Decoding Symbolism in Language Models

Meiqi Guo **Rebecca Hwa** **Adriana Kovashka**
Department of Computer Science, University of Pittsburgh, Pittsburgh PA, USA
meiqi.guo@pitt.edu {hwa, kovashka}@cs.pitt.edu

Abstract

This work explores the feasibility of eliciting knowledge from language models (LMs) to decode symbolism, recognizing something (e.g., roses) as a stand-in for another (e.g., love). We present our evaluative framework, ***Symb**olism **A**nalysis* (**SymbA**), which compares LMs (e.g., RoBERTa, GPT-J) on different types of symbolism and analyzes the outcomes along multiple metrics. Our findings suggest that conventional symbols are more reliably elicited from LMs while situated symbols are more challenging. Results also reveal the negative impact of the bias in pre-trained corpora. We further demonstrate that a simple re-ranking strategy can mitigate the bias and significantly improve model performances to be on par with human performances in some cases.

1 Introduction

Symbolism is an important literary device that helps to persuade ideas concisely (Symons, 2014). A system that can decode symbolism should recognize that one item (e.g., a baby) is a stand-in for something else (e.g., innocence). It has applications in understanding persuasive texts as well as the visual media (Liu et al., 2022; Guo et al., 2021; Akula et al., 2023). For example, a social media moderator needs to know that certain seemingly benign phrase or object may signal some banned behavior; an intelligent writing tutor should recognize (in)appropriate usages of symbolism in student essays; a persuasive text/image generator may convey its message more effectively by appropriate uses of symbolism. With these potential applications in mind, this work explores whether state-of-the-art LMs encapsulate enough implicit and abstract knowledge to infer symbolic relationships. Specifically, we ask: given some observed physical object or content (referred to as the *signifier*), can LMs predict an appropriate corresponding conceptual symbolic reference (referred to as the *signified*)[1]?

Decoding symbolism is a challenging task (even for humans). First, symbols serve many different purposes, from representing figures of speech and modes of thought to denoting various signs, passwords, and customs (Jones, 1918). Thus, some types of signifier-signified relationships may be more difficult to decode than others. Prior work suggests that LMs encapsulate some commonsense knowledge (Speer et al., 2017); therefore, we anticipate LMs may capture the more semantically related symbolic relationships (e.g., a fork as a symbol of food because it is *UsedFor* eating), but what about those involving a longer chain of reasoning? How do additional factors such as the complexity of the LM and the choice of the prompt impact the performances of different LMs? Second, symbols may be situational: the same signifier may be a stand-in for different references under different scenarios. For example, while a baby often represents innocence, when depicted as being held by a harried parent, that baby comes to symbolize burden and responsibility. It is crucial to examine the extent to which LMs can identify the appropriate signified concept based on the situational context. Finally, while symbolism is often used to emphasize common human concepts (e.g., *love*), it is also an apt device to represent rare, difficult concepts. This dichotomy poses a challenge for LMs, which are susceptible to biases from their pre-training corpora (Shwartz and Choi, 2020; Guo et al., 2020; Holtzman et al., 2021) because the bias leads to a strong preference for the more commonly signified concepts (e.g., *love*) while penalizing symbolic links with rarer words.

To assess their capacity to decode symbolism, we have developed an evaluative framework called SymbA (**Sym**bolism **A**nalysis) to empirically com-

[1] Our terminologies are derived from media studies (Williamson, 1978) rather than any specific linguistic theory for broader NLP applications.

Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics
Volume 1: Long Papers, pages 3311–3324
July 9-14, 2023 ©2023 Association for Computational Linguistics

pare three classes of LMs: word embedding (Word2Vec), which serves as a baseline, masked (BERT and RoBERTa), and autoregressive (GPT-2 and GPT-J). The evaluative task is: given a prompt containing a signifier, return a ranked list of potential signified concepts. Models are also evaluated on a multiple-choice task against a human upperbound.

Two sets of evaluative data[2] are curated to highlight different aspects of the symbolic relationships. One set consists of *conventional symbol pairs* that we compiled from commonly used symbols in English literature, which tend to be context invariant. The other is a subset that we sampled from a visual advertisement corpus (Hussain et al., 2017) that contains *situated symbolic pairs*; the local context immediately surrounding the signifier and the intended signified are annotated by humans. By modifying the prompt to exclude/include the local description, we observe the impact of the situated context. Additional fine-grained categorizations of the evaluative data help to reveal the characteristics of symbolic relationships that pose the greatest challenge to the LMs. Moreover, we propose ways for quantifying and tempering the bias in LMs favoring commonly signified concepts.

Overall, we find that LMs can capture aspects of symbolic knowledge, with the newer, larger models significantly outperform their previous iterations. Surprisingly, advanced LMs performed better on conventional symbolism (more idiomatic) than symbolism in ads (more semantically related), where they fared significantly worse than Word2Vec. This reveals the negative impact of the hypothesized bias in pre-training corpora. We demonstrate that the proposed debiasing method improves performance; the increase is the most dramatic for *situated* ads symbols (e.g., RoBERTa improved by 260%). After reranking, GPT-J and RoBERTa achieve performances comparable to human on the multiple choice task. Further analyses suggest LMs perform better on explicit relationships such as *UsedFor* than implicit ones, and the debiased models are sufficiently robust with respect to the probing prompts.[3]

2 Background

Decoding Symbolism The use of symbolism is an important literary device that helps authors to write more persuasively and convey more ideas in fewer words. To gain a deeper understanding of what is communicated, NLP systems need to be able to decode symbolic usages in text. To our knowledge, this is an under-explored problem in NLP, though there has been related work on recognizing metaphoric and idiomatic usages (Chakrabarty et al., 2022; Neidlein et al., 2020; Kurfalı and Östling, 2020; Shutova et al., 2016; Li et al., 2013). Like symbols, metaphors and idioms also replace some intended target concept with different words; however, a metaphor emphasizes *some common property* it shares with the target concept. An idiom is an expression that conveys a fixed target meaning that is not composed from the literal meaning of its individual words. In contrast, a symbol serves as a *stand-in* for a more complex and abstract concept under certain context; it may not share any obvious property with the abstract concept, and it may not be associated with solely one concept (Langacker, 1996).

