THE CONTEXT COUNTS: CHARACTERIZING LLM COUNTING FAILURES ACROSS INPUT REGIMES

Anonymous authors
Paper under double-blind review

ABSTRACT

Counting is an essential skill underpinning complex forms of quantitative reasoning, yet large language models (LLMs) remain unreliable even on seemingly simple counting tasks. Prior work has shown that contextual cues and training-frequency biases affect counting performance. Building on this literature, we characterize not only how often LLMs miscount, but also the magnitude and direction of their errors across two regimes: non-contextual counting (tallying items in line-separated lists) and contextual counting (a filter-then-count operation over unstructured text passages). Our experiments reveal three findings: (1) miscount rates rise with target count, although most errors are relatively small and undercounting predominates; (2) greater list diversity improves performance, even when the added variation comes from case perturbations on a fraction of instances, interspersed sporadically in an otherwise homogeneous list; and (3) similar miscount rates across degrees of context corruption conceal distinct error patterns: GPT 5.2 makes larger relative errors on partially corrupted than fully corrupted contexts, while Claude Opus 4.5 begins to refuse to perform the counting task on contexts with corruption rates as low as 25%. Together, these results show how changes to input structure affect counting errors in ways that miscount rates alone do not capture. We hope this behavioral account provides a basis for future mechanistic studies of LLM counting.

1 INTRODUCTION

Counting provides an obvious testbed for evaluating numerical understanding in large language models (LLMs). In human cognition, the ability to enumerate discrete quantities arises well before formal schooling or linguistic proficiency. Classic studies in developmental psychology (Gelman & Gallistel, 2022; Wynn, 1990; Le Corre & Carey, 2007; Sarnecka & Carey, 2008) have shown that children acquire numerical competence through a sequence of conceptual milestones: learning that numbers map to sets of objects, that these mappings follow an ordered and exhaustive pattern, and that the final number in a count represents the total cardinality of the set. By roughly age five, most children understand these principles robustly, making counting one of the earliest examples of conceptual abstraction in human cognition. Adapting such developmental paradigms provides a principled lens for identifying the cognitive abstractions or representations responsible for the downstream task performance of LLMs (Frank, 2023).

In contrast, it remains unclear whether LLMs acquire an analogous understanding of counting. Despite their impressive performance across a wide range of language tasks, LLMs are known to struggle with systematic, rule-governed computation. Counting, while seemingly simple, exposes this limitation acutely. It demands that a model track discrete quantities, maintain state across a sequence, and produce exact numerical outputs: capabilities that do not obviously emerge from next-token prediction on natural language. Understanding what kinds of counting failures occur and the input conditions under which they emerge in these models is therefore both a scientifically rich and practically important problem.

In this paper, we present a systematic characterization of counting failures, organized across two structurally distinct counting regimes. In the first regime (non-contextual counting), we study counting failures that occur when counting the number of entities in a list. In the second regime (contextual counting), we examine failures that arise in a more complex setting; counting occurrences

Reviewers: please read the Reviewer Guidelines (iclr.cc/Conferences/2027/ReviewerGuidelines) and the AI Policy for Reviewers (iclr.cc/Conferences/2027/AIPolicyForReviewers). If you used AI to expand, edit, or polish your review, please provide the input text to the LLM. Better still, consider skipping the LLM and submitting your original text: we, and the authors, are much more interested in your unedited thoughts than in what an LLM has to say. AI-assisted or not, you are putting your name and reputation behind your review: LLM-generated falsehoods, hallucinations or misrepresentations are subject to disciplinary action, which may include desk-rejecting all papers you have authored.

of a single target entity embedded within some broader linguistic or narrative context, and requires understanding of string matching operations. We present detailed analyses and interventions characterizing LLM counting behavior and their robustness to input ablations.

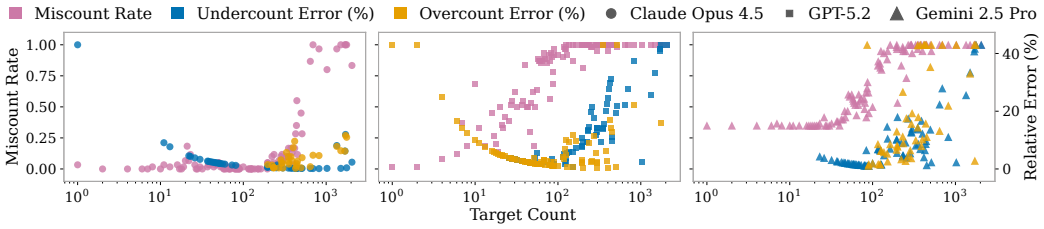


Figure 1: **Miscount rate and error magnitudes across models.** Panels show how errors vary across target counts for Claude Opus 4.5, GPT 5.2, and Gemini 2.5 Pro on the non-contextual counting task. Pink markers show the proportion incorrect model counts (left axis), whereas blue and orange markers show the median relative errors among undercounts and overcounts, respectively (right axis). Lower values indicate better performance for both metrics. Error magnitudes above the pooled 95th percentile are clipped for display. Target count is plotted on a logarithmic scale. Note that at higher counts, while the average miscount rates approach 1, a majority of the miscounts are within 20% of the expected count.

