM achieves better accuracy for target words that are verbs or nouns.

Number of senses: The polysemous aspect of a target word may affect the accuracy of the WiC task. As an initial step, we examine the relationship between the number of senses a target word has and the accuracy of the WiC task. Table 14 compares the average number of senses in WordNet (Miller et al., 1990) for correctly predicted OK instances and incorrectly predicted NG instances across T-instances, F-instances, and all instances, along with the p-values from one-sided t-tests. According to the table, there is no substantial difference in the average number of WordNet senses between correctly and incorrectly predicted instances for both T-instances and all instances. However, for F-instances, the average number of senses for correctly predicted instances is substantially higher than for incorrectly predicted ones. In other words, in F-instances, where the contextual meanings of the target word differ, having a target word with a broader range of meanings is associated with higher accuracy. This implies that, from a WSD perspective, the greater the number of sense candidates, the higher the tendency for predictions of differing meanings, which could be a plausible result.

	Noun	Sign	Verb	p-value
T-instances	0.72	<	0.748	0.10
F-instances	0.859	>	0.725	*0.03
All	0.795	>	0.737	0.07

Table 13: Break down of the accuracy scores by part-of-speech.

	Correct	Sign	Incorrect	p-value
T-instances	4.62	<	5.01	0.77
F-instances	6.78	>	5.46	*0.02
All	5.75	>	5.2	0.09

Table 14: Average number of WordNet senses.

	Correct	Sign	Incorrect	p-value
T-instances	8.417	>	7.845	0.1
F-instances	8.473	>	8.033	0.19
All	8.466	>	7.924	0.06

Table 15: Average length of contextual sentences.

Length of contextual sentence: This aspect is primarily related to the WiC dataset rather than the LLM's capabilities or characteristics. As the richness of a contextual sentence in each data instance is difficult to quantify directly, we use the length of a contextual sentence, measured by the number of tokens, as a practical proxy. Table 15 compares the average length of contextual sentences for cor-

rectly predicted instances and incorrectly predicted instances across T-instances, F-instances, and all instances, along with the p-values from one-sided t-tests. The results indicate that, on average, contextual sentences for correctly predicted instances are slightly longer than those for incorrectly predicted instances. Although these differences are not statistically significant at the 0.05 level, longer contextual sentences may be more effective in providing clues for semantic distinction.

D More Details on the Error Analysis

Table 16 classifies the Noun instances by the potential causes of disagreement. As observed from the T-instances column, the LLM appears highly sensitive to syntactic and expressive features. For nearly half of the T-instances, the gold annotations are questionable, as indicated by the 26 disputably polysemous instances. On the other hand, the LLM predicted one-third (nine) of the potentially polysemous instances as synonymous. While this observation alone does not allow for a strong conclusion, it suggests that the LLM might be making more appropriate decisions regarding the range of semantic denotations.

Type	T-instances	F-instances
Polysemous	26	12
Idiom/Collocation	10	2
Synonymous	1	9
Metaphoric	9	1
Relative/Group Noun	3	1
Subtle nuance	5	1
Technical term	2	0
Total	56	26

Table 16: Breakdown of causes of disagreement for Noun instances.

Table 17 classifies the Verb instances by the potential causes of disagreement. From this table, we can observe that both the gold annotations and the LLM primarily base their decisions on verb frame syntax and semantics, as evidenced by the significant number of instances classified by case element meaning (labeled as "Case") and transitive/intransitive distinctions (labeled as "VI-VT"). This suggests that a prompting strategy directing the LLM to consider these features before making final predictions may be beneficial.

Type	T-instances	F-instances
Case	20	20
VI-VT	5	6
Idiom/Collocation	7	3
Syntax	0	3
Synonymous	3	1
Total	35	33

Table 17: Breakdown of causes of disagreement for Verb instances.

E Details of the Ensembling Experiment

The preliminary experiment on ensembling predictors was conducted using the following settings.

E.1 Ensembling Algorithm

A simple stacking algorithm (Džeroski and Ženko, 2004) was employed, which uses the predictions from the selected predictors as features. In the experiment, we also incorporated the agreements between each pair of predictors, enhancing the input to the meta-classifier and resulting in slightly better accuracies in most cases.

E.2 Meta-classifier

Among various popular classification algorithms, we reported results using a meta-classifier with a multi-layer perceptron provided by scikit-learn [7]. The network was configured with three hidden layers, with dimensions of 32, 128, and 32.

E.3 Search Algorithm

To identify an optimal combination of predictors, we employed a greedy search algorithm, as outlined in Algorithm 1. The algorithm operates in the following manner:

- Initialization: It begins with a single predictor as a seed.
- Iterative Process: Predictors are iteratively added based on their potential to improve the accuracy score on the test set. A meta-classifier is trained on the training split of the WiC dataset, utilizing the predictions made by the individual predictors in the current combination.
- Termination: This process continues until no further improvements can be made.

[7] https://scikit-learn.org/stable/modules/neural_networks_supervised.html

Algorithm 1 Greedy Selection of Predictors

```
1: Input: Seed predictor, seed_pred, and List of candidate predictors, cand_preds
2: Output: Selected predictors, sel_preds
3: sel_preds ← [seed_pred]
4: max_accuracy_so_far ← seed_pred.accuracy_on_val_set
5: cand_preds ← remove(cand_preds, best_pred)
6: while cand_preds is not empty do
7:    cand_pred ← greedily_select_a_cand_pred(cand_preds)
8:    new_model ← train_meta_classifier_on_train_split(sel_preds + [cand_pred])
9:    new_accuracy ← evaluate_meta_classifier_on_val_set(new_model)
10:   if new_accuracy > max_accuracy_so_far then
11:      sel_preds ← sel_preds + [cand_pred]
12:      cand_preds ← remove(cand_preds, cand_pred)
13:      max_accuracy_so_far ← new_accuracy
14:   else
15:      break
16:   end if
17: end while
18: Return sel_preds
```

- Final Selection: The algorithm returns a set of predictors that is presumed to be optimal for the given seed.

After running the algorithm, we evaluate the meta-classifier, which uses the selected predictors, on the test split.

We run this algorithm 32 times, each time using one of the 32 current predictors (combinations of 4 LLMs and 8 adjectives) as a seed. The best combination of predictors is selected from these 32 runs. The final metric values for this best combination are detailed in Section 6.

In the algorithm, the most crucial external function is greedily_select_a_cand_pred in line 7. In this greedy search function, the score S for each of the candidate predictors is calculated as follows and the one yielding the maximum value is selected.

$$S = (|A| + |B|) \times |C|$$

where:

- $|A|$ is the number of instances correctly classified by the current predictors.
- $|B|$ is the number of instances correctly classified by the additional predictor.
- $|C|$ is the number of instances correctly classified by both.

In other words, this score function aims to maximize the number of instances correctly classified by both the existing predictors and the new predictor, while maintaining the number of instances already correctly classified. The validity of this score function is supported by its alignment with results from an exhaustive search of a few candidate predictors, suggesting that it can effectively select near-optimal combination of predictors.