Beyond metaphor recognition, our objectives are also aligned with metaphor interpretation, which aims to connect the surface and target concepts (Rosen, 2018; Shutova, 2010; Veale and Hao, 2008; Kintsch, 2000). Some prior approaches explored connecting them through shared features or logical sequences, but such a path may not exist for symbolism. Instead of searching for a path through a discrete space, we elicit the signified associated with the given signifier from the implicit representation of a trained language model.

A somewhat related idea was recently investigated by Chakrabarty et al. (2021) in which a metaphoric verb is masked so that the language model could predict a more literal verb given the surrounding context. Different from our objectives, however, their work does not require the language model to capture the relationship between the metaphoric verb and the literal verb; in contrast, our work explicitly investigates whether a language model will predict the appropriate signified when probed with a signifier.

Language Models Since language models serve as the basis of our symbol decoder, we discuss two common approaches. Their training regimes lead to different token representation that may impact the

[2] Both datasets are predominantly expressed in English so that our studies focus on the interpretation of symbolism from a Western cultural perspective (Jones, 1918); this English-dominant scope matches the pre-training corpora of LMs.

[3] Our code and data is publicly available at : https://github.com/MeiqiGuo/ACL2023-SymbA

ability of each to associate an appropriate signified with the given signifier.

Autoregressive Language Models are trained to predict the ground-truth next token given previous ones. Pretrained autoregressive language models such as GPT (Radford et al., 2018, 2019; Brown et al., 2020) are able to generate fluent and coherent human-sounding sentences; however, they can only generate text along one direction and have no access to the context on the other side.

Masked Language Models are trained to predict the ground-truth masked token given the right and left context. BERT and its variations fall in this group (Devlin et al., 2019; Liu et al., 2019). Bidirectional attention helps the model learn more complete representations of tokens than the unidirectional models. Consequently, masked language models usually achieve better performance after fine-tuning on downstream NLP tasks than the autoregressive models. However, they underperform on text generation because of the masking scheme and the independence assumption between masked tokens (Wang and Cho, 2019).

Scoring by PMI PMI has been used for scoring candidates in many NLP applications, including zero-shot question answering (Brown et al., 2020), surface form competition (Holtzman et al., 2021), dialogue generation (Zhou et al., 2019; Yao et al., 2017) as well as knowledge elicitation from language models (Davison et al., 2019). In the context of this work, it serves as a means to re-rank the strength of association between signfier-signified pairs and a method of analysis to identify situations for which re-ranking improves performance.

3 SymbA Probe

We introduce the SymbA (Symbolism Analysis) framework for evaluating language model's ability to decode symbols. SymbA includes 1066 symbolic pairs from two data sources, a debiasing method and two analytical tools.

3.1 Symbolism Data Sources

Conventional Literary Symbolism Based on the sheer volume of its pretraining text, a language model should have encountered many conventional, widely-used symbols. Such symbolic relationships are often taught in high-school English classes as well as other writing courses.

To curate a collection of conventional symbolism, we consulted multiple sources, including Brown (1997), Hancock (1972), ConceptNet (Speer et al., 2017) and an educational website[4]. Our dataset consists of 132 signifiers that are commonly used in literature. It covers a diverse set of signifiers that can be categorized into eleven groups of semantically related items, as shown in Tab. 1. Of the eleven types, Object, Animal, Plants and Nature are the most frequent types; while Action, Directions, Number and Christianity have limited instances. There are 536 signifier-signified pairs since each signifier may have several signifieds. The vocabulary size of the signified is 333.

Signifier Type	Count	Example (signifier: signified)
Color	12	pink: femininity, flesh, ...
Nature	17	dawn: hope, illumination
Plants	18	rose: beauty, love, ...
Weather	9	mist: confusion, mystery, ...
Animal	19	lion: pride, power, ...
Setting	14	forest: evil, mystery, ...
Object	22	trophy: victory
Action	3	kiss: intimacy, fellowship, ...
Number	7	seven: creation, abundance, ...
Christianity	7	angel: messenger, purity, ...
Directions	4	west: descending, old

Table 1: Our conventional symbolism dataset groups the signifiers into 11 types.

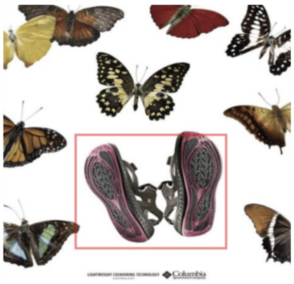

Figure 1: A situated symbolism sample. Each sample contains a signifier-signified pair and a localized description. Here the signifier is *sandal*; the signified is *freedom*; the localized description is *sandals that look like a butterfly*.

Situated Symbolism Symbols that arose from specific circumstances, which we refer to as *situated symbolism*, are not idiomatic or set by conventions. There is a great deal of variation in terms of the challenge of the task. At an extreme, one might consider a literary author taking chapters to develop and evolve a symbol, such as the meaning of Hester Prynne's "A" in "The Scarlet Letter"; such a grand scale is out of the scope of this work. Here, we

[4] https://www.dvusd.org/cms/lib/AZ01901092/Centricity/Domain/2891/Gawain%20Symbols.pdf

focus on a more manageable context range, limited to the message conveyed in a static visual advertisement (Hussain et al., 2017). We chose this domain because the ad offers a self-contained narrative for the context; any symbolic reference has to either be resolved through information directly presented in the ad or relies on commonly shared knowledge by the viewers.

The advertisement dataset provides a bounding box around the signifier in each ad image and its corresponding signified symbol reference (e.g. danger, happiness, etc.). The vocabulary size of the signified is 53. However, aside from the bounding box, there is no textual annotation that describes the signifier. Thus, we supplemented their dataset with additional annotations.[5] We opted to create a balanced dataset for evaluation by randomly sampling 10 ads from each signified group for a total of 530 instances.[6] We then asked 11 annotators (3 authors and 8 non-authors) to describe the visual signifier in the bounding box with a short natural language phrase or sentence, noted as *localized description*.[7] Because each description is typically a short phrase or a sentence, we then manually annotated the head noun of the description as the signifier (referred as a task *without context*); the localized description is considered as the *context* for the signifier (cf. Fig 1, *sandal* is selected as the signifier, while *that look like a butterfly* is a context stimulus).

