UNIVERSITY OF AMSTERDAM

UvA-DARE (Digital Academic Repository)

Do Pre-Trained Language Models Detect and Understand Semantic Underspecification? Ask the DUST!

Wildenburg, F.; Hanna, M.; Pezzelle, S.

DOI
10.48550/arXiv.2402.12486
10.18653/v1/2024.findings-acl.572

Publication date
2024

Document Version
Final published version

Published in
The 62nd Annual Meeting of the Association for Computational Linguistics : Findings of the Association for Computational Linguistics: ACL 2024

License
CC BY

Link to publication

Citation for published version (APA):
Wildenburg, F., Hanna, M., & Pezzelle, S. (2024). Do Pre-Trained Language Models Detect and Understand Semantic Underspecification? *Ask the DUST!*. In L.-W. Ku, A. Martins, & V. Srikumar (Eds.), *The 62nd Annual Meeting of the Association for Computational Linguistics : Findings of the Association for Computational Linguistics: ACL 2024: ACL 2024 : August 11-16, 2024* (pp. 9598-9613). Association for Computational Linguistics. https://doi.org/10.48550/arXiv.2402.12486, https://doi.org/10.18653/v1/2024.findings-acl.572

General rights
It is not permitted to download or to forward/distribute the text or part of it without the consent of the author(s) and/or copyright holder(s), other than for strictly personal, individual use, unless the work is under an open content license (like Creative Commons).

Disclaimer/Complaints regulations
If you believe that digital publication of certain material infringes any of your rights or (privacy) interests, please let the Library know, stating your reasons. In case of a legitimate complaint, the Library will make the material inaccessible and/or remove it from the website. Please Ask the Library: https://uba.uva.nl/en/contact, or a letter to: Library of the University of Amsterdam, Secretariat, P.O. Box 19185, 1000 GD Amsterdam, The Netherlands. You will be contacted as soon as possible.

UvA-DARE is a service provided by the library of the University of Amsterdam (https://dare.uva.nl)

Download date:14 Oct 2026

Do Pre-Trained Language Models Detect and Understand Semantic Underspecification? *Ask the DUST!*

Frank Wildenburg
College of Informatics
University of Amsterdam
f.c.l.wildenburg@uva.nl

Michael Hanna
ILLC
University of Amsterdam
m.w.hanna@uva.nl

Sandro Pezzelle
ILLC
University of Amsterdam
s.pezzelle@uva.nl

Abstract

In everyday language use, speakers frequently utter and interpret sentences that are *semantically underspecified*, namely, whose content is insufficient to fully convey their message or interpret them univocally. For example, to interpret the underspecified sentence "Don't spend too much", which leaves implicit what (not) to spend, additional linguistic context or outside knowledge is needed. In this work, we propose a novel **D**ataset of semantically **U**nderspecified **S**entences grouped by **T**ype (**DUST**) and use it to study whether pre-trained language models (LMs) correctly identify and interpret underspecified sentences. We find that newer LMs are reasonably able to identify underspecified sentences when explicitly prompted. However, interpreting them correctly is much harder for any LMs. Our experiments show that when interpreting underspecified sentences, LMs exhibit little uncertainty, contrary to what theoretical accounts of underspecification would predict. Overall, our study reveals limitations in current models' processing of sentence semantics and highlights the importance of using naturalistic data and communicative scenarios when evaluating LMs' language capabilities.

1 Introduction

Speakers can almost effortlessly deal with *semantic underspecification*, a widespread phenomenon that occurs when a linguistic signal does not fully convey all the information required for communication to succeed (Frisson, 2009; Harris, 2020). This is possible because, in a normal state of affairs, humans have access to further linguistic or extra-linguistic information coming from the conversation, the surrounding context, or shared knowledge. If this is the case, speakers will have no trouble understanding the sentences "I'll meet you there" or "I saw you on the hill with the telescope", even though these sentences leave underspecified where "there" is, or which interlocutor had "a telescope". It has been argued that underspecification and the related phenomenon of ambiguity are not a hindrance in language, but rather a desirable feature of human language communication (Piantadosi et al., 2012); they allow for more concise utterances, which makes language more efficient. Indeed, humans excel at making inferences (Grice, 1969), which is less cognitively taxing compared to articulating speech (Levinson, 2000).

S1	Don't spend too much.	
S2	Don't spend too much cash.	
Exp. 1	*Is **S2** more specified than **S1**?*	✓
S1	The bag is on the chair. It is green	
S2	The bag is on the chair. The chair is green.	
Exp. 2	*Does **S1** mean the chair is green?*	?
	*Does **S2** mean the chair is green?*	✓
	*Does **S1** mean the chair isn't green?*	?
	*Does **S2** mean the chair isn't green?*	✗

Figure 1: In experiment 1 (top), we test if LMs distinguish underspecified sentences (**S1**) from more specified counterparts (**S2**) when explicitly prompted. In experiment 2 (bottom), we test if LMs correctly interpret **S1** and **S2** in a more naturalistic communicative setting.

While humans are good at dealing with semantically underspecified language by leveraging additional information, how pre-trained transformer language models (LMs) behave when faced with this phenomenon is an open question. Despite the growing literature exploring the semantic capabilities of last-generation LMs (Ettinger, 2020; Rogers et al., 2020; Hanna et al., 2023), little attention has been paid to this problem. Furthermore, the few previous works generally did not distinguish between the various facets of ambiguity and underspecification (Liu et al., 2023) or only focused on ambiguity (Stengel-Eskin et al., 2024; Ortega-Martín et al., 2023; Fantechi et al., 2023).

Findings of the Association for Computational Linguistics: ACL 2024, pages 9598–9613
August 11-16, 2024 ©2024 Association for Computational Linguistics

However, handling semantically underspecified language is critical for LMs. Since underspecified sentences can license multiple interpretations, choosing one arbitrarily can lead to undesired or even harmful consequences for communication (see Hutchinson et al., 2022; Pezzelle, 2023, for an in-depth discussion of this issue). Correctly processing underspecified language could have real-world consequences for NLP systems. For example, if an embodied virtual assistant were to misinterpret a question like "Can I hang the painting with the cup?", there could be risks: while the answer may be *yes* if the cup is the subject of the painting, the system should answer *no* if this is not the case. In machine translation, models should carefully handle underspecification; for example, when translating "they", models must determine whether the pronoun refers to a group of people or an individual of unknown, nonbinary, or intentionally underspecified gender. Therefore, LMs should (1) recognize semantically underspecified inputs and (2) interpret them appropriately, ideally with no biases toward a default reading.

We build on and extend previous work by asking two new questions: (1) to what extent can LMs detect if a sentence is (under)specified? (2) How do LMs interpret such underspecified sentences compared to more specified counterparts? To this end, we introduce **DUST**[1], a **D**ataset of semantically **U**nderspecified **S**entences (and more specified counterparts) grouped by the **T**ype of underspecification they belong to, and propose a suite of experiments to answer the questions above using DUST. To categorize instances of underspecification, we build on Egg's (2010) proposed taxonomy.

Our experiments and analysis show that (1) distinguishing semantically underspecified sentences from specified counterparts is not a trivial task for current LMs. While newer, better-performing models achieve reasonable performance when given explicit instructions, other models fare only slightly better than random. Moreover, (2) when asked to interpret underspecified sentences in a more naturalistic communicative scenario without explicit guidance, all LMs fall into the trap of interpreting them similarly to their more specified versions. This suggests that, in the absence of a specific prompt, these models assign biased or default interpretations to underspecified and ambiguous sentences.

[1]We release DUST and code for our experiments here: https://github.com/frank-wildenburg/DUST

Our findings confirm that current LMs, including the best-performing Llama 2 (Touvron et al., 2023) and Mistral (Jiang et al., 2023), struggle with underspecified and ambiguous language (in line with Liu et al., 2023). This reveals more general limitations with the processing of sentence semantics. Moreover, our study highlights the importance of methodological choices, such as experimental setting, or the level of informativeness of prompts, in fine-grained evaluations of LMs' capabilities.

