LIEDER: Linguistically-Informed Evaluation for Discourse Entity Recognition

Anonymous ACL submission

Abstract

Discourse Entity (DE) recognition is the task of identifying novel and known entities introduced within a text. While previous work has found that large language models have basic, if imperfect, DE recognition abilities (Schuster and Linzen, 2022), it remains largely unassessed which of the fundamental semantic properties that govern the introduction and subsequent reference to DEs they have knowledge of. We propose the Linguistically-Informed Evaluation for Discourse Entity Recognition (LIEDER) dataset that allows for a detailed examination of language models' knowledge of four crucial semantic properties: EXISTENCE, UNIQUENESS, PLURALITY, and NOVELTY. We find evidence that state-of-the-art large language models exhibit sensitivity to all of these properties except NOVELTY, which demonstrates that they have yet to reach human-level language understanding abilities.

1 Introduction

One central component of language understanding is the ability to recognize entities in text. The large body of research in Natural Language Processing focuses on the task of Named Entity Recognition, where a system must identify whether a noun phrase (NP), typically a proper name, refers to a known individual of a certain semantic class (Li et al., 2022). The recognition of discourse entities (DEs), in contrast, involves identifying not only the occurrence of known entities but also novel ones that are introduced within a text. The recognition of DEs takes place at two sites: introduction sites and reference sites. Introduction refers to the first time where an entity appears in a discourse. Reference sites are subsequent mentions of an entity that has been previously introduced.

As humans, not only are we able to recognize DEs at both of these sites, but we also have knowledge of **how** to coordinate the introduction and subsequent reference to entities using appropriate linguistic means. For example, we know that the introduction of DEs is typically done using indefinite NPs such as *a man* in '*A man walked into the room*.' We also know that subsequent mentions often involve definite NPs like *the man* in '*The man sat down*.'

Schuster and Linzen (2022) present an evaluation suite for DE recognition that focuses on the question of whether language models are sensitive to the linguistic context in which DEs are introduced. They find that transformer-based language models do not always demonstrate a clear preference for referring to entities that have been properly introduced into the discourse. While this work provides important insight into LM abilities with discourse reference, it does not engage directly with the underlying linguistic properties responsible for DE introduction and reference. As a result, it does not provide a means of assessing more precisely what LMs know about the linguistic encoding of discourse reference.

Semantics research has established properties of definite and indefinite NPs from which their use in introducing and referring to entities follows, four of which are particularly relevant here: EXISTENCE, UNIQUENESS, PLURALITY, and NOVELTY. We will define and discuss these properties in detail in Section 3. A good language model (LM) should reflect knowledge of all of these properties. In this paper, we provide a novel dataset, which builds on Schuster and Linzen's work, that provides a method of testing these properties directly.[1] Our results, across a number of state-of-the-art LLMs, provide evidence for knowledge of EXISTENCE, UNIQUENESS and PLURALITY (all conditions on the use of definite NPs to refer to DEs), but difficulty with NOVELTY (a condition on the introduction of DEs

[1] All code, data, and results will be made available in our GitHub Repository upon acceptance.

by indefinite NPs) unless information about distinctiveness is made explicit. In addition, we find that transformer LMs, unlike humans, show sensitivity to linear distance in establishing DE reference. Taken together, these results suggest that state-of-the-art large language models (LLMs) do not reach human-level language understanding abilities.

2 Assessing Discourse Entity Recognition

Schuster and Linzen (2022) (henceforth SL) develop an evaluation suite that probes LLM performance on DE recognition. Such evaluation was centered on the ability of indefinite phrases to introduce DEs that can be referred to by subsequent occurrences of definites. They identified pairs of contexts that differ in their ability to support DE introduction. The simplest case, and the one we focus on in our experiments, is *affirmative-negation*: indefinites in the context of affirmative sentences introduce DEs that can be referred to by a following definite NP, but they do not in the context of sentences with negation. This is seen in the following example, where we use F to represent a felicitous completion, and I to represent an infelicitous completion:

(1) CONTEXT: John owns a dog but he doesn't own a cat.

  a. F: His dog follows him everywhere.

  b. I: His cat follows him everywhere.

SL argue that if a language model is sensitive to the difference between contexts which do and do not introduce DEs (such as the two conjuncts in (1) CONTEXT), we would expect the following inequality to hold for the probabilities assigned to felicitous (F) and infelicitous (I) continuations:

$$p(\text{F}|\text{CONTEXT}) > p(\text{I}|\text{CONTEXT})$$

They found that the models they examined (which included GPT-2 variants and GPT-3) showed above chance performance on distinguishing F and I continuations in the context of *affirmative-negation*, with GPT-3 showing the most human-like performance.