Human Evaluation The language model selects the signified from a large fixed set (333 for literary symbols and 53 for ad symbols); the same task may be challenging for a human. An alternative is to conduct a simpler experiment: we asked humans to select the correct answer from 4 candidates (negative candidates were randomly chosen from the fixed vocabulary). We compute the Krippendorff's alpha score (Krippendorff, 2011) for measuring the adjusted inter-rater agreement. The score is 0.64 for the conventional symbols; and 0.60 or 0.57 for the ad symbols, respectively with or without the situated context.[8] These scores suggest moderate or substantial inter-rater agreement (Landis and Koch, 1977; Hartling et al., 2012), which demonstrates the quality of our data. We also report the human performance on completing these tasks in Sec 4.3.

3.2 Debiasing Method

Our hypothesis is that a model's prediction candidates that appear more frequently in the pre-training corpus tend to be ranked higher than its appropriate position; similarly, rarer signifieds may be unfairly penalized. For example, the language model may consider "freedom" as a more probably predicted candidate than "serenity" since the latter word has been rarely seen during the pre-training. In order to reduce the bias effect brought by the pre-training frequency, we propose a new approach for ranking the predictions by considering the prior probability of each candidate.

Assuming that x represents the signifier, y represents the signified, t represents the prompt (e.g. "is a symbol of") and θ represents the parameters of the language model, the conditional probability of y is represented as $p(y|x,t,\theta)$. Commonly, the top candidate y_{pred} is selected by having the highest probability: $y_{pred} = argmax_y\, p(y|x,t,\theta)$ (Petroni et al., 2019; Jiang et al., 2020). In our approach, we re-rank the previously-selected top k candidates after normalizing the conditional probability by the prior probability of each candidate:

$$y_{pred}(k) = argmax_{y \in Y_k}\, log \frac{p(y|x,t,\theta)}{p(y|t,\theta)}$$

where Y_k is the set of previously-selected top k candidates. The intuition is that a high $p(y|x,t,\theta)$ might not mean a good collocation between x and y if $p(y|t,\theta)$ is also high. For example, a certain signified (e.g. love) might have a high probability when following the prompt (e.g. "is a symbol of"), no matter which signifier is given. Our re-ranking approach aims to reduce this bias effect.

3.3 Analytical Tools

Semantic Relatedness For quantitatively measuring the semantic relatedness between the symbolic pair, we develop a heuristic metric based on the pointwise mutual information (PMI). This metric measures how frequently a signifier-signified

[5] We considered a captioning generation model (Anderson et al., 2018) on the COCO datasets; however, the domain gap between symbolic and general non-ad image was too large for the resulting captions to prompt language models.

[6] We manually checked each instance and made sure there is no offensive content.

[7] The coding manual is in Appendices. We qualitatively checked the inter-rater agreement between 3 annotators for 20 samples. While they do not always use the exact same wording, their descriptions agree 90% of the time.

[8] The raw agreement scores (Artstein and Poesio, 2008) between two annotators are: 72.7% for conventional symbols, 70% for ad symbols with situated context, and 67.9% without.

Relationship Type	Count	Example (signifier - signified)	Example (situated signifier - signified)
UsedFor	52	makeup - beauty	cartoon candy running on a treadmill - health
HasProperty	46	child - youth	workers sitting closely in a sofa - comfort
RelatedTo	47	mountain - adventure	cigarette smoke in the shape of mushroom cloud - danger
Others	94	chocolate - love	foot stepping on tombstone - death
Indirect	116	giraffe - love	shoes made out of red bull cans - strong

Table 2: Relationship types of *signifier-signified* in the set of advertising symbolism.

pair co-occur within the same sentences in a textual corpus. We assume that if the pair co-occur frequently, then the symbolic relationship leans towards a factoid thus is considered as "easy" knowledge; on the other hand, if the pair rarely co-occur in the same sentence, then it leans towards implicit commonsense reasoning thus considered as "hard" knowledge. We use this metric for measuring the knowledge difficulty.

For a given signifier x and signified y, the PMI score is computed by

$$pmi(x,y) = log\frac{p(x,y)}{p(x)p(y)} = log\frac{\frac{N(x,y)}{N}}{\frac{N(x)}{N}\frac{N(y)}{N}}$$

where $N(x,y)$ is the number of sentences containing both x and y; $N(x)$ or $N(y)$ is respectively the number of sentences containing x or y; N is the total number of sentences in the corpus. A higher PMI score indicates easier knowledge.

Symbolic Relationship Types For investigating the fine-grained types of each symbolic relationship, we further annotate each signifier-signified pair according to a pre-defined taxonomy of commonsense relationships (Speer et al., 2017). The symbolic associations used in ads are creative and diverse, while the conventional set mostly contains the narrowly-defined symbolic relationship (i.e. SymbolOf in Speer et al. (2017)). Therefore we conduct this analysis on the advertisement set only. As shown in Tab. 2, we specifically study the three most frequent types (i.e., UserFor, HasProperty, and RelatedTo) that appear in the ad set. We combine minor types, such as Synonym, Antonym, IsA, Causes, SymbolOf, etc., into one type named Others. We classify symbolism knowledge whose type can't be clearly determined as Indirect.

4 Experiments

We first evaluate the performance of different language models for decoding the symbolism, with or without the situated context. We then conduct experiments for verifying the biased-prior hypothesis as well as measuring the effectiveness of the debiasing method. We further investigate the fine-grained performance with respect to the knowledge difficulty and the relationship types.

4.1 Setup

We compare five language models that represent different pre-training strategies, architectures and sizes: Word2Vec (Mikolov et al., 2013), BERT (Devlin et al., 2019), RoBERTa (Liu et al., 2019), GPT-2 (Radford et al., 2019) and GPT-J-6B (Wang and Komatsuzaki, 2021). As for baseline models, we consider random guessing and co-occurrence ratio.

Random Baseline: rank signified candidates by a random order (average over 10 random runs).

Co-occurrence Baseline: rank signified candidates by its co-occurrence ratio with the signifier according to BookCorpus (Zhu et al., 2015). The ratio is computed by $\frac{N(x,y)}{N(y)}$ with the same notations as defined in Sec 3.3.