2 Related work

2.1 Semantic Underspecification

Semantic underspecification is a phenomenon in which the semantic material of a sentence "leaves open" possibilities for readers of the text that may then be "filled in" through non-linguistic information (Zwicky and Sadock, 1975; Frisson, 2009; Egg, 2010; Harris, 2020). The phenomenon has traditionally been studied through the lens of (formal) linguistics and semantics (Zwicky and Sadock, 1975; Lappin, 2000; Egg, 2010; Harris, 2020, *inter alia*), although it has also been studied in other fields, such as the neuroscience of language processing (Frisson, 2009) and information theory (Piantadosi et al., 2012; Franzon and Zanini, 2022).

Semantic underspecification is often studied in tandem or in contrast with ambiguity, a related phenomenon. The difference between the two is that underspecified sentences sometimes have only one reading (which may be clarified by non-linguistic information), whereas ambiguous sentences have multiple readings (Zwicky and Sadock, 1975). However, all ambiguous sentences are also underspecified; after all, some non-linguistic information will disambiguate between readings. Hence, underspecification can be seen as a generalization of ambiguity. Despite this, the terms are sometimes used interchangeably or in tandem; Sennet (2023) points out that "often simple underspecificity will suffice for a charge of ambiguity".

Egg (2010) gives a detailed categorization of underspecification, grouping instances into four types based on whether the instance's constituent parts comprise the same semantic value across its readings, and whether it is possible to give a single syntactic analysis for all the readings. As this classification allows us to explore the interaction between the semantic and syntactic dimensions of underspecification and their effects on LMs, we use it as the theoretical backbone to build our dataset.

2.2 Semantic Underspecification in NLP

NLP has long studied problems around semantic underspecification. Early work explicitly modeled underspecification by creating formal, symbolic representations of underspecified sentences that captured each sentence's potential meanings, without generating them (Poesio, 1994; Niehren et al., 1997; Pinkal, 1999, *inter alia*). NLP systems could then use such representations to make processing sentences more tractable, despite their potentially numerous interpretations (Wahlster, 2000).

Another line of work focuses instead on identifying or resolving underspecification. Stengel-Eskin et al. (2024), for example, identify the meanings of ambiguous sentences by training a model to map from such sentences to formal representations of their multiple potential meanings. Berzak et al. (2015) train a model to resolve underspecification and determine the correct reading of an ambiguous caption, given an accompanying image. More generally, many classic NLP tasks, such as word sense disambiguation and coreference resolution, involve resolving a word's meaning in a context where it is underspecified. Where underspecification cannot be resolved, studies have tried to identify or generate clarifications or clarification questions (Roth et al., 2022; Testoni and Fernández, 2024).

Some recent work has addressed ambiguity and underspecification in the context of pre-trained LMs. Considering multi-modal models, Prasad et al. (2023) find that vision-language architectures often struggle with underspecified inputs; specifying inputs improves performance. Pezzelle (2023) reports similarly negative results, discovering that CLIP (Radford et al., 2021) sometimes prefers invalid but highly specified captions to valid but underspecified ones. Multi-modal models' challenges with underspecification concern not only performance but also ethics: Hutchinson et al. (2022) warn that image generation models might rely on social biases to fill in underspecified details.

Related questions have been studied in a unimodal, text-only context as well, though existing work focuses on ambiguity, rather than underspecification more broadly. For example, Ortega-Martín et al. (2023) and Fantechi et al. (2023) study ChatGPT's ability to explicitly identify ambiguity, and report mixed results. Most pertinently, Liu et al. (2023) study how pre-trained LMs process ambiguous sentences, and find that they are unable to use context to infer which potential reading of an ambiguous sentence is correct. In the present work, we aim to expand the existing literature to cover not just ambiguity, but all types of underspecification.

T	Phenomenon	#S	Source
1	Logical Form	35	LAVA
	Ellipsis	18	LAVA
2	PP attachment amb.	48	LAVA
	VP attachment amb.	60	LAVA
	Conjunction amb.	40	LAVA
3	Referential amb.	36	LAVA
	Referential amb.	273	WSC
	Added compound	774	CLAIRE
	Fused head	532	CLAIRE
	Implicit reference	216	CLAIRE
	Metonymic reference	91	CLAIRE

Table 1: Number of sentences (#S) and source per type (T) and phenomenon of underspecification in DUST. Phenomena containing underspecification without ambiguity are *italicized*. 'amb.' stands for ambiguity.

3 Dataset

To study how LMs deal with semantically underspecified language, and since there currently exists no comprehensive resource on underspecification, we construct DUST, a **D**ataset of **U**nderspecified **S**entences by **T**ype, consisting of 2,123 English underspecified sentences and equally many specified counterparts, based on Egg's (2010) categorization; see Table 1 for an overview. Below, we discuss the construction of the dataset by type; note that although Egg's taxonomy includes 4 types of underspecification, we only include 3 in our dataset due to the features of one type (more details below).

Type 1 Egg (2010)'s first type consists of *semantically and syntactically homogeneous expressions*, i.e., sentences with multiple readings that all share the same syntactic structure, and word/token-level meaning. To cover this type of underspecification, we collect 53 sentences from the Language and Vision Ambiguities (LAVA) dataset (Berzak et al., 2015), a multimodal dataset containing ambiguous sentences and visual data that disambiguated them. For each sentence in this dataset, we created a more specified counterpart using its visual disambiguations. An example of such a sentence is

Andrei approached Danny; Yevgeni, too.

which leaves underspecified whether Yevgeni approached Danny, or was approached by Andrei. A

potential specified version of this sentence is

> Andrei approached Danny; Andrei approached Yevgeni, too.

Type 2 The second type consists of *semantically but not syntactically homogeneous expressions*. The multiple readings of such expressions stem from the multiple ways of analyzing the expression's structure; different analyses can lead to different meanings. The LAVA dataset (Berzak et al., 2015) provides 108 sentences containing VP and PP attachment ambiguity and conjunction ambiguity, for example

> Andrei looked at the green bag with a telescope.

which leaves ambiguous whether 'with a telescope' attaches to 'the green bag' or to 'looked at'. It is thus unclear whether the bag contains a telescope, or was looked at using one. A more specified counterpart of this sentence might be

> Andrei looked at the green bag through a telescope.

Type 3 The third type consists of *syntactically but not semantically homogeneous expressions*, which share a single syntactic structure but do not share the same semantic material in its constituent parts. It is unique in that, besides instances of referential ambiguity, all examples of type 3 in DUST are underspecified but not ambiguous. It is thus particularly interesting for our work, as underspecified but not ambiguous expressions are important for LMs to handle, but currently understudied. Examples include deictic expressions and expressions that are underspecified due to missing information.

As examples of this type of underspecification, we first collect 89 sentences from LAVA that contain referential ambiguity or missing information. We then extend our collection with sentences from the CLArifying Insertions from REvision Edits (CLAIRE) dataset (Roth et al., 2022), which consists of wikiHow instructional texts, and revisions that clarify the original sentences by inserting additional information. We treat the pre-edit text as underspecified due to missing information, and the post-edit text as more specified. Due to the original authors' pre-processing, some pre-edit sentences are ungrammatical; we thus score pre-edit sentences' grammaticality with GRUEN (Zhu and Bhat, 2020) and exclude low-scoring sentences.

We also include the original 273 sentences from the Winograd Schema Challenge (WSC; Levesque et al., 2012) in our dataset as examples of referential ambiguity. For each sentence from this dataset, we crafted a more specified counterpart by changing the gender or plurality of one of the antecedents in the sentence, removing the referential ambiguity. An example sentence of this type is

> Don't spend too much.

which does not specify what should not be spent. A more specified counterpart might be

> Don't spend too much cash.

Type 4 Egg's fourth and final type, consisting of expressions that are *neither syntactically nor semantically homogeneous*, concerns phenomena that occur at the word level, such as homonymic expressions. While a word (e.g. "plant") can have multiple syntactically and semantically distinct readings, most sentences that contain homonyms have readings that rely on the same syntactic structure. For example, "he walked to the bank" contains the homonym "bank", which could refer to a riverbank of a financial institution, but the syntactic structure of the sentence is the same across both readings.