SL further explored the systematicity of these contrasts, where a contrast is counted as systematic only when the model correctly predicts felicity on all possible orderings of the operators and nouns in the conjoined clauses that comprise the context:

(2) a. Bob owns a dog but he doesn't own a cat.
  b. Bob owns a cat but he doesn't own a dog.
  c. Bob doesn't own a cat but he owns a dog.
  d. Bob doesn't own a dog but he owns a cat.

The relative order of the affirmative and negative sentences does not impact the felicity of the definite descriptions. The continuation *His dog follows him everywhere* should be felicitous after both (2a) and (2c). Similarly, the continuation *His cat follows him everywhere* should be felicitous after both (2b) and (2d). However, SL find that the models are much less successful under this more demanding measure; even GPT-3 systematically distinguishes felicitous and infelicitous continuations only barely above 50% of the time, with other models showing lower performance. Schuster and Linzen do not, however, identify what underlies the models' failure in systematic performance on this task.

While intriguing, these results do not tell us what specifically causes difficulty for LLMs in this task, and how it relates to semantic properties of definite and indefinite NPs. To evaluate language models' DE recognition abilities, we believe that it is important to decompose models' performance in a more granular manner. We turn now to a paradigm that builds on this previous work to do precisely this.

3 Criteria for Discourse Entity Recognition

Informed by theoretical research in natural language semantics, we propose that a thorough evaluation of DE recognition and reference abilities should examine language models' knowledge of four fundamental semantic properties: EXISTENCE, UNIQUENESS, PLURALITY, and NOVELTY. We will define and discuss each of them in order.

Existence As SL argued, a language model with human-level understanding abilities should not use definite descriptions to refer to entities that have not been introduced into the discourse. We define this requirement as EXISTENCE (Russell, 1905). For example, given the context *John doesn't own a dog*, a language model should recognize that the continuation *The dog barks at night* is infelicitous because of the non-existence of a dog DE.

Uniqueness A language model should use a singular definite description to refer to a previously introduced entity only when the referent is unique relative to the discourse. We will call this requirement UNIQUENESS (Russell, 1905; Heim and Kratzer,

Expression	Requirements
Indefinites	NOVELTY
Singular definites	EXISTENCE, UNIQUENESS
Plural definites	EXISTENCE, PLURALITY

Table 1: Expressions used for introducing and referencing DEs and their corresponding requirements.

1998). For example, given the context *John owns a dog and Mark owns a dog too*, the model should consider the continuation *The dog barks at night* as infelicitous. Since more than one dog has been introduced into the discourse, the singular definite phrase is not licensed. On the other hand, if the context is *John does not own a dog but Mark owns a dog*, UNIQUENESS is satisfied, so the same continuation should be judged to be felicitous.

Plurality A language model should only use a plural definite description if the set of DE contains more than one individual of the relevant sort, a requirement we call PLURALITY (Landman, 1989). Notice that UNIQUENESS and PLURALITY cannot be satisfied or violated at the same time – referent expressions that require uniqueness (i.e., singular definites) are infelicitous after contexts that satisfy PLURALITY. In contrast, referent expressions that require plurality (i.e., plural definites) are infelicitous after contexts that satisfy UNIQUENESS. For example, the context *John owns a dog and Mark owns a dog too* satisfies PLURALITY since there are two dog DEs, and therefore supports a plural but not singular continuation (i.e., *The dogs bark/*dog barks*).

Novelty The last requirement concerns the use of indefinite NPs: a language model should recognize that an occurrence of an indefinite noun phrase is associated with the introduction of a new entity into the discourse. Following Heim (1982), we will call this requirement NOVELTY. In the context sentence *John owns a dog and Mark owns a dog too*, this means that two distinct dogs are introduced as DEs.

Table 1 summarizes relevant expression types and the corresponding requirements that a language model should know in order to correctly introduce and refer to DEs. In the next section, we will describe our evaluation dataset and show how it evaluates model performance with respect to the four requirements.