Word2Vec: rank signified candidates by the cosine similarity between the signifier word vector and each signified candidate vector. For situated symbolism, the signifier word vector is replaced by the context vector that is the summation of each token vector in the localized description.[9]

BERT (336M parameters): rank signified candidates by the probability of the masked token by querying the language model with a cloze prompt (i.e. "[signifier] is a symbol of [MASK].")[10]. For decoding general symbolism, "[signifier]" is replaced by the signifier token; for decoding situated symbolism, "[signifier]" is replaced by the localized description of the signifier.[11] Notice that the majority of signifieds are tokenized as single word pieces, with only around 20% requiring multiple word pieces. For these cases, we use the stemmed piece to transform them into a single word piece.

RoBERTa (355M parameters): same as

[9] 'word2vec-google-news-300' in gensim 4.1.2

[10] Since prompt selection is not a focus of this work, we simply picked a prompt that echoes the surface text for the "SymbolOf" relation presented in Speer et al. (2017).

[11] 'bert-large-uncased' in transformers 4.8.2

	Conventional Symbolism			Advertising Symbolism					
				w/o context			w/ context		
	P@1	P@5	P@10	P@1	P@5	P@10	P@1	P@5	P@10
Random	1.29	5.15	10.45	2.48	11.43	23.83	2.12	9.77	20.30
Co-occur	7.58	18.94	35.61	16.10	42.86	**57.89**	13.96	**34.53**	46.42
Word2Vec	5.30	25.76	46.21	**18.42**	**43.23**	**57.89**	**14.53**	32.64	47.17
BERT	10.61	27.27	40.15	10.15	25.56	39.85	11.51	27.17	39.81
RoBERTa	19.70	33.33	42.42	13.16	33.08	45.86	10.00	27.55	45.47
GPT-2	6.06	16.67	26.52	4.51	17.67	30.08	7.36	19.43	37.74
GPT-J	**27.27**	**46.97**	**56.06**	10.90	28.20	42.48	13.96	33.77	**50.00**
GPT-J (open vocab)	15.15	39.39	48.48	2.63	11.28	16.92	4.91	13.02	18.68

Table 3: Model performance (P@n) for decoding symbolism.

	Color	Nature	Plants	Weat.	Anim.	Setting	Object	Action	Num.	Christ.	Direct.
RoBERTa	**50.00**	35.29	11.11	11.11	10.53	7.14	**31.82**	0.00	0.00	14.29	0.00
GPT-J	41.67	35.29	**33.33**	**33.33**	**36.84**	7.14	27.27	**33.33**	0.00	14.29	0.00

Table 4: Model performance (P@1) on each signifier group of conventional literary symbolism.

BERT.[12]

GPT-2 (124M parameters): rank signified candidates by the probability of the next token by querying the language model with the first part of the sentence (i.e. "[signifier] is a symbol of").[13]

GPT-J (6B parameters): same as GPT-2.[14]

We evaluate each model based on how highly it ranks the ground-truth signified against others in a fixed vocabulary. We also evaluate GPT-J's performance under an open-vocabulary setting. We use the precision at n ($P@n$) as the evaluative metric. To account for multiple valid signifieds for a given signifier, this value is 1 if at least one of the valid signifieds is ranked among the top n predictions, and 0 otherwise. Experiments are conducted on the GPU model of NVIDIA Quadro RTX 5000, 16G memory, driver version 460.84 and CUDA version 11.2.

4.2 Model Performance on Decoding Symbolism

We find the three classes of LMs excel under different conditions.

Newer LMs outperform their previous iterations. Tab 3 shows the overall performance for decoding symbolism through our SymbA probe. For decoding conventional symbols, GPT-J outperforms all other models by a substantial margin overall; even under the more challenging open-vocabulary setting, GPT-J still has a comparable performance with the fixed-vocabulary setting of BERT or RoBERTa. We observe a significant improvement when the same type of language model is scaled up: GPT-J performs 21 points better than GPT-2; RoBERTa performs 9 points better than BERT in $P@1$. Surprisingly, Word2Vec and GPT-2 perform worse than the Co-occur baseline and only around 5 points better than a random guess. By looking to $P@n$ with varying n, BERT and RoBERTa are more accurate at top 1 or 5 predictions than Word2Vec, while Word2Vec has a better convergence when n is equal to 10.

Variations in signifiers' types impact decoding. Tab 4 compares RoBERTa and GPT-J's performances by signifier types. Both excel at decoding *Colors*, but they falter on *Numbers* and *Directions*. GPT-J outperforms RoBERTa on average, but it has slightly lower accuracy for *Colors* and *Objects*. We conjecture that the Web data used to pre-train GPT-J may be more multi-modal such that color attributes may be shown visually.

Bias is more severe when decoding ad symbols. For the advertising symbolism without context, Word2Vec has the best result, and GPT-2 has the worst. It is surprising that powerful language models such as RoBERTa perform worse than the simple Word2Vec or the Co-occur baseline on this task. We have similar observations for decoding situated ad symbolism. The main reason is that these advanced language models encounter the prior-bias problem thus their performance for decoding symbolism decreases. We provide more experimental results in the following section.

4.3 Effectiveness of Debiasing

The hypothesized bias exists, and re-ranking significantly reduces it. We first compute the corre-

[12] 'roberta-large' in transformers 4.24.0

[13] 'gpt2' in transformers 4.8.2

[14] 'EleutherAI/gpt-j-6B' in transformers 4.24.0

Model	Pearson score before	Pearson score after
BERT	0.375	-0.107
RoBERTa	0.355	-0.123
GPT-2	0.483	-0.192
GPT-J	0.363	-0.244

Table 5: Pearson correlation scores between candidates' frequency and prediction probability before or after normalized by the prior probability.