Unlike the three types described above, the phenomena belonging to this type cannot be easily studied using an experimental setup based on minimal pairs—the one used in this work—where the two interpretations are embedded in two phrases or sentences that only differ by a single, minimal intervention (see Appendix A for a preliminary exploration of the problem). For this reason, we do not include type 4 in our benchmark and leave a comprehensive exploration of it to future work.

4 Models

We focus on pre-trained autoregressive models, including both older and newer (generally stronger) LMs to consider the influence of general performance improvements on LMs' underspecification processing. As our experiments require sentence probabilities, we use only openly available models, which provide these. Specifically, we consider the following models, accessed using HuggingFace's Transformers library (Wolf et al., 2020):[2]

- GPT-2 XL (Radford et al., 2019), a 1.5 billion parameter decoder-only LM trained on a 40GB dataset of webpages.

[2]See Appendix C for more model details.

- FLAN-T5 XXL (Chung et al., 2022), a 11 billion parameter encoder-decoder LM. FLAN-T5 is an enhanced version of T5, finetuned in a mixture of language modeling tasks.
- OPT-13b (Zhang et al., 2022), a 13 billion parameter decoder-only transformer model.
- Llama 2 7b and 13b (Touvron et al., 2023), 7 and 13 billion parameter[3] models trained on publicly available online data.
- Mistral 7b v0.1 (Jiang et al., 2023), a 7.3 billion parameter model designed to provide a balance between performance and efficiency.

Except for Flan-T5, the models we use in our research were not instruction-tuned during pretraining. This allows us to test the abilities of base LMs to process underspecification.

5 Detecting Semantic Underspecification

We test whether LMs recognize that a sentence is more or less specified by looking at the perplexity it assigns to a prompt comparing the degree of (under)specification of two embedded sentences. This task seeks to replicate the process of assessing human speakers' understanding of this phenomenon by asking them to provide a metalinguistic assessment. We use a perplexity-based approach, rather than evaluating the generative behavior of the models in response to a prompt, as there is growing evidence that prompt-based approaches are not suitable for this purpose (Hu and Levy, 2023). For comparison, we include a preliminary experiment exploring model-generated responses in Appendix D.

5.1 Experimental Setup

We create inputs of the form "This is an underspecified sentence: `[sentence1]`. This is its more specified counterpart: `[sentence2]`", where `[sentence1]` and `[sentence2]` are a pair of sentences from DUST. For each pair, we create a version of the input where the sentences are correctly labeled as under- and more specified and one where their labels are switched.

If the models can recognize underspecification, we would expect that the inputs where the specification labels are correct would receive a higher probability / lower perplexity than the same input with incorrect labels. That is, the input

> This is an underspecified sentence: 'Andrei left the chair with a green telescope'. This is its more specified counterpart: 'Andrei left the chair on which lay a green telescope'.

should be judged by LMs as more likely than the input where the blue (underspecified) and red (more specified) sentences are switched. We test this by computing, for a given prompt, the product of the model perplexities assigned to each token in it.

To ensure prompt diversity, and because models may not have been exposed to terminology such as "underspecified" during training, we also use alternate versions of our prompts, where "under-" and "more specified" are replaced by "(un)ambiguous', or "contain (little/a lot of) (information/detail)". We also create prompts that reverse the order in which the under- and more specified sentences are presented.[4] In total, we create 33,968 input pairs: 2,123 sentence pairs × 4 prompt variants × 4 orders. Then, for each model and input pair, we record whether the model correctly assigns a lower perplexity to the specification-matched input than to its mismatched counterpart.

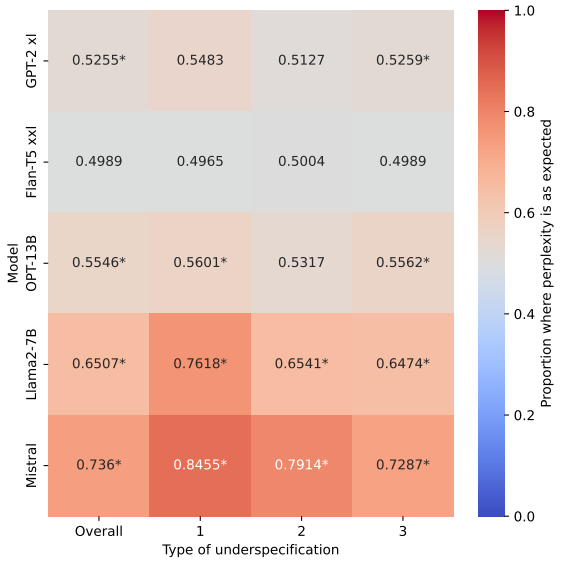

Figure 2: Proportion of specification-matched inputs that receive lower perplexity than their specification-mismatched counterparts; higher is better. Asterisks (*) indicate performance significantly ($p < .05$) different from chance (50%). The leftmost column, *Overall*, reports the proportion computed over the whole dataset.

Sanity check Our experimental setup aims to determine if models can recognize underspecifica-

[3]Due to compute limits, we only use the 13 billion parameter variant of Llama 2 in our second experiment

[4]For more example inputs, see Appendix E.

tion when prompted to do so; however, prior work suggests that models may struggle to understand their prompts (Webson and Pavlick, 2022). So, we first verify the soundness of our setup by using it to test models in an easier domain: sentiment. We gather "very positive" and "very negative" sentences from the SST-5 dataset (Socher et al., 2013), and insert them into prompts of the form "This is a positive sentence: `[sentence1]`. This is a negative sentence: `[sentence2]`". All models assign lower perplexity to sentiment-matched inputs with at least 65% and an average of 75% accuracy (see Appendix G for details). This indicates that current LMs can be tested using this experimental setup.

5.2 Results

Newer models perform better Our results (Figure 2) indicate that some LMs can recognize underspecification. All models besides Flan-T5 do so at a rate significantly higher than chance; Flan-T5's poor performance may be due to its architecture, i.e., a fine-tuned encoder-decoder model and not a decoder-only model trained on causal language modeling like the others. Stronger models more often prefer specification-matched prompts: Mistral, which performs significantly better than all other models across the board ($p < .001$), achieves an overall accuracy of 0.74. The second-best model, Llama 2, lags behind Mistral by almost 10 accuracy points, with an overall accuracy of 0.65. In turn, this LM significantly ($p < 0.001$) outperforms OPT (0.55) and GPT-2 (0.53) by as many accuracy points, indicating a clear ranking between the various models.

Given the models' high performance in the control experiment with sentiment, the generally lower accuracy observed here is likely due to models' difficulties in recognizing and identifying underspecification, rather than prompt-related challenges.

Results vary across types Even for top models, performance across different types of underspecification is not uniform, with gaps of up to 12 accuracy points between the best- and worst-performing types. Mistral, for example, achieves a peak in performance on type 1 (0.85), followed by type 2 (0.79), and type 3 (0.73). For the second best-performing model, Llama 2, type 1 is also the easiest (0.76). However, in contrast with Mistral, types 2 and 3 are equally challenging; Llama 2 achieves similar performance (0.65) on both. These different patterns suggest that the models not only differ in their quantitative ability to perform the task but also in the types of errors they make.

Qualitative analysis To shed light on the cases where each model succeeds and fails, we conduct a qualitative analysis on a handful of samples, i.e., one underspecified sentence per linguistic phenomenon, with the best-performing Mistral, Llama 2, and OPT LMs. The actual sentences considered in this analysis can be found in Table 2.

Among types 1 and 2 of underspecification, we find that for almost all phenomena, the qualitative inspection closely mirrors the quantitative results reported in Figure 2 – Mistral is consistently better than all other models, and Llama 2 outperforms OPT. However, this is much less the case for the *conjunction scopal ambiguity*, where both OPT-13B and Llama2-7B perform at a similar level and are much closer to Mistral. This is in line with the overall quantitative trends in Appendix F. We hypothesize that this may be because the underspecified sentences displaying this phenomenon can be considered difficult to parse. If true, this would suggest that LMs may use some notion of sentence complexity (e.g. the difficulty to parse it) as a stand-in for underspecification.

We also observe that, for referential ambiguity, performance for all models is very low. This may be because most sentences containing referential ambiguity are part of the Winograd Schema Challenge dataset, which may be part of the training data of the tested LMs. As an effect of this, the perplexity assigned to these underspecified sentences may be lower than that of the more uncommon control sentences.