4 The LIEDER Dataset

The Linguistically-Informed Evaluation of DE Recognition (LIEDER) evaluation dataset adapts the structure of SL's paradigm: a context example is provided that consists of two conjoined clauses, each containing an indefinite NP with the same head noun. This is followed by a continuation, test sentence containing a definite description. As in SL, we vary the conjoined clauses as to whether they introduce DEs or not (affirmative or negative). However, in order to evaluate the four linguistic properties defined in the previous section, we make two innovations: (i) we allow zero, one, or both of the conjoined clauses to include negation and thereby fail to introduce a DE; and (ii) we allow the definite description in the continuation to be either singular or plural. Example items in our dataset are shown in Table 2.

The *Context type* column indicates which sides of the conjunction introduce DEs. For example, pos_neg indicates that a DE is introduced in the first conjunct (pos), and no DEs are introduced in the second conjunct (neg). Since we consider all four context types with both singular and plural continuations, there are 8 different context-continuation combinations. These 8 combinations are then crossed with 16 distinct base sentence pairs for the context and continuation, resulting in 128 examples in total.[2]

Different combinations of context type and continuation result in differences in felicity of our two-sentence discourse. The singular continuation is felicitous only when the context type is either pos_neg or neg_pos since these contexts introduce exactly one DE. In contrast, the plural continuation is felicitous only when the type is pos_pos. This means that of the eight context-continuation pairs, of which three are felicitous and five are infelicitous.

Success in the LIEDER dataset requires that a model accurately distinguishes felicitous from infelicitous context-continuation pairs. Because metalinguistic judgments elicited from language models may not reflect the full extent of the model's knowledge (Hu and Levy, 2023), we instead compare felicity using the probabilities the model assigns to the continuation given the context. We

[2] SL consider other pairs of sentential operators that differ in whether they introduce DEs, specifically *managed-failed* and *know-doubt*. The LIEDER dataset also includes contexts that combine these sentential operators.

Context type	Context	Singular Continuation	Plural Continuation
pos_neg	John owns a dog but Mark doesn't own a dog.	The dog is very cute.	#The dogs are very cute.
neg_pos	John doesn't own a dog but Mark owns a dog.	The dog is very cute.	#The dogs are very cute.
pos_pos	John owns a dog and Mark owns a dog too.	#The dog is very cute.	The dogs are very cute.
neg_neg	John doesn't own a dog and Mark doesn't own a dog either.	#The dog is very cute.	#The dogs are very cute.

Table 2: Example contexts and continuations in the LIEDER dataset. Infelicitous continuations are marked with #.

assume that the probability a model assigns to a felicitous case should be greater than the probability it assigns to an infelicitous one. With 3 felicitous pairs and 5 infelicitous ones, this means we have 15 probability comparisons in total. These are depicted in Table 3.

Importantly, success in each of these comparisons can be tied to the linguistic requirements described in Section 3 that are involved in the introduction of and reference to DEs. For example, if the model assigns higher probability to a continuation with a singular definite in a neg_pos context as compared to a neg_neg context, this provides evidence for the model's awareness of the EXISTENCE requirement that singular definite NP imposes; otherwise, the singular should be possible in this context. On the other hand, if the model assigns higher probability to a singular definite in a pos_neg context as compared to a pos_pos context, this indicates that the model is aware of the UNIQUENESS requirement imposed on singular definiteness and the NOVELTY condition imposed on indefinites, since otherwise the pos_pos context could be taken to introduce only a single DE. In addition, if the model assigns higher probability to a singular definite in a neg_pos context than it does to a plural definite in the same context, this provides evidence of sensitivity to the PLURALITY requirement on plural definites, as the context introduces only a single entity. Because of the correspondences between the example types and the linguistic requirements, LIEDER can therefore be used to assess the details of the knowledge of DEs in a language model.

5 Experiment 1: Applying LIEDER

Models We investigated the performance of four open-source large language models on LIEDER. They are GPT-2 (Radford et al., 2019), Llama-7B (Touvron et al., 2023a), Llama 2-7B (Touvron et al., 2023b), and Code Llama-7B (Rozière et al., 2024). GPT-2 has 1.5 billion parameters, and all three other models have 7 billion parameters. In each case, we use the huggingface transformer PyTorch API (Wolf et al., 2019) to load and perform inference.