	Conventional	Advertising	
		w/o context	w/ context
BERT$\rightarrow_R$	10.61 → 12.88	10.15 → 17.29	11.51 → 22.08
RoBERTa$\rightarrow_R$	19.70 → 20.45	**13.16** → **25.19**	10.00 → **26.04**
GPT-2$\rightarrow_R$	6.06 → 7.58	4.51 → 9.77	7.36 → 19.43
GPT-J$\rightarrow_R$	**27.27** → **28.03**	10.90 → 22.18	**13.96** → 22.82

Table 6: Measuring the effectiveness (P@1) of the reranking approach for decoding symbolism (original → re-ranked).

lation between each signified's (y_i) frequency and its predicted probability, $p(y_i|x, t, \theta)$ for verifying the biased-prior hypothesis introduced in Sec 3.2. We use BookCorpus as the source for estimating y_i's frequency and use the advertising symbolism as testing samples. The Pearson correlation scores are reported in Tab 5. The original Pearson scores before normalizing the prior probability are always above 0.3. These results reveal that the correlation level between these two factors is positively moderate (Cohen, 2013). Our hypothesis is thus verified. Then we demonstrate that our proposed re-ranking approach mitigates this bias. By considering the prior probability of y_i, we compute the Pearson correlation score between y_i's frequency and $\frac{p(y_i|x,t,\theta)}{p(y_i|t,\theta)}$. The scores all decrease to a low level, from -0.107 to -0.244, which can be interpreted as no or slight correlation (Cohen, 2013). However, even though the absolute correlation score decreases, there exists a shift from a positive to a negative correlation level, which implies that this bias has been overcorrected.

Debiased LMs rival human performances in some cases. As shown in Tab 6, language models after re-ranking have better performance on decoding symbolism than the original ones. In particular, the improvement for larger models such as RoBERTa is more than 200% on decoding ad symbolism. The re-ranking approach boosts RoBERTa to a relatively high accuracy, 25.19 (or 26.04) for decoding ad symbolism without (or with) the situated context. We further compare models' performance with humans under a simplified 4-choice task. As shown in Tab 7, we find that GPT-J after re-ranking can impressively understand conventional symbolism even better than humans.[15] For ad symbols, RoBERTa after re-ranking achieves performance close to humans, with only 4 points behind.

	Conventional	Advertising	
		w/o context	w/ context
Human	77.27	71.43	68.00
RoBERTa$\rightarrow_R$	68.18 → 77.27	35.71 → 67.86	42.00 → 64.00
GPT-J$\rightarrow_R$	72.73 → 90.91	53.57 → 64.29	50.00 → 62.00

Table 7: Accuracy on the multi-choice task: human versus LMs (original → re-ranked).

Debiased RoBERTa and GPT-J have different strengths. Tab 6 and Tab 7 show that GPT-J is better at decoding conventional symbols and RoBERTa is better at decoding advertising symbols. We conduct further analysis to explain the observations in the next section (Sec 4.4).

4.4 Fine-grained Performance with Analytical Tools

Further experiments using the two analytical tools in SymbA probe help us better understand situations in which LMs fail and how re-ranking helps.

Analysis by Knowledge Difficulties: 1) RoBERTa is better at semantically-related symbols while GPT-J is better at distantly-related ones. We first measure the difficulty distribution of both symbolism sets. The knowledge difficulty for each symbolic pair is measured by the PMI score introduced in Sec 3.3. The mean of PMI scores for the ad set and the literary set are respectively -0.997 (with ±1.56 variance) and -3.872 (with ±5.96 variance). It reveals that the symbolism samples in the ad set are much easier than in the literary one, which suggests our headline finding. In order to provide more insights, we further split the pairwise samples into several difficulty groups and report the model performance on each of them in Tab 8. The literary set contains mostly hard cases (only 5% of them have PMI > -2). The knowledge difficulty of ads symbolism is more diverse, covering both easy and hard ones. By comparing RoBERTa and GPT-J in each PMI group, we conclude consistent findings that GPT-J is generally better at harder cases and worse at easier ones. In particular, GPT-J$_R$ performs better when PMI is extremely low, which suggests that

[15] The human annotators are from a variety of cultural backgrounds; they have not received task specific training. Thus, the reported scores represent the ability of a typical person rather than the upper-bound performance of literary experts.

PMI score	-inf (75)	<-6 (76)	-6 to -5 (37)	-5 to -4 (136)	-4 to -3 (129)	-3 to -2 (56)	>-2 (27)
(Example	blue - conservatism	gold - dominion	ladder - connection	night - death	apple - sin	dove - purity	three - tripartite)
RoBERTa $\rightarrow_R$	1.33 → 1.33	5.26 → **5.26**	5.41 → 0.00	5.88 → 0.74	6.20 → **8.53**	3.57 → 8.93	3.70 → 18.52
GPT-J $\rightarrow_R$	1.33 → **4.00**	**7.89** → 2.63	5.41 → **2.70**	**7.35** → **4.41**	**6.98** → 6.98	**5.36** → **16.07**	**18.52** → **22.22**

PMI score	-inf (20)	<-2 (79)	-2 to -1 (108)	-1 to 0 (87)	0 to 1 (45)	>1 (16)
(Example	igloo - refreshing	gun - death	bird - freedom	dragon - adventure	beach - vacation	ornaments - christmas)
RoBERTa $\rightarrow_R$	5.00 → 5.00	6.33 → **5.06**	**12.04** → **10.19**	**10.34** → **18.39**	**13.33** → 48.89	6.25 → **68.75**
GPT-J $\rightarrow_R$	5.00 → **10.00**	6.33 → 1.27	10.19 → 7.41	8.05 → 17.24	8.89 → **51.11**	6.25 → 50.00

Table 8: Model performance (P@1) on the conventional literary symbolism (upper) and the advertising symbolism (lower), on different PMI scores (measure of difficulty, from high to low). Comparing RoBERTa with GPT-J, the higher P@1 is bolded. Comparing the effectiveness of the re-ranking approach (original → re-ranked), the improvement is marked in green and the drop is marked in red. We also provide an example in each PMI group for gaining more insights.