Takeaways and discussion Overall, our results suggest that modern LMs can moderately identify underspecification if explicitly asked to do so via prompting. In particular, the observation that newer models perform better suggests that pre-training with more and better data, using more parameters, and relying on various (even if minor) architectural improvements could eventually lead to models that can accurately recognize underspecified language.

However, our current approach does involve significant prompting, which has several disadvantages when evaluating LMs' linguistic abilities. Model performance is highly sensitive to the specific prompt used (Mizrahi et al., 2024). Moreover, previous work has shown that using meta-linguistic prompts, which explicitly ask for a model's linguistic judgment, often yields different results than eval-

Phenomenon	Sentence	OPT	Llama	Mistral
Logical Form	Danny approached Andrei; Yevgeni, too	0.25	0.75	1
Ellipsis	Andrei and Danny put down a yellow chair	0.5	0.75	1
PP attach. amb.	Andrei approached the person with a yellow bag	0.5	0.75	1
VP attach. amb.	Danny looked at Andrei moving a green chair	0.5	0.75	1
Conjunction amb.	Andrei and Danny held the yellow bag and chair	0.75	0.75	1
Referential amb.	Although they ran at about the same speed, Sue beat Sally because she had such a bad start	0.25	0.25	0.25
Added compound	Get yourself a flannel shirt and wear it over a plain tee shirt.	0.75	0.75	0.75
Fused head	This means you have broken the seal and can now twist off the lid.	0.5	1	0.75
Implicit reference	4. Do not slurp.	0.75	0.75	1
Metonymic ref.	Think about your plant's activity	0.25	1	1

Table 2: Underspecified sentences and proportion of examples (across prompts and orderings) where the specification-matched input received lower perplexity than its mismatched counterpart for OPT-13B, Llama2-7B, and Mistral 7B; higher is better. Horizontal lines demarcate the types of underspecification as per Egg's (2010) taxonomy. Phenomena containing underspecification without ambiguity are *italicized*. 'amb.' stands for ambiguity, 'ref.' for reference.

uating linguistic tasks using naturalistic data (Hu and Levy, 2023). While our experiments do not directly ask models if a given sentence is underspecified (we instead compare two versions of the same sentence), our inputs still do not reflect naturalistic data. Our second experiment is motivated by the need to account for this issue.

6 Interpreting Underspecified and Specified Sentences

The results of our first experiment suggest that some LMs may be able to recognize underspecification when asked about it. However, as discussed above, there is no guarantee that the results obtained using metalinguistic prompts are indicative of the actual capabilities of LMs. In the second experiment, we use a more naturalistic setting where underspecification is not mentioned in the prompt.

6.1 Experimental Setup

We create two specified versions of each DUST sentence originating from the LAVA dataset, corresponding to the possible readings of the original sentence. We also create two continuations to the sentence, which again correspond to distinct readings of the original. Each continuation is compatible with the underspecified sentence, but only one of the fully specified sentences. We slightly adjust the LAVA sentences to make them more correct.[5]

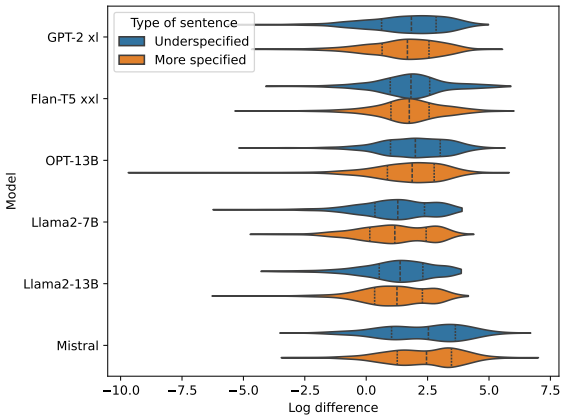

Figure 3: Log difference of perplexity of correct and incorrect continuations for underspecified (blue) and more specified (orange) sentences, averaged over prompts. The underspecified condition is not significantly lower than the more specified condition for any model.

We then embed these sentences and continuations in simple templates like "`[sentence]`. That is, `[continuation]`". Given an underspecified sentence, where both continuations are equally plausible, LMs should ideally assign roughly equal perplexity to both.[6] For more specified sentences,

[5]Uses of 'put-down' and 'picked-up' were changed to 'put down' and 'picked up', and sentences of the form "$\texttt{NNP}_1$ V $\texttt{NNP}_2$. Also $\texttt{NNP}_3$." were changed to "$\texttt{NNP}_1$ V $\texttt{NNP}_2$; $\texttt{NNP}_3$ too."

[6]Underspecified sentences could still have a more frequent "default reading", making one continuation more likely in human speakers' interpretation. For example, when resolving referential ambiguity, it has been documented that a reader might default to the first possible referent encountered in the text. Alternatively, it has been proposed that, when parsing

in contrast, only one interpretation is possible; the corresponding continuation should therefore receive a higher probability than the incompatible one.

For example, given the underspecified sentence "Danny looked at Andrei with a telescope," the continuations "That is, Andrei had a telescope," and "That is, Danny had a telescope," should be similarly likely. However, given the sentence "Danny looked at Andrei, who had a telescope," the first continuation would be more likely.

We experiment with connecting the sentence and continuation in different ways; besides inserting "That is," between them, we also try using no connector ("`[sentence]. [continuation]`"), and stating a more explicit connection: "`[sentence]`. Therefore, it is more likely that `[continuation1]` than `[continuation2]`.".

We record the absolute value of the difference in the perplexities assigned to each continuation given an underspecified sentence. We expect models will generally have only weak preferences between continuations when given underspecified sentences, leading to smaller absolute differences; specified sentences should have larger differences. For each specified sentence, we also record if LMs prefer the plausible continuation over the implausible.

6.2 Results

LMs do not have stronger preferences toward specified, rather than underspecified sentences Our results (Figure 3) indicate that the tested models do not interpret underspecified sentences and their specified counterparts correctly: there is no statistically significant difference between the differences in perplexities assigned to continuations of underspecified sentences, and differences in those assigned to continuations of more specified sentences. This suggests that models might incorrectly assign one single interpretation to underspecified sentences, or they might also fail to assign only one interpretation to more specified sentences; a combination of these is also possible.

The more explicit the prompt, the better the results Our results (Figure 4) indicate that the type of prompt used heavily influenced the degree to which models preferred the correct, plausible continuation; however, unlike in experiment 1, performance is poor overall. In the base case, where no prompts were used, only Llama 2 7B performs significantly above chance. With the "that is" prompt, model performance improves, though there is no clear trend in which models perform best. It is only with the most extensive prompt that we recover both better performance and the trend from experiment 1, where newer models perform better, with Mistral once again performing significantly ($p = .002$) better than all other models.

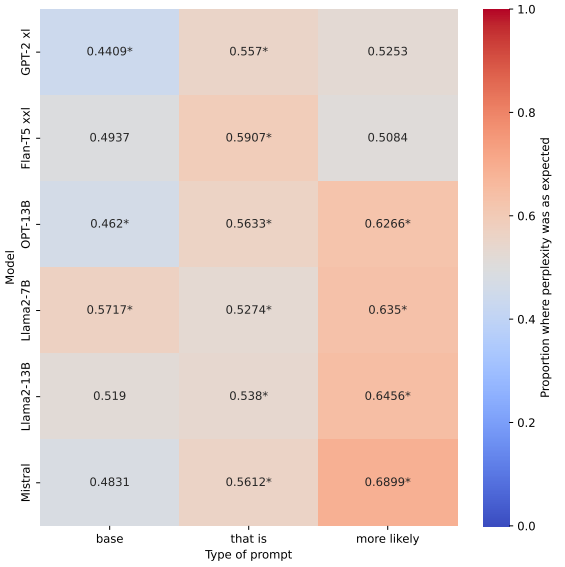

Figure 4: Proportion of inputs where the plausible continuation to a specified sentence received lower perplexity; higher is better. Asterisks (*) indicate performance that differs significantly ($p < .05$) from chance (50%).