Comparison Type	Requirement	Section
p(sg|pos_neg)>p(sg|pos_pos)	UNIQUENESS, NOVELTY	5.1.1
p(sg|neg_pos)>p(sg|pos_pos)	UNIQUENESS, NOVELTY	5.1.1
p(sg|neg_pos)>p(sg|neg_neg)	EXISTENCE	5.1.1
p(sg|pos_neg)>p(sg|neg_neg)	EXISTENCE	5.1.1
p(pl|pos_pos)>p(pl|pos_neg)	PLURALITY	5.1.2
p(pl|pos_pos)>p(pl|neg_pos)	PLURALITY	5.1.2
p(pl|pos_pos)>p(pl|neg_neg)	EXISTENCE, PLURALITY	5.1.2
p(sg|pos_neg)>p(pl|pos_neg)	PLURALITY	5.1.3
p(sg|pos_neg)>p(pl|neg_pos)	PLURALITY	5.1.3
p(sg|pos_neg)>p(pl|neg_neg)	EXISTENCE, PLURALITY	5.1.3
p(sg|neg_pos)>p(pl|neg_pos)	PLURALITY	5.1.3
p(sg|neg_pos)>p(pl|pos_neg)	PLURALITY	5.1.3
p(sg|neg_pos)>p(pl|neg_neg)	EXISTENCE, PLURALITY	5.1.3
p(pl|pos_pos)>p(sg|pos_pos)	UNIQUENESS, NOVELTY	5.1.3
p(pl|pos_pos)>p(sg|neg_neg)	EXISTENCE	5.1.3

Table 3: Comprehensive list of all comparison types, the requirements they test, and the experiment that tests for them.

Metric To perform the probability comparisons discussed in Section 4, we provided the model with the context of each test item and calculated the total log probability for the entire continuation. For each comparison type, we compare the log probability of a felicitous continuation with an infelicitous one, and judge it as accurate if the felicitous probability is higher. We then calculate the accuracy for each comparison type over the 16 items of each type.

Human Judgments In addition to evaluating our dataset on large language models, we also conducted a human experiment modeled on the language modeling task. Participants were asked to provide a rating on the acceptability of a given continuation using a continuous slider with possible values between 1 and 7. Each participant provided ratings for 16 experimental sentences (one for each context type-continuation pair), each with different lexical content. This means that comparisons of items were done across subjects. Using the Prolific platform, we recruited 80 participants who were native English speakers with perfect or corrected vision and without language disorders. Acceptability ratings on the same sentences were averaged, and the resulting averages were compared within an item to compute accuracies that could be compared to the language model results.

5.1 Results

Because the subjects of all continuations are definite in LIEDER, we refer to singular definite continuations as singular continuations and plural ones as plural continuations. Discussion of the results will be broken down based on the plurality of the continuations involved in the comparisons.

5.1.1 Singular Continuations

Singular continuations are felicitous after contexts where one and only one relevant DE is introduced (those that satisfy both EXISTENCE and UNIQUENESS), namely pos_neg and neg_pos. On the other hand, they are infelicitous for contexts that are of the type pos_pos (violating UNIQUENESS) and neg_neg (violating EXISTENCE). As a result, if the language models we are examining reach human-level understanding abilities, we would expect that the probabilities associated with the two felicitous cases are greater than the two infelicitous cases respectively, which gives us four comparison types.

The results for these comparisons are shown in Figure 1, respectively in the four panels. As the first two panels in the figure suggest, all models have ceiling or near ceiling performance on disliking singular continuations that follow contexts where no DEs have been introduced (neg_neg). In other words, for a given singular continuation, they have a strong preference for contexts where one and only one DE is introduced over ones where none are introduced. Such preference indicates that all models know EXISTENCE.

In contrast, in the last two panels in Figure 1, we see that model accuracies are lower when the infelicitous context is pos_pos as compared to pos_neg or neg_pos (i.e., two relevant DEs are introduced). This drop in accuracy suggests that the language models do not consider pos_pos to be worse than pos_neg or neg_pos in licensing the same singular definite continuations.

Why might pos_pos contexts be confusing for large language models? In the context of our linguistic requirements on DE introduction and reference, there are two possible answers to this question:

Hypothesis 1: During training, the models have successfully learned the EXISTENCE requirement, but they failed to learn UNIQUENESS.

Hypothesis 2: During training, the models have successfully learned both the EXISTENCE and UNIQUENESS requirements but fail to recognize that two distinct DEs have been introduced in pos_pos contexts, resulting in difficulties in distinguishing the infelicitous pos_pos from felicitous pos_neg and neg_pos. To put it in another way, they fail at the NOVELTY requirement.

At this stage, we lack evidence that supports one hypothesis over another. Experiment 2 focuses on teasing these two hypotheses apart.