Relationship type	UsedFor		HasProperty		RelatedTo		Others	Indirect
	default	specific	default	specific	default	specific	default	default
RoBERTa	5.77	23.08	10.87	4.35	8.51	4.26	20.21	3.45
RoBERTa$_R$	21.15	21.15	15.22	17.39	19.15	14.89	37.23	4.31
GPT-J	9.62	19.23	10.87	19.57	4.26	2.13	14.89	2.59
GPT-J$_R$	21.15	23.08	17.39	26.09	17.02	10.64	28.72	3.45

Table 9: Model performance (P@1) on relationship types when using the default prompt ("is a symbol of") or a type-specific prompt (respectively "is used for", "has the property of" or "relates to" for the relationship type of "UsedFor", "HasProperty" or "RelatedTo").

Relationship Type	PMI mean ± variance
UsedFor	-0.39 ± 2.35
HasProperty	-1.02 ± 1.31
RelatedTo	-0.86 ± 0.75
Others	-0.51 ± 1.33
Indirect	-1.71 ± 0.93

Table 10: The PMI score for each relationship type.

GPT-J can better interpret very rare symbols.

2) Debiasing improves semantically-related symbolic pairs without hurting distantly-related ones. By comparing the model performance before or after re-ranking in Tab 8, we find that the re-ranking approach can make great improvement for both RoBERTa and GPT-J on decoding easy cases (up to 62% increase on P@1 for PMI $>$ 1), with little decrease on hard cases. The intuition is that the prior probability of the signified, as a denominator term for computing the PMI score, tends to be small when PMI is large (easy cases). So normalizing by this small prior probability increases the ranking of the correct signified for easy cases. Similarly, the performance on hard cases after re-ranking is expected to decrease. It is interesting to find that the impact of the re-ranking approach is significantly positive for easy cases and only slightly negative on hard cases, which brings an overall improvement. By looking into their performance in different difficulty groups, the accuracy of GPT-J$_R$ and RoBERTa$_R$ generally increases when the knowledge difficulty decreases; unexpectedly, original models have a quite stable performance, even a little worse on the easiest cases (PMI $>$ 1).

Analysis by Relationship Types: 1) Breakdown by relationship types is consistent with analysis by knowledge difficulties. We first measure the difficulty level of each relationship type introduced in Tab 2. We show the result in Tab 10. Indirect is the most difficult (because the logical reasoning between these symbolic pairs is hard to identify); and UserFor is the easiest. Model performance on each relationship type is shown in Tab 9. Consistent with what we have observed before, reranking improves more for the type of *UsedFor*, *Others* and *RelatedTo*, which are easier (PMI $>$ -1) than other types; and RoBERTa performs better than GPT-J when decoding these types of symbols.

2) Debiasing improves LMs' robustness without prompt engineering. We experiment with a type-specific prompt for each relationship type, *e.g.*, we replace the default "is a symbol of" by "is used for" when probing a symbol in the type of UsedFor. We find that the type-specific prompt can sometimes greatly facilitate the original models on decoding knowledge: RoBERTa increases 17 points for UsedFor; GPT-J increases around 9 points for UsedFor or HasProperty. At first glance, this suggests that these LMs do have knowledge about the semantic relationships between the signifier and signified, but the general prompt cannot elicit the desired response. However, we also observe that type-specific prompts have little impact for the re-ranked models, *e.g.*, RoBERTa performs same when prompted by the default or the type-specific template. While language models are sensitive to the prompt template, the re-ranking approach helps to stabilize their performance. We believe that improving debiasing methods, more so than prompt engineering, holds the key to developing robust models.

5 Conclusion

In this work, we have assessed the feasibility of eliciting symbolic knowledge from different types of language models. By evaluating LMs through the SymbA probe, we find that advanced large language models (e.g. GPT-J and RoBERTa) can achieve human-level performance on a simplified 4-choice task of identifying the intended signified concept from a given signifier. However, there is still ample room for improvement when the model is prompted to select from a large set of candidates. We have also validated that these models are biased in favor of commonly occurring signified concepts. The debiasing method based on re-ranking can significantly improve the performance and increase the robustness with respect to the probing template. Our work shows the potential of incorporating language models as a source of knowledge about symbolic relationships for real-world applications that involve understanding and interpreting non-literal expressions.

6 Limitations

Because decoding symbolism is a challenging new problem, our approach and experimental results have some limitations. First, our work builds on available resources, which may have a bias toward an English/Euro-centric perspective. Second, the evaluative datasets that we curated have a limited coverage of possible symbols even within the English literary tradition. Third, as mentioned in Section 3.1, our study on situated symbolism is limited to symbolic pairs that can be found in static visual advertisements rather than longer form text or videos. Finally, while we have proposed one debiasing method based on re-ranking with PMI, which worked well for our experimental setting, there may be other methods and metrics more suited to different settings. We believe that despite these limitations, our proposed evaluative framework and methodology offers a good starting point for further exploration.

7 Acknowledgements

This work was partially supported by National Science Foundation Grant No. 1718262, Google/Amazon/Adobe gifts, and University of Pittsburgh Computer Science CS50 fellowship.

References

Arjun R Akula, Brendan Driscoll, Pradyumna Narayana, Soravit Changpinyo, Zhiwei Jia, Suyash Damle, Garima Pruthi, Sugato Basu, Leonidas Guibas, William T Freeman, et al. 2023. Metaclue: Towards comprehensive visual metaphors research. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 23201–23211.

Peter Anderson, Xiaodong He, Chris Buehler, Damien Teney, Mark Johnson, Stephen Gould, and Lei Zhang. 2018. Bottom-up and top-down attention for image captioning and visual question answering. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 6077–6086.

Ron Artstein and Massimo Poesio. 2008. Survey article: Inter-coder agreement for computational linguistics. *Computational Linguistics*, 34(4):555–596.

Douglas Brown. 1997. The penguin dictionary of symbols. *Reference Reviews*.

Tom Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared D 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 Ziegler, Jeffrey Wu, Clemens Winter, Chris 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. In *Advances in Neural Information Processing Systems*, volume 33, pages 1877–1901. Curran Associates, Inc.

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.

Tuhin Chakrabarty, Xurui Zhang, Smaranda Muresan, and Nanyun Peng. 2021. MERMAID: Metaphor generation with symbolism and discriminative decoding. In *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 4250–4261, Online. Association for Computational Linguistics.

Jacob Cohen. 2013. *Statistical power analysis for the behavioral sciences*. Academic Press.