This trend, where more explicit prompts yield better results, is surprising: prior work (Hu and Levy, 2023) suggested that metalinguistic prompts might more poorly capture LM capabilities, underestimating them. We hypothesize that this could occur because the continuations we crafted are sometimes more like conclusions that follow from the first sentence (as in NLI), and less like genuinely probable continuations. We note, however, that other work has observed that both LMs and humans sometimes default to NLI when given sentence pairs without instructions (Webson et al., 2023).

Qualitative analysis We also examine the degree to which models assign less univocal interpretations to underspecified sentences than their specified counterparts. Table 3 shows the absolute difference in perplexity assigned to the continuations of one under- and one more specified sen-

a sentence, speakers could leave ambiguities unresolved if resolving is not strictly necessary (Swets et al., 2008). Since we consider both sentence readings in our experiment, we hypothesize that the effects of such default readings cancel out and therefore do not affect our results.

Sentences and Continuations	OPT-13B	Llama2-7B	Mistral
Andrei looked at Danny holding a yellow bag	20.49	2.26	48.51
Andrei looked at Danny while holding a yellow bag	13.77	**2.74**	32.61
→[Andrei / Danny] had a yellow bag			
Andrei looked at Danny holding a yellow bag	5.07	2.42	12.81
Andrei looked at Danny while holding a yellow bag	**6.03**	**5.54**	**13.32**
→*That is,* [Andrei / Danny] had a yellow bag			
Andrei looked at Danny holding a yellow bag	1.47	0.68	2.91
Andrei looked at Danny while holding a yellow bag	1.34	0.23	**5.84**
→*Therefore, it is more likely that* [Andrei...] *than* [Danny...]			

Table 3: For each prompt type (base, *that is*, and *more likely*), an underspecified sentence and one more specified counterpart, along with the absolute difference in perplexities assigned to the continuations of each. The LM is right when this number is higher for the second, more specified, than the first, underspecified, sentence (**bold** cases).

tence, by prompt and model. On this particular example, the models unanimously assign the more specified sentence a more univocal reading only on the "that is" prompt; on the others, they disagree, echoing the noisiness of our quantitative results.

This noisiness is also reflected in the differences between linguistic phenomena and between type of prompt. We find that for referential and VP attachment ambiguity, a greater proportion of inputs resulted in the correct continuation receiving lower perplexity. This seems to suggest that correctly interpreting sentences containing these two phenomena is handled better by the models, although this differs greatly per model and type of prompt. However, we note that relatively few sentences contain these linguistic phenomena, limiting our ability to draw strong conclusions from this observation.

7 Discussion

Underspecification, much like ambiguity (Liu et al., 2023), remains a challenging phenomenon for LMs. Older LMs, such as GPT-2, perform near chance level at recognizing underspecification; newer models, such as Llama 2 and Mistral, perform much better, but still leave ample room for improvement. Processing sentences containing underspecification is an even harder task for LMs. Models seem to fail to recognize when underspecified sentences license continuations that their more specified counterparts do not; moreover, their interpretations of more specified sentences are often incorrect.

The striking difference between the results of our two experiments highlights the importance of carefully choosing a setup when evaluating model capabilities. In the first, metalinguistic prompts elicited good underspecification judgments from high-performing models. But, in surprising contrast to previous work (Hu and Levy, 2023), testing models' ability to process underspecification in a more naturalistic setting led to lower performance. This is important: LM use cases involving underspecification will most likely involve processing underspecification, rather than identifying it upon explicit request. Our second experiment may thus be a better indicator of LMs' practical abilities. However, future work may be needed to compare LM processing of underspecified sentences to results of human studies, which have shown that speakers do have default interpretations of such sentences (Kurtzman and MacDonald, 1993; Dwivedi, 2013).

By introducing DUST and studying underspecification in LMs as distinct from ambiguity, we have taken the first step towards evaluating LMs' performance on a commonplace but understudied phenomenon that can affect LM behavior. Our findings show that current LMs are limited in their ability to deal with underspecification, especially in genuine communicative scenarios. Hence, a thorough evaluation of the abilities of LMs should include (various facets of) underspecification, unlike current benchmarks, in which ambiguous and underspecified sentences are often systematically excluded. We hope that our research further showcases the relevance of underspecification as a direction of research in the study of language models.

Limitations

DUST is arguably a small dataset, and would benefit from expansion. While we considered existing resources and extracted linguistic data with the desired features from those, future work could expand it by collecting new data via human annotation,

generation, or other data-driven approaches. This holds particularly true for type 1, which contains much fewer examples than the other types.

While the present work performs an in-depth evaluation of how LMs behave when faced with semantic underspecification, our research does not explore the inner mechanisms that underlie this capability. We acknowledge that doing so would provide complementary evidence that may be needed to shed full light on the phenomenon. Moreover, research could focus on how LMs handle underspecification in more naturalistic scenarios, e.g., in the context of real-world NLP applications, which is something the current work does not explore.

Our research builds on the formal categorization of semantic underspecification by Egg (2010). While this theoretical framework is both comprehensive and generally suitable for our purposes, we are aware that other theoretical accounts may define the semantic underspecification slightly differently, by including more/less phenomena or by categorizing them according to different criteria. Future work could explore whether and how our findings generalize to other formalizations.

Ethics Statement

While this work presents no serious ethical concerns, a general consideration needs to be made about the use of pre-trained LMs. As is commonly acknowledged, these models should be used with caution as they could perpetuate harmful biases present in their training data. Furthermore, there is a risk that they will generate false or misleading output. In our work, we minimize these risks as we do not use the LMs to generate output, but only to score the plausibility of sentences fed as input. At the same time, we are also aware that some biases may also be present in the linguistic data we used.

Acknowledgements

Some of the contents of this work are based on FW's Master's thesis. Full acknowledgments from the first author can be found therein. We thank the anonymous ARR reviewers for their valuable feedback and the members of the Dialogue Modelling Group at the University of Amsterdam for their insightful comments. MH's research is supported by an OpenAI Superalignment Fellowship.

References

Yevgeni Berzak, Andrei Barbu, Daniel Harari, Boris Katz, and Shimon Ullman. 2015. Do You See What I Mean? Visual Resolution of Linguistic Ambiguities. In *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing*, pages 1477–1487, Lisbon, Portugal. Association for Computational Linguistics.

Marc Brysbaert, Amy Beth Warriner, and Victor Kuperman. 2014. Concreteness ratings for 40 thousand generally known English word lemmas. *Behavior Research Methods*, 46(3):904–911.

Hyung Won Chung, Le Hou, Shayne Longpre, Barret Zoph, Yi Tay, William Fedus, Yunxuan Li, Xuezhi Wang, Mostafa Dehghani, Siddhartha Brahma, Albert Webson, Shixiang Shane Gu, Zhuyun Dai, Mirac Suzgun, Xinyun Chen, Aakanksha Chowdhery, Alex Castro-Ros, Marie Pellat, Kevin Robinson, Dasha Valter, Sharan Narang, Gaurav Mishra, Adams Yu, Vincent Zhao, Yanping Huang, Andrew Dai, Hongkun Yu, Slav Petrov, Ed H. Chi, Jeff Dean, Jacob Devlin, Adam Roberts, Denny Zhou, Quoc V. Le, and Jason Wei. 2022. Scaling Instruction-Finetuned Language Models. ArXiv:2210.11416 [cs].

Veena Dwivedi. 2013. Interpreting quantifier scope ambiguity: Evidence of heuristic first, algorithmic second processing. *PLoS ONE*, 8.

Markus Egg. 2010. Semantic Underspecification. *Language and Linguistics Compass*, 4(3):166–181.

Allyson Ettinger. 2020. What BERT Is Not: Lessons from a New Suite of Psycholinguistic Diagnostics for Language Models. *Transactions of the Association for Computational Linguistics*, 8:34–48.

Alessandro Fantechi, Stefania Gnesi, and Laura Semini. 2023. Rule-based NLP vs ChatGPT in Ambiguity Detection, a Preliminary Study. In *Joint Proceedings of REFSQ-2023 Workshops*, Barcelona.