From the last two panels in Figure 1, we can see that accuracy for p(sg|neg_pos)>p(sg|pos_pos) is uniformly higher than p(sg|pos_neg)>p(sg|pos_pos) across models, while human performance differs very little. We claim that the source of this contrast is a preference for singular definites in the context of neg_pos over pos_neg, both of which ought to be felicitous contexts. Figure 2 evalulates this claim, illustrating the percentage of test examples where models and humans consider neg_pos to a better context for singular definite continuations than pos_neg. All models except Llama exhibit this preference for over half of the examples. Interestingly and perhaps surprisingly, humans also demonstrate a distance effect, showing a preference for neg_pos over pos_neg contexts for singular definites on a majority of examples. Thus, the presence of DISTANCE sensitivity need not be interpreted to be a deficiency of LM performance, but perhaps a reflection of patterns in human language use. Nonetheless, though humans show a DISTANCE effect, they demonstrate awareness of the UNIQUENESS and NOVELTY requirement, leading to 75% accuracy on p(sg|pos_neg)>p(sg|pos_pos), where no model achieves greater than chance performance.

The presence of this distance effect provides a possible explanation of the failure in systematic behavior that SL observed in their work. A reanalysis of their data (which involved only pos_neg and neg_pos cases) finds that the position of the DE-introducing sentence impacts model accuracy (Figure 3; Est= -0.5102 with neg_pos as the base level; $p = 0.0161$).[3]

[3]We applied a linear mixed-effect model using the lme4 (Bates et al., 2015) library in R with a main effect of DISTANCE and a random effect of ITEM, collapsing across different models. We also examined the effect of DISTANCE with respect to other sentence types, where DISTANCE is also significant.

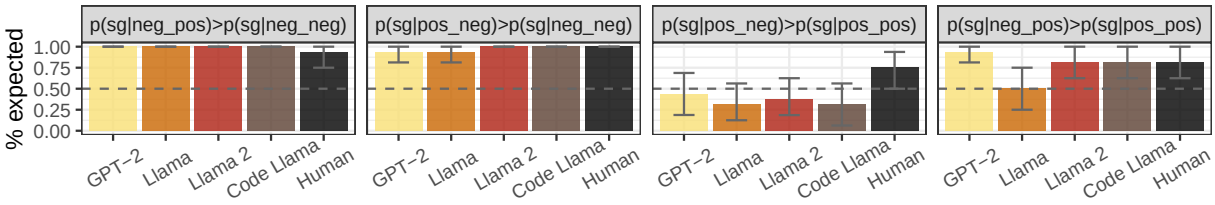

Figure 1: Results for singular continuations by model and comparison type. The dotted lines indicate chance performance and the error bars indicate bootstrapped 95% confidence intervals.

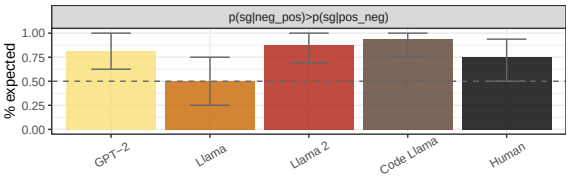

Figure 2: Preference for neg_pos over pos_neg by model.

5.1.2 Plural Continuations

Plural continuations are felicitous only following pos_pos contexts. Hence, we would expect the probability of a plural continuation given pos_pos to be greater than those given pos_neg, neg_pos, and neg_neg.

Results for these three comparisons are shown in Figure 4. Apart from GPT-2, other models exhibit near-ceiling accuracy, which demonstrates that out of the four possible contexts, they consider pos_pos to be the best one prior to plural continuations, consistent with human judgments. The lower accuracy of GPT-2 compared to Llama, Llama 2, and Code Llama could be attributed to GPT-2 having fewer model parameters and less training data. Regardless, the models' ceiling performance on plural continuations serves as evidence that they have learned both the existence and the plurality requirements for plural definite descriptions.

5.1.3 Comparing Singluar and Plural Continuations

We finally compare across singular and plural continuations. There are 8 such comparison types in total. We will focus on three of them that are particularly informative, which are shown in Figure 5.