Joe Davison, Joshua Feldman, and Alexander Rush. 2019. Commonsense knowledge mining from pretrained models. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 1173–1178, Hong Kong, China. 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, Minneapolis, Minnesota. Association for Computational Linguistics.

Meiqi Guo, Rebecca Hwa, and Adriana Kovashka. 2021. Detecting persuasive atypicality by modeling contextual compatibility. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 972–982.

Meiqi Guo, Rebecca Hwa, Yu-Ru Lin, and Wen-Ting Chung. 2020. Inflating topic relevance with ideology: A case study of political ideology bias in social topic detection models. In *Proceedings of the 28th International Conference on Computational Linguistics*, pages 4873–4885, Barcelona, Spain (Online). International Committee on Computational Linguistics.

Edward L Hancock. 1972. *Techniques for Understanding Literature: A Handbook for Readers and Writers*. Wadsworth Publishing Company.

Lisa Hartling, Michele Hamm, Andrea Milne, Ben Vandermeer, P Lina Santaguida, Mohammed Ansari, Alexander Tsertsvadze, Susanne Hempel, Paul Shekelle, and Donna M Dryden. 2012. Validity and inter-rater reliability testing of quality assessment instruments.

Ari Holtzman, Peter West, Vered Shwartz, Yejin Choi, and Luke Zettlemoyer. 2021. Surface form competition: Why the highest probability answer isn't always right. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pages 7038–7051, Online and Punta Cana, Dominican Republic. Association for Computational Linguistics.

Zaeem Hussain, Mingda Zhang, Xiaozhong Zhang, Keren Ye, Christopher Thomas, Zuha Agha, Nathan Ong, and Adriana Kovashka. 2017. Automatic understanding of image and video advertisements. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 1705–1715.

Zhengbao Jiang, Frank F. Xu, Jun Araki, and Graham Neubig. 2020. How can we know what language models know? *Transactions of the Association for Computational Linguistics*, 8:423–438.

Ernest Jones. 1918. The theory of symbolism. *British Journal of Psychology*, 9(2):181.

Walter Kintsch. 2000. Metaphor comprehension: A computational theory. *Psychonomic bulletin & review*, 7(2):257–266.

Klaus Krippendorff. 2011. Computing krippendorff's alpha-reliability.

Murathan Kurfalı and Robert Östling. 2020. Disambiguation of potentially idiomatic expressions with contextual embeddings. In *Proceedings of the Joint Workshop on Multiword Expressions and Electronic Lexicons*, pages 85–94, online. Association for Computational Linguistics.

J Richard Landis and Gary G Koch. 1977. The measurement of observer agreement for categorical data. *biometrics*, pages 159–174.

Ron Langacker. 1996. Cognitive linguistics symposium. In *Proceedings of the Eighteenth Annual Conference of the Cognitive Science Society: July 12-15, 1996, University of California, San Diego*, volume 18, page 15. Psychology Press.

Hongsong Li, Kenny Q. Zhu, and Haixun Wang. 2013. Data-driven metaphor recognition and explanation. *Transactions of the Association for Computational Linguistics*, 1:379–390.

Yinhan Liu, Myle Ott, Naman Goyal, Jingfei Du, Mandar Joshi, Danqi Chen, Omer Levy, Mike Lewis, Luke Zettlemoyer, and Veselin Stoyanov. 2019. Roberta: A robustly optimized bert pretraining approach. *arXiv preprint arXiv:1907.11692*.

Zhexiong Liu, Meiqi Guo, Yue Dai, and Diane Litman. 2022. ImageArg: A multi-modal tweet dataset for image persuasiveness mining. In *Proceedings of the 9th Workshop on Argument Mining*, pages 1–18, Online and in Gyeongju, Republic of Korea. International Conference on Computational Linguistics.

Tomás Mikolov, Kai Chen, Greg Corrado, and Jeffrey Dean. 2013. Efficient estimation of word representations in vector space. In *1st International Conference on Learning Representations, ICLR 2013, Scottsdale, Arizona, USA, May 2-4, 2013, Workshop Track Proceedings*.

Arthur Neidlein, Philip Wiesenbach, and Katja Markert. 2020. An analysis of language models for metaphor recognition. In *Proceedings of the 28th International Conference on Computational Linguistics*, pages 3722–3736, Barcelona, Spain (Online). International Committee on Computational Linguistics.

Fabio Petroni, Tim Rocktäschel, Sebastian Riedel, Patrick Lewis, Anton Bakhtin, Yuxiang Wu, and

Alexander Miller. 2019. Language models as knowledge bases? In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 2463–2473, Hong Kong, China. Association for Computational Linguistics.

Alec Radford, Karthik Narasimhan, Tim Salimans, and Ilya Sutskever. 2018. Improving language understanding by generative pre-training.

Alec Radford, Jeffrey Wu, Rewon Child, David Luan, Dario Amodei, Ilya Sutskever, et al. 2019. Language models are unsupervised multitask learners.

Zachary Rosen. 2018. Computationally constructed concepts: A machine learning approach to metaphor interpretation using usage-based construction grammatical cues. In *Proceedings of the Workshop on Figurative Language Processing*, pages 102–109, New Orleans, Louisiana. Association for Computational Linguistics.

Ekaterina Shutova. 2010. Automatic metaphor interpretation as a paraphrasing task. In *Human Language Technologies: The 2010 Annual Conference of the North American Chapter of the Association for Computational Linguistics*, pages 1029–1037, Los Angeles, California. Association for Computational Linguistics.

Ekaterina Shutova, Douwe Kiela, and Jean Maillard. 2016. Black holes and white rabbits: Metaphor identification with visual features. In *Proceedings of the 2016 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 160–170, San Diego, California. Association for Computational Linguistics.

Vered Shwartz and Yejin Choi. 2020. Do neural language models overcome reporting bias? In *Proceedings of the 28th International Conference on Computational Linguistics*, pages 6863–6870, Barcelona, Spain (Online). International Committee on Computational Linguistics.

Robyn Speer, Joshua Chin, and Catherine Havasi. 2017. Conceptnet 5.5: An open multilingual graph of general knowledge. In *Thirty-first AAAI conference on artificial intelligence*.

Arthur Symons. 2014. *The symbolist movement in literature*. Carcanet.