Francesca Franzon and Chiara Zanini. 2022. The Entropy of Morphological Systems in Natural Languages Is Modulated by Functional and Semantic Properties. *Journal of Quantitative Linguistics*, 30(1):42–66.

Steven Frisson. 2009. Semantic Underspecification in Language Processing. *Language and Linguistics Compass*, 3(1):111–127.

H. P. Grice. 1969. Utterer's meaning and intention. *The Philosophical Review*, 78(2):147–177.

Michael Hanna, Yonatan Belinkov, and Sandro Pezzelle. 2023. When language models fall in love: Animacy processing in transformer language models. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pages 12120–12135, Singapore. Association for Computational Linguistics.

Daniel W. Harris. 2020. What Makes Human Communication Special? In *Unpublished book manuscript*. CUNY Graduate Center. Draft of October 27, 2020.

Jennifer Hu and Roger Levy. 2023. Prompting is not a substitute for probability measurements in large language models. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pages 5040–5060, Singapore. Association for Computational Linguistics.

Ben Hutchinson, Jason Baldridge, and Vinodkumar Prabhakaran. 2022. Underspecification in Scene Description-to-Depiction Tasks. In *Proceedings of the 2nd Conference of the Asia-Pacific Chapter of the Association for Computational Linguistics and the 12th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, pages 1172–1184, Online only. Association for Computational Linguistics.

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.

Victor Kuperman, Hans Stadthagen-Gonzalez, and Marc Brysbaert. 2012. Age-of-acquisition ratings for 30,000 English words. *Behavior Research Methods*, 44(4):978–990.

Howard S. Kurtzman and Maryellen C. MacDonald. 1993. Resolution of quantifier scope ambiguities. *Cognition*, 48:243–279.

Shalom Lappin. 2000. An Intensional Parametric Semantics for Vague Quantifiers. *Linguistics and Philosophy*, 23(6):599–620.

Hector Levesque, Ernest Davis, and Leora Morgenstern. 2012. The Winograd Schema Challenge. In *Proceeding of the Thirteenth International Conference on Principles of Knowledge Representation and Reasoning*, pages 552–561. Institute of Electrical and Electronics Engineers Inc.

Stephen C. Levinson. 2000. *Presumptive meanings: the theory of generalized conversational implicature*. The MIT Press. OCLC: 956673720.

Alisa Liu, Zhaofeng Wu, Julian Michael, Alane Suhr, Peter West, Alexander Koller, Swabha Swayamdipta, Noah A. Smith, and Yejin Choi. 2023. We're Afraid Language Models Aren't Modeling Ambiguity. ArXiv:2304.14399 [cs].

Greg Maciejewski and Ekaterini Klepousniotou. 2016. Relative Meaning Frequencies for 100 Homonyms: British eDom Norms. *Journal of Open Psychology Data*, 4:e6.

Moran Mizrahi, Guy Kaplan, Dan Malkin, Rotem Dror, Dafna Shahaf, and Gabriel Stanovsky. 2024. State of what art? a call for multi-prompt llm evaluation.

Subhabrata Mukherjee and Pushpak Bhattacharyya. 2012. Wikisent: Weakly supervised sentiment analysis through extractive summarization with wikipedia. In *Machine Learning and Knowledge Discovery in Databases*, pages 774–793, Berlin, Heidelberg. Springer Berlin Heidelberg.

Joachim Niehren, Manfred Pinkal, and Peter Ruhrberg. 1997. A Uniform Approach to Underspecification and Parallelism. In *35th Annual Meeting of the Association for Computational Linguistics and 8th Conference of the European Chapter of the Association for Computational Linguistics*, pages 410–417, Madrid, Spain. Association for Computational Linguistics.

Peter Norvig. 2009. Natural language corpus data. *Beautiful data*, pages 219–242.

Miguel Ortega-Martín, Óscar García-Sierra, Alfonso Ardoiz, Jorge Álvarez, Juan Carlos Armenteros, and Adrián Alonso. 2023. Linguistic ambiguity analysis in ChatGPT. ArXiv:2302.06426 [cs].

Sandro Pezzelle. 2023. Dealing with Semantic Underspecification in Multimodal NLP. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 12098–12112, Toronto, Canada. Association for Computational Linguistics.

Steven T. Piantadosi, Harry Tily, and Edward Gibson. 2012. The communicative function of ambiguity in language. *Cognition*, 122(3):280–291.

Manfred Pinkal. 1999. On Semantic Underspecification. In Harry Bunt and Reinhard Muskens, editors, *Computing Meaning: Volume 1*, Studies in Linguistics and Philosophy, pages 33–55. Springer Netherlands, Dordrecht.

Massimo Poesio. 1994. Ambiguity, Underspecification and Discourse Interpretation. In *Proceedings of the First International Workshop on Computational Semantics*.

Archiki Prasad, Elias Stengel-Eskin, and Mohit Bansal. 2023. Rephrase, augment, reason: Visual grounding of questions for vision-language models.

Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, Gretchen Krueger, and Ilya Sutskever. 2021. Learning transferable visual models from natural language supervision.

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

Anna Rogers, Olga Kovaleva, and Anna Rumshisky. 2020. A primer in BERTology: What we know about how BERT works. *Transactions of the Association for Computational Linguistics*, 8:842–866.

Michael Roth, Talita Anthonio, and Anna Sauer. 2022. SemEval-2022 Task 7: Identifying Plausible Clarifications of Implicit and Underspecified Phrases in Instructional Texts. In *Proceedings of the 16th International Workshop on Semantic Evaluation (SemEval-2022)*, pages 1039–1049, Seattle, United States. Association for Computational Linguistics.

Adam Sennet. 2023. Ambiguity. In Edward N. Zalta and Uri Nodelman, editors, *The Stanford Encyclopedia of Philosophy*, spring 2023 edition. Metaphysics Research Lab, Stanford University.

Richard Socher, Alex Perelygin, Jean Wu, Jason Chuang, Christopher D. Manning, Andrew Ng, and Christopher Potts. 2013. Recursive deep models for semantic compositionality over a sentiment treebank. In *Proceedings of the 2013 Conference on Empirical Methods in Natural Language Processing*, pages 1631–1642, Seattle, Washington, USA. Association for Computational Linguistics.

Elias Stengel-Eskin, Kyle Rawlins, and Benjamin Van Durme. 2024. Zero and few-shot semantic parsing with ambiguous inputs.

Benjamin Swets, Timothy Desmet, Charles Clifton, and Fernanda Ferreira. 2008. Underspecification of syntactic ambiguities: Evidence from self-paced reading. *Memory & Cognition*, 36(1):201–216.

Alberto Testoni and Raquel Fernández. 2024. Asking the right question at the right time: Human and model uncertainty guidance to ask clarification questions.

Hugo Touvron, Louis Martin, Kevin Stone, Peter Albert, Amjad Almahairi, Yasmine Babaei, Nikolay Bashlykov, Soumya Batra, Prajjwal Bhargava, Shruti Bhosale, Dan Bikel, Lukas Blecher, Cristian Canton Ferrer, Moya Chen, Guillem Cucurull, David Esiobu, Jude Fernandes, Jeremy Fu, Wenyin Fu, Brian Fuller, Cynthia Gao, Vedanuj Goswami, Naman Goyal, Anthony Hartshorn, Saghar Hosseini, Rui Hou, Hakan Inan, Marcin Kardas, Viktor Kerkez, Madian Khabsa, Isabel Kloumann, Artem Korenev, Punit Singh Koura, Marie-Anne Lachaux, Thibaut Lavril, Jenya Lee, Diana Liskovich, Yinghai Lu, Yuning Mao, Xavier Martinet, Todor Mihaylov, Pushkar Mishra, Igor Molybog, Yixin Nie, Andrew Poulton, Jeremy Reizenstein, Rashi Rungta, Kalyan Saladi, Alan Schelten, Ruan Silva, Eric Michael Smith, Ranjan Subramanian, Xiaoqing Ellen Tan, Binh Tang, Ross Taylor, Adina Williams, Jian Xiang Kuan, Puxin Xu, Zheng Yan, Iliyan Zarov, Yuchen Zhang, Angela Fan, Melanie Kambadur, Sharan Narang, Aurelien Rodriguez, Robert Stojnic, Sergey Edunov, and Thomas Scialom. 2023. Llama 2: Open foundation and fine-tuned chat models.