For p(sg|neg_pos)>p(pl|neg_pos) and p(sg|pos_neg)>p(pl|pos_neg), all of the LLMs achieve near-ceiling accuracy, consistent with human judgments. The high performance on these two comparisons suggests a preference for singular continuations over plural ones when one and only one relevant entity has been introduced into the discourse using a singular definite description. This preference reflects models' knowledge of UNIQUENESS, which speaks against **Hypothesis 1** that we proposed in Section 5.1.1. If the models know UNIQUENESS, why do they perform at chance for the comparison p(sg|pos_neg)>p(sg|pos_pos) in Figure 1? The only possibility left is **Hypothesis 2**: they take a singular continuation following a pos_pos context to be possible because they do not recognize that two distinct DEs have been introduced.

p(pl|pos_pos)>p(sg|pos_pos) provides further support for pos_pos being a difficult context for the models. We saw above that all models consider pos_pos to be the best context preceding plural continuations. Hence, the problem must be that somehow the models assign an incorrectly high probability to p(sg|pos_pos) because of failure to enforce the NOVELTY condition with two indefinites. We test this hypothesis further in Experiment 2.

6 Experiment 2: Facilitating Novelty

As described in Section 4, pos_pos contexts introduce two DEs using two instances of the same indefinite description. For example, in the context sentence *John owns a dog and Mark owns a dog*, two dogs have been introduced using the same indefinite description *a dog*. However, language models might have difficulty understanding that these are two different dogs because not only do they need to recognize DEs through distinct occurrences of the indefinite description *a dog*, but they must also know the NOVELTY condition to consider these distinct occurrences as distinct DEs.

6.1 Dataset

One way to test if the LLMs do fail to recognize two different DEs in pos_pos contexts is to make

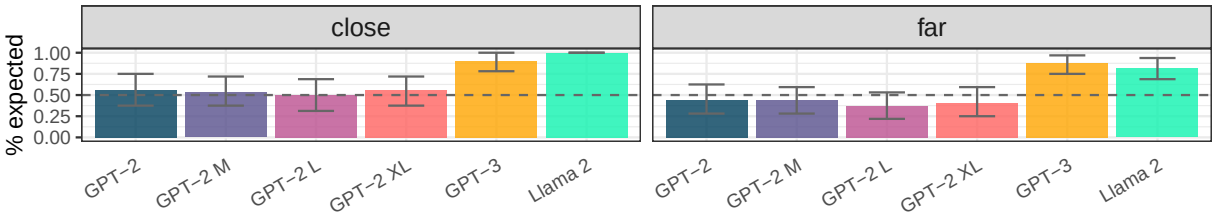

Figure 3: Decomposition of results for *affirmative-negation* type sentences in Schuster and Linzen (2022) by DISTANCE. Data for GPT-2, GPT-2 M, GPT-2 L, GPT-2 XL, and GPT-3 are retrieved from their GitHub Repository.

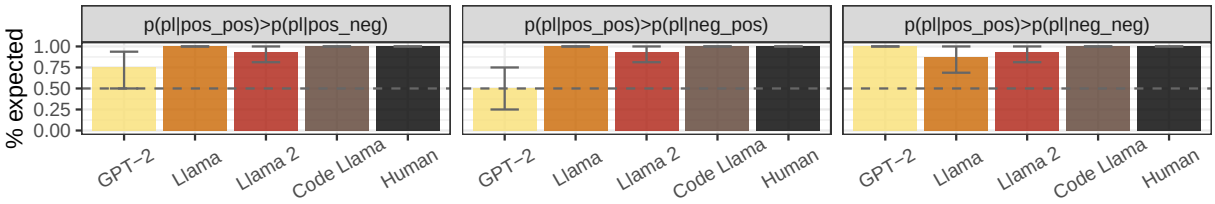

Figure 4: Results for plural comparisons by model and comparison type.

it explicit using lexical cues that the first entity is distinct from the second entity. Then, we can see if the performance relating to pos_pos contexts has increased. If the accuracy does increase when the distinction is explicit, then there is evidence that the LLMs fail to recognize the distinction in the implicit case. Accordingly, we make the following modification to our dataset: for each context sentence of the type pos_pos, we add the adjective "different" to the second indefinite description. All other context sentences and continuations are kept the same.

(3) a. John owns a dog and Mark owns a dog.

b. John owns a dog and Mark owns a different dog.

As an example, (3a) becomes (3b) in the updated dataset. We will use *Implicit* to refer to the original dataset and *Explicit* to refer to the updated dataset in the rest of this paper.

6.2 Results

Results for the resulting felicity comparisons are shown in Figure 6.

We applied a linear mixed-effect model with two fixed effect predictors: VERSION (a categorical variable with two levels: *Implicit* or *Explicit*) and COMPARISON TYPE, and ITEM as a random effect. This model gives a significant coefficient for VERSION (p = 0.017) indicating that accuracy increases significantly from *Implicit* to *Explicit*. This increase supports **Hypothesis 2**, i.e., difficulty with the NOVELTY condition. When the distinctness of the two indefinites is made lexically explicit, the models can better recognize that two distinct DEs are introduced in pos_pos contexts.