Tony Veale and Yanfen Hao. 2008. A fluid knowledge representation for understanding and generating creative metaphors. In *Proceedings of the 22nd International Conference on Computational Linguistics (Coling 2008)*, pages 945–952, Manchester, UK. Coling 2008 Organizing Committee.

Alex Wang and Kyunghyun Cho. 2019. BERT has a mouth, and it must speak: BERT as a Markov random field language model. In *Proceedings of the Workshop on Methods for Optimizing and Evaluating Neural Language Generation*, pages 30–36, Minneapolis, Minnesota. Association for Computational Linguistics.

Ben Wang and Aran Komatsuzaki. 2021. GPT-J-6B: A 6 Billion Parameter Autoregressive Language Model. `https://github.com/kingoflolz/mesh-transformer-jax`.

Judith Williamson. 1978. *Decoding advertisements: ideology and meaning in advertising*. Marion Boyers.

Lili Yao, Yaoyuan Zhang, Yansong Feng, Dongyan Zhao, and Rui Yan. 2017. Towards implicit content-introducing for generative short-text conversation systems. In *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing*, pages 2190–2199, Copenhagen, Denmark. Association for Computational Linguistics.

Kun Zhou, Kai Zhang, Yu Wu, Shujie Liu, and Jingsong Yu. 2019. Unsupervised context rewriting for open domain conversation. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 1834–1844, Hong Kong, China. Association for Computational Linguistics.

Yukun Zhu, Ryan Kiros, Rich Zemel, Ruslan Salakhutdinov, Raquel Urtasun, Antonio Torralba, and Sanja Fidler. 2015. Aligning books and movies: Towards story-like visual explanations by watching movies and reading books. In *The IEEE International Conference on Computer Vision (ICCV)*.

A Instructions for Annotators

*Please describe the object which is in the red box.

*The description should be 1) in a short noun phrase, i.e. maximum 8 words (e.g. tooth under an umbrella); 2) capable to tell its symbolic meaning that is already given (e.g. blood signifies danger; lemon signifies refreshing; tooth under an umbrella signifies protection and heath).

*Instruction for corner cases:

1) If there are multiple objects in the red box, please first identify several objects which relate to the given symbolic meaning, then describe them and their relationship in a short phrase, e.g. tooth under an umbrella.

2) If some attributes of the target object is essential for telling its symbolic meaning, please describe the attribute (e.g. color, shape, status, action) with the class name together, e.g. bleeding arm

*In summary, the goal is to infer the given symbolic meaning from your written description. If you meet some cases which are not covered by the instruction, please write a description which helps most for inferring the given symbolic meaning.

*Some examples of expected annotations are shown on the first page of this form: [link]

ACL 2023 Responsible NLP Checklist

A For every submission:

☑ A1. Did you describe the limitations of your work?
sec 6

☒ A2. Did you discuss any potential risks of your work?
No user; no ethic concern

☑ A3. Do the abstract and introduction summarize the paper's main claims?
Left blank.

☒ A4. Have you used AI writing assistants when working on this paper?
Left blank.

B ☑ Did you use or create scientific artifacts?

sec 3.1

☑ B1. Did you cite the creators of artifacts you used?
sec 3.1

☒ B2. Did you discuss the license or terms for use and / or distribution of any artifacts?
They are published and publicly available

☑ B3. Did you discuss if your use of existing artifact(s) was consistent with their intended use, provided that it was specified? For the artifacts you create, do you specify intended use and whether that is compatible with the original access conditions (in particular, derivatives of data accessed for research purposes should not be used outside of research contexts)?
sec 3.1

☑ B4. Did you discuss the steps taken to check whether the data that was collected / used contains any information that names or uniquely identifies individual people or offensive content, and the steps taken to protect / anonymize it?
sec 3.1

☑ B5. Did you provide documentation of the artifacts, e.g., coverage of domains, languages, and linguistic phenomena, demographic groups represented, etc.?
sec 3.1

☑ B6. Did you report relevant statistics like the number of examples, details of train / test / dev splits, etc. for the data that you used / created? Even for commonly-used benchmark datasets, include the number of examples in train / validation / test splits, as these provide necessary context for a reader to understand experimental results. For example, small differences in accuracy on large test sets may be significant, while on small test sets they may not be.
sec 3.1

C ☑ Did you run computational experiments?

Left blank.

☑ C1. Did you report the number of parameters in the models used, the total computational budget (e.g., GPU hours), and computing infrastructure used?
sec 4.1

The Responsible NLP Checklist used at ACL 2023 is adopted from NAACL 2022, with the addition of a question on AI writing assistance.

☒ C2. Did you discuss the experimental setup, including hyperparameter search and best-found hyperparameter values?
We didn't train the mode. We evaluated models.

☑ C3. Did you report descriptive statistics about your results (e.g., error bars around results, summary statistics from sets of experiments), and is it transparent whether you are reporting the max, mean, etc. or just a single run?
sec 4

☑ C4. If you used existing packages (e.g., for preprocessing, for normalization, or for evaluation), did you report the implementation, model, and parameter settings used (e.g., NLTK, Spacy, ROUGE, etc.)?
sec 4.1

D ☑ Did you use human annotators (e.g., crowdworkers) or research with human participants?
sec 3.1

☑ D1. Did you report the full text of instructions given to participants, including e.g., screenshots, disclaimers of any risks to participants or annotators, etc.?
Appendice

☒ D2. Did you report information about how you recruited (e.g., crowdsourcing platform, students) and paid participants, and discuss if such payment is adequate given the participants' demographic (e.g., country of residence)?
No, because our human annotation only has 530 samples and our annotators are volunteer PhD students and faculties.

□ D3. Did you discuss whether and how consent was obtained from people whose data you're using/curating? For example, if you collected data via crowdsourcing, did your instructions to crowdworkers explain how the data would be used?
Not applicable. Left blank.

□ D4. Was the data collection protocol approved (or determined exempt) by an ethics review board?
Not applicable. Left blank.

☒ D5. Did you report the basic demographic and geographic characteristics of the annotator population that is the source of the data?
Because it was not a large annotation dataset and we only have 11 annotators. It is part of our evaluation probe but not the major contribution of our work.