Wolfgang Wahlster. 2000. Mobile Speech-to-Speech Translation of Spontaneous Dialogs: An Overview of the Final Verbmobil System. In Wolfgang Wahlster, editor, *Verbmobil: Foundations of Speech-to-Speech Translation*, pages 3–21. Springer Berlin Heidelberg, Berlin, Heidelberg.

Albert Webson, Alyssa Marie Loo, Qinan Yu, and Ellie Pavlick. 2023. Are language models worse than humans at following prompts? it's complicated. In *The 2023 Conference on Empirical Methods in Natural Language Processing*.

Albert Webson and Ellie Pavlick. 2022. Do prompt-based models really understand the meaning of their prompts? In *Proceedings of the 2022 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 2300–2344, Seattle, United States. Association for Computational Linguistics.

Thomas Wolf, Lysandre Debut, Victor Sanh, Julien Chaumond, Clement Delangue, Anthony Moi, Pierric Cistac, Tim Rault, Rémi Louf, Morgan Funtowicz, Joe Davison, Sam Shleifer, Patrick von Platen, Clara Ma, Yacine Jernite, Julien Plu, Canwen Xu, Teven Le Scao, Sylvain Gugger, Mariama Drame, Quentin Lhoest, and Alexander M. Rush. 2020. Huggingface's transformers: State-of-the-art natural language processing.

Susan Zhang, Stephen Roller, Naman Goyal, Mikel Artetxe, Moya Chen, Shuohui Chen, Christopher Dewan, Mona Diab, Xian Li, Xi Victoria Lin, Todor Mihaylov, Myle Ott, Sam Shleifer, Kurt Shuster, Daniel Simig, Punit Singh Koura, Anjali Sridhar, Tianlu Wang, and Luke Zettlemoyer. 2022. OPT: Open Pre-trained Transformer Language Models.

Wanzheng Zhu and Suma Bhat. 2020. GRUEN for Evaluating Linguistic Quality of Generated Text. In *Findings of the Association for Computational Linguistics: EMNLP 2020*, pages 94–108, Online. Association for Computational Linguistics.

Arnold M. Zwicky and Jerrold M. Sadock. 1975. Ambiguity tests and how to fail them. *Syntax and Semantics*, 4:1–36.

A Type 4 of underspecification

In an attempt to collect sentences containing this type of underspecification, we collect sentences containing homonymic expressions from a sample of English Wikipedia (Mukherjee and Bhattacharyya, 2012). We do so by selecting sentences that contain any homonym from a list of 100 homonyms, selected by Maciejewski and Klepousniotou (2016) based on linguistic principles, dictionary entries, and subjective ratings. We created more specified counterparts by selecting random sentences from the that contain none of the homonyms from the list. We collect a total of 980 sentence pairs; note that unlike other pairs in DUST, these are not minimal pairs. One (partial) sentence of this type (though not from our dataset) is

The elderly fish.

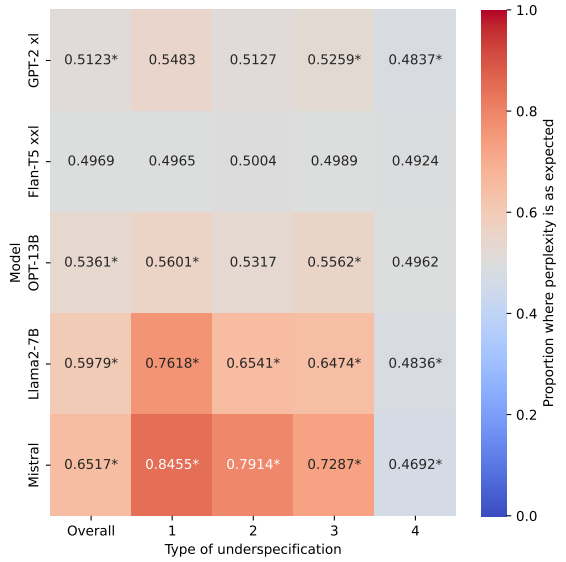

Figure 5: Proportion of specification-matched inputs with lower perplexity than their specification-mismatched counterparts; higher is better. Asterisks (*) indicate performance significantly ($p < .05$) above or below chance (50%).

which is underspecified because 'fish' can be both a noun or a verb in this context. A more specified counterpart of this (partial) sentence could be

The elderly people fish.

Note, however, that the reading where 'fish' is a noun is not a full sentence, and would only be grammatical when placed in a context (e.g. an enumeration) where it is suitable.

Running experiment 1 with these sentences included, we get the results shown in Figure 5. We can see that models achieve poor performance on type 4 sentences across models. We hypothesize that this is because that type of underspecification does not consist of minimal pairs; though one item of the pair does include a homonymic expression, this does not guarantee that it is overall less specified than its counterpart. Due to these caveats, we have excluded this type from our dataset.

B Dataset Information

In this section, we review important licensing and privacy information regarding the datasets composing DUST, as well as DUST itself. DUST is composed of data from 3 datasets.

- LAVA (Berzak et al., 2015), available at https://web.mit.edu/lavacorpus/, was released with an unclear (potentially open-source) license. LAVA contains images of the authors, which DUST does not include. However, the names used in the sentences in LAVA refer to its authors. These sentences contain no other personally identifiable or offensive content.

- CLAIRE (Roth et al., 2022), available at https://github.com/acidAnn/claire, is composed of WikiHow articles released under a CC BY-NC-SA 3.0 license, and was itself released under the same license. We did not filter the articles and revisions of which CLAIRE is comprised for personally identifiable or offensive content.

- The Winograd Schema Challenge dataset (Levesque et al., 2012), available at https://huggingface.co/datasets/winograd_wsc, was released under a CC BY 4.0 license. All sentences were created by experts, and do not contain personally identifiable or offensive content.

DUST is a non-commercial dataset which, like all of its component datasets, is intended for research purposes. We have also provided attribution to the creators of the component datasets, allowing it to be released in accordance with all of the licensing terms of these component datasets. Owing to the WikiHow data contained within, we also release DUST with a CC BY-NC-SA license.

C Model and Experimental Details

In this section, we provide further information regarding our models and experiments. We use the HuggingFace transformers implementations of all models, available at https://huggingface.co/models. We also use the HuggingFace weights for all models except Llama 2, which must be downloaded separately at https://llama.meta.com/llama-downloads/. To compute perplexities, we use LM-PPL: https://github.com/asahi417/lmppl.

We ran these experiments on compute nodes equipped with Nvidia A100 GPUs (40GB RAM); for all models but Llama 2 13B, one such GPU should suffice. The runtime of our experiments is no more than 5 GPU days.

D MCQ experiment

In this experiment, we test whether language mod-

els can recognize semantic underspecification by explicitly asking them to generate an answer about the underspecification of a sentence pair. In practice, we prompt GPT2-xl, OPT-13B, and Llama2-7B to generate a response using the following prompt:

> Here are two sentences. A: 'Andrei left the chair with a green telescope'. B: 'Andrei left the chair on which lay a telescope'. Which one of these is more semantically underspecified? Please respond by outputting only A or B. Answer:

where the red and blue sentences are replaced by either an underspecified or its corresponding control sentence from the dataset. The order in which the two sentences are placed is randomized to prevent bias in the model from unduly influencing the final accuracy. Each sentence pair in the dataset is included in a prompt once. In Table 4, we report model accuracy and number of A, B, or other responses.

model	acc.	#A	#B	#other
GPT2-xl	0.31	696	638	789
OPT-13B	0.49	2105	18	0
Llama2-7B	0.48	888	1138	97

Table 4: MCQ task. Model accuracy and number of responses per class by GPT2, OPT, and Llama 2.

The results show that the models perform very poorly when the task is formulated as a multiple-choice task. While the low accuracy might be a result of the models being unable to do this task—something that could perhaps improve when using instruction-tuned models—the observation that all the models are either biased towards one of the options or unable to consistently answer the prompt with one of the two given options, or both, suggests that they are incapable of performing the task in this experimental setup. This matches earlier findings (e.g., Hu and Levy, 2023) and validates our decision to perform perplexity-based evaluation over a MCQ-like type of experiment even further.