7 Discussion

In Experiment 1, we saw that all four transformers, GPT-2, Llama, Llama 2, and Code Llama, displayed clear knowledge of EXISTENCE and PLURALITY. They inarguably consider singular continuations to be bad after contexts that introduce no DEs and plural continuations to be good only when more than one DE has been introduced. However, they failed to show clear knowledge of UNIQUENESS, as indicated by their at-chance performance for the comparison type p(sg|pos_neg)>p(sg|pos_pos). We believe that such underperformance is not due to a lack of knowledge of UNIQUENESS because the results in Figure 5 support knowledge of the UNIQUENESS requirement. Instead, we believe that the problem lies in NOVELTY. Specifically, the models cannot identify that pos_pos contexts introduce two distinct DEs.

We suspect that this context might be particularly challenging because the two DEs are introduced using the same indefinite phrase (such as the two occurrences of *a dog* in *John owns a dog and Mark owns a dog too*). Hence, if the strategy that a model adopts for recognizing DEs relies heavily on distinct lexical cues, then pos_pos is the exact case where they might fail. One might imagine that it is often sufficient to just pay attention to

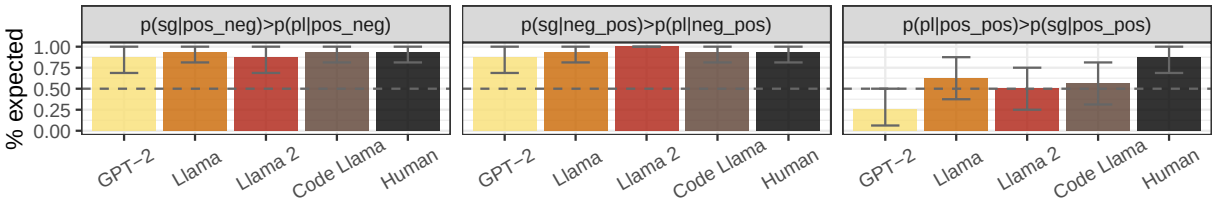

Figure 5: Results by model and comparison type for comparisons across singular and plural continuations.

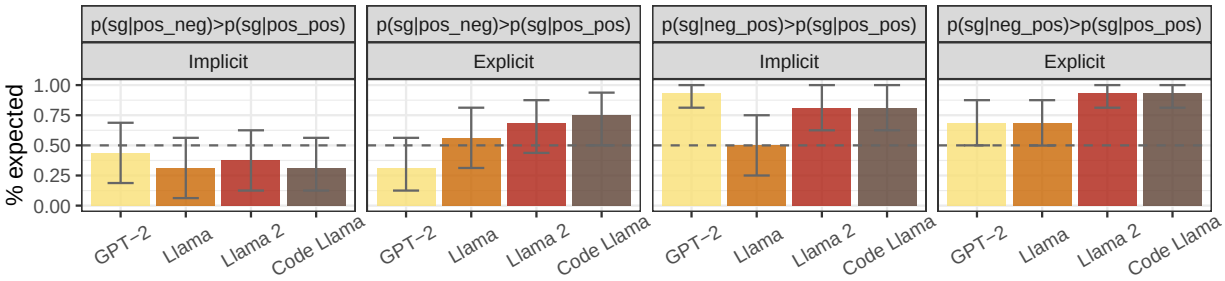

Figure 6: Experiment 2 results by model, version, and comparison type for singular continuations.

unique nouns for DE recognition. For example, given the sentence *John owns a cat and a dog*, the information that *a cat* and *a dog* are two distinct DEs are lexically encoded. However, such information is not available in our pos_pos contexts, and thus models must know NOVELTY in order to excel. Indeed, supplying explicit information about distinctiveness does increase the models' performance regarding UNIQUENESS, as shown in Experiment 2.

Taken together, these results all point to language models' mastery of EXISTENCE, UNIQUENESS, and PLURALITY and a lack of knowledge on NOVELTY when minimal information is given.

Distance Effect Results from Experiment 1 also show a clear effect of DISTANCE. In our reanalysis of SL, LMs did a better job of recognizing DEs when the entities are introduced closer to the continuation that refers to them. Within results from our own evaluation suite, there is also a clear preference for introducing DEs closer to the definite description when the singular continuations are equally felicitous. It seems as if as the sentence unfolds, the status of a DE as one that can be referred to gradually decreases. In other words, there is a greater cost associated with referring to a DE that is introduced earlier in time than those that are introduced later.

8 Conclusion

In this paper, we proposed the Linguistically-Informed Evaluation for Discourse Entity Recognition (LIEDER) dataset. Our paradigm allows for a detailed examination of language models' knowledge of four properties of definite and indefinite NPs that language models ought to know in order to successfully coordinate the introduction and reference of DEs. These properties are EXISTENCE, UNIQUENESS, PLURALITY, and NOVELTY. We demonstrated that despite mastering EXISTENCE, UNIQUENESS, and PLURALITY, the open-source large language models that we examined lack understanding of the NOVELTY requirement, which is a novel discovery. Despite such deficiency, we showed that language models GPT-2, GPT-3, Llama, Llama 2, and Code Llama all successfully captured human preference of referring to DEs that are introduced closer to their reference point, which we call an effect of DISTANCE. On a broader level, although modern neural language models are becoming less reliant on linguistics theories than previous rule-based systems, the success of LIEDER in helping to identify language models' deficiency in DE recognition highlights the importance of linguistic considerations in evaluating the performance of modern language models.

9 Ethics Statement

9.1 Limitations

The current study only focuses on English, which has overt determiners for indefinite and definite NPs. There are other languages that do not have determiners equivalent to *a* and *the* in English. For example, Mandarin Chinese makes use of demonstratives that can serve similar purposes as English determiners. Our evaluation paradigm is thus limited in that it cannot be directly used to evaluate DE recognition on language models trained on other languages without considering language-specific properties relating to DE introduction and reference.

9.2 Risks

All participants in the human experiment were recruited through Prolific under the approval of the Yale University IRB. At the beginning of the experiment, they were presented with consent forms that indicated the potential risks associated with participation, and only those who consented were allowed to proceed with the task. The risk was minimal. All identifier information has been removed from the data to guarantee anonymity. Participants received compensation equivalent to $12/hr, which is around 70% higher than the federal minimum wage.

References

Douglas Bates, Martin Mächler, Ben Bolker, and Steve Walker. 2015. Fitting linear mixed-effects models using lme4. *Journal of Statistical Software*, 67(1):1–48.

Irene Heim and Angelika Kratzer. 1998. Semantics in generative grammar.

Irene Roswitha Heim. 1982. *The semantics of definite and indefinite noun phrases*. University of Massachusetts Amherst.

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.

Fred Landman. 1989. Groups, i. *Linguistics and philosophy*, pages 559–605.

Jing Li, Aixin Sun, Jianglei Han, and Chenliang Li. 2022. A survey on deep learning for named entity recognition. *IEEE Transactions on Knowledge and Data Engineering*, 34(1):50–70.

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

Baptiste Rozière, Jonas Gehring, Fabian Gloeckle, Sten Sootla, Itai Gat, Xiaoqing Ellen Tan, Yossi Adi, Jingyu Liu, Romain Sauvestre, Tal Remez, Jérémy Rapin, Artyom Kozhevnikov, Ivan Evtimov, Joanna Bitton, Manish Bhatt, Cristian Canton Ferrer, Aaron Grattafiori, Wenhan Xiong, Alexandre Défossez, Jade Copet, Faisal Azhar, Hugo Touvron, Louis Martin, Nicolas Usunier, Thomas Scialom, and Gabriel Synnaeve. 2024. Code llama: Open foundation models for code.

Bertrand Russell. 1905. On denoting. *Mind*, 14:479–493.

Sebastian Schuster and Tal Linzen. 2022. When a sentence does not introduce a discourse entity, transformer-based models still sometimes refer to it. In *Proceedings of the 2022 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 969–982, Seattle, United States. Association for Computational Linguistics.

Hugo Touvron, Thibaut Lavril, Gautier Izacard, Xavier Martinet, Marie-Anne Lachaux, Timothée Lacroix, Baptiste Rozière, Naman Goyal, Eric Hambro, Faisal Azhar, et al. 2023a. Llama: Open and efficient foundation language models. *arXiv preprint arXiv:2302.13971*.

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

Thomas Wolf, Lysandre Debut, Victor Sanh, Julien Chaumond, Clement Delangue, Anthony Moi, Pierric Cistac, Tim Rault, Rémi Louf, Morgan Funtowicz, et al. 2019. Huggingface's transformers: State-of-the-art natural language processing. *arXiv preprint arXiv:1910.03771*.