E Experiment 1 Prompts

Suppose we have the underspecified sentence 'Andrei left the chair with a blue telescope' and the more specified counterpart 'Andrei left the chair on which lay a blue telescope'. Examples of specification-matched prompts we would then obtain are:

> This is an underspecified sentence: 'Andrei left the chair with a blue telescope'. This is its more specified counterpart: 'Andrei left the chair on which lay the blue telescope'.

and

> This is a sentence that contains a lot of detail: 'Andrei left the chair on which lay the blue telescope'. This is a sentence that contains little detail: 'Andrei left the chair with a blue telescope'.

and examples of mismatched prompts are:

> This is an ambiguous sentence: 'Andrei left the chair on which lay the blue telescope'. This is its unambiguous counterpart: 'Andrei left the chair with a blue telescope'.

and

> This is a sentence that contains a lot of information: 'Andrei left the chair with a blue telescope'. This is its counterpart that contains little information 'Andrei left the chair on which lay the blue telescope'.

These examples show all variations of phrasing and order of parts

F Experiment 1 Results per Phenomenon

In Figure 6, we report the results of Experiment 1 split by linguistic phenomenon.

G Sentiment perplexity

To test whether the experimental design of experiment 1 functions correctly, we first ran this experiment with sentiment classification instead of recognition of underspecification as a goal. The models were prompted with prompts of the form "prompt1: 'sentence1'. prompt2: 'sentence2'.", where the prompts are of the form "This is a positive/negative sentence" and the sentences are sentences rated 'very positive' or 'very negative' in the SST-5 dataset (Socher et al., 2013). The results of this can be seen in Figure 7.

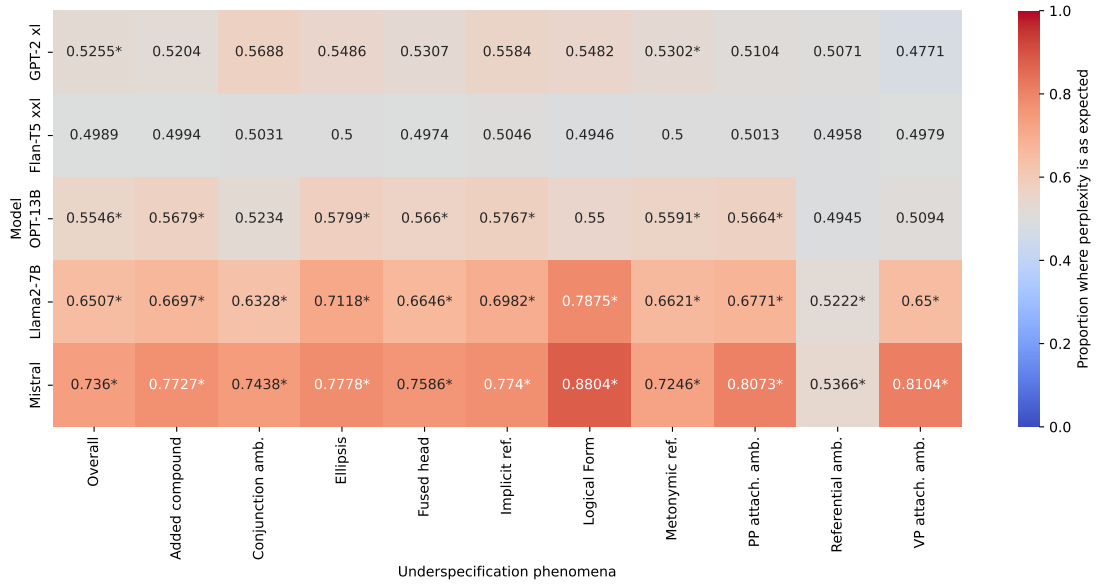

Figure 6: Proportion of specification-matched inputs with a lower perplexity than their specification-mismatched counterparts, split by linguistic phenomenon; higher is better. Asterisks (*) indicate performance significantly ($p < .05$) above or below chance (50%).

H Experiment 2 Prompts

Suppose we have the underspecified sentence 'Andrei looked at Danny moving a green bag' and the more specified counterpart 'Andrei looked at Danny who was moving a green bag '. Examples of specification-matched prompts we would then obtain, from low to high levels of prompting, are:

> Andrei looked at Danny who was moving a green bag. Danny was moving a green bag.
>
> Andrei looked at Danny who was moving a green bag. That is, Danny was moving a green bag.
>
> Andrei looked at Danny who was moving a green bag. Therefore, it is more likely that Danny was moving a green bag than Andrei was moving a green bag.

and examples of mismatched prompts are

> Andrei looked at Danny who was moving a green bag. Andrei was moving a green bag.
>
> Andrei looked at Danny who was moving a green bag. That is, Andrei was moving a green bag.
>
> Andrei looked at Danny who was moving a green bag. Therefore, it is more likely that Andrei was moving a green bag than Danny was moving a green bag.

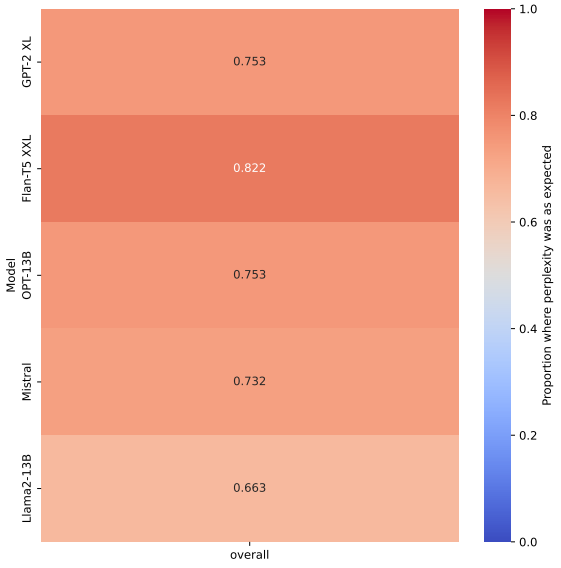

Figure 7: Percentage of sentences from experiment 1 with sentiment instead of underspecification where the perplexity of the correct prompts is lower than that of the incorrect prompts, averaged over orders of prompts

Variable	Coefficient	Standard Error	z	$P > \|z\|$	$[0.025$	$0.975]$
sen. len.	-0.155	0.0623	-2.3418	0.2135	-0.2775	-0.0328
avg. AoA.	-0.2616	0.2443	-1.046	0.3642	-0.7405	0.2173
avg. conc.	1.7596	0.65	2.6663	**0.0262**	0.4855	3.034
avg. word freq.	-2.9801	2.1483	-1.394	0.1245	-7.1908	1.2307
avg. word len.	0.0367	0.4773	0.062	0.3468	-0.8987	0.972

Table 5: Average regression coefficients averaged over all tested models with sentence length, average age of acquisition (Kuperman et al., 2012), average concreteness (Brysbaert et al., 2014), average word frequency (Norvig, 2009) and average word length as independent variables, and whether the perplexity of the correct continuation of the specified counterpart is higher than that of the incorrect continuation as the dependent variable

I Could models be detecting surface statistics instead of underspecification?

To investigate whether the models' ability to interpret underspecification as underspecification correlates with some surface-level statistic of the sentences in the dataset, we fit a logistic regression model with surface-level descriptive statistics about each sentence as independent variables and the model 'correctness' from Figure 4 as the dependent variable. The results of this can be seen in Table 5.

These results suggest that models are better able to recognize semantic underspecification when the words in the sentence are more concrete. No other surface-level statistic we test shows a significant correlation with the ability of the models to interpret underspecification.

This agrees with intuition: unlike other features like age of acquisition, word frequency or sentence length, concreteness is something humans also associate with (under)specification – for example, when a speaker wants to make things less underspecified, they might say "let's make things concrete". However, the fact that models do not correctly interpret underspecified sentences when these sentences are abstract in nature does pose a problem, given the fact that certain types of text (e.g. legal texts or product specification documents) can be very abstract while requiring all potential underspecification to be correctly interpreted.