Prior work has primarily relied on exact-match metrics to evaluate counting performance and how factors such as input length, semantic content, tokenization, prompt phrasing affect it. Others have examined isolated aspects of counting behavior such as sensitivity to scale, input composition, and error direction. Our work sits at their intersection, providing a unified evaluation framework that characterizes counting failures by their frequency, magnitude, and direction across distinct input regimes and varying degrees of perturbation. Our experiments and analysis reveal three key findings:

1. Exact-match metrics obscure the nuances of counting failures. At high target counts, models are less likely to produce exact matches, despite the generated counts being numerically close to the target count, in general (Figure 1). Exact-match metrics, therefore, provide an incomplete view of counting behavior by treating near misses and large errors identically, necessitating more composite metrics that can uncover systematic error patterns, such as models' tendency to undercount than overcount when miscounts occur.
2. List composition significantly affects counting performance. For a fixed target count, models make more errors when counting repetitions of a single entity than when counting heterogeneous entities from the same semantic category. For example, models are more likely to miscount a list containing 68 repetitions of "apple" than a list containing 68 fruits.
3. When the target entity is embedded within natural context, counting performance exhibits sensitivity to contextual interference. Progressive corruption of the surrounding context shows that partially corrupted passages produce higher error rates than either uncorrupted or fully corrupted passages, revealing a non-monotonic relationship between contextual coherence and counting performance.

Our findings provide a detailed behavioral account of counting in LLMs and highlight the importance of evaluating not only whether models count correctly, but also how their errors vary with magnitude, input composition, and contextual structure. They have direct implications for the design, evaluation, and deployment of LLMs in settings that require precise quantitative reasoning.

2 Related Work

Prior studies have documented systematic weaknesses of LLMs in tasks requiring arithmetic, numerical consistency, and magnitude comparisons (Wallace et al., 2019; Geva et al., 2020; Shah et al., 2023). Other works have also shown how counting accuracy mirrors the inductive biases in the training corpus (McCoy et al., 2023; Razeghi et al., 2022). Even specialized models with mathematical knowledge rely on heuristic patterns rather than robust numerical reasoning (Cobbe et al.,

2021; Lewkowycz et al., 2022). However, eliciting intermediate computation, with *scratchpads* or *chain-of-thought* prompting (Nye et al., 2021; Wei et al., 2022; Kojima et al., 2022), has been shown to substantially improve performance on algorithmic tasks like simple word-counting (Xu & Ma (2025)). Counting performance has also been shown to be sensitive to a variety of other factors: query phrasing and input parameter population (Ball et al., 2024), repeated symbols (Fu et al., 2024; Qian et al., 2023; Barbero et al., 2024; Wang et al., 2025), tokenization (Zhang et al., 2024), context length (Song et al., 2025; Hsieh et al., 2024; Liu et al., 2024; Levy et al., 2024; Shi et al., 2023).

Counting tasks also provide a valuable basis for evaluating the formal language expressivity and circuit complexity of transformer architectures (Hahn, 2020; Bhattamishra et al., 2020; Merrill & Sabharwal, 2023; Yang & Chiang, 2024; Strobl et al., 2024). Vocabulary size, embedding dimension, and context lengths have been shown to influence performance (Yehudai et al., 2024; Chang & Bisk, 2025) in the theoretical setting. Numerous mechanistic studies on this topic have explored transformer circuits that respond to pattern-matching, embedding size, vocabulary size, context length, and positional embeddings (Behrens et al., 2025; Golkar et al., 2024; Zsámboki et al., 2025; Hasani et al., 2026) when performing numerical tasks.

Although prior work has yielded valuable insights into LLM counting, evaluation remains largely limited to exact-match accuracy, obscuring how failure patterns vary across input structures. We address this gap by systematically characterizing counting behavior under diverse input conditions along multiple error dimensions, including frequency, direction, and relative magnitude.

3 Experimental Setup

3.1 Counting Regimes

We characterize LLM counting behavior across two regimes, designed to isolate the effects of target diversity, input structure, and contextual interference on LLM counting failures. They require different, yet overlapping counting skills and operate on different data sources as described below.

Non-contextual counting. In this setting, models are instructed to count the total number of items in a list of T_c line-separated occurrences of one or more distinct entities. We construct a dataset with 1800 samples that is generated from six manually identified lexical categories with 10 distinct entities each, and 30 target counts sampled from the $[1, 2048]$ range.

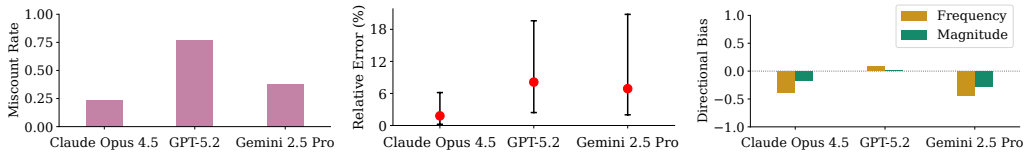
Contextual counting. This setting is designed to evaluate how models count entities naturally occurring within some context. We sample $N = 300$ nouns uniformly at random from the WordNet (Miller, 1992) lexical database and use Wikipedia's Python API to extract articles with titles containing the target noun. If one or more matches are identified, we select one at random to obtain $N = 156$ passages containing at least one instance of the target string that it was matched on. We programmatically count the number of occurrences of the T_s in the corresponding article. Only occurrences of the T_s with no inflectional forms and exact matches with leading or trailing punctuations/whitespaces count towards the final T_c. The counting task is then presented to the model in a three-shot setting with chain-of-thought prompting, alongside comprehensive instructions detailing the task. This setting is designed to identify counting errors induced by the contextual interference and surface form variations of the T_s.

3.2 Models

We conduct experiments on the following models: Gemini 2.5 Pro (Comanici et al., 2025), GPT 5.2 (OpenAI, 2026), and Claude Opus 4.5 (Anthropic, 2025) using OpenRouter (2025) and an internal organization-specific API. All our experiments are conducted with medium reasoning effort, unless stated otherwise. We verify that API requests do not leverage external tools to perform counting tasks. The prompts used for each task setting can be found in Tables 3 and 5 in Appendix A.

3.3 Evaluation Metrics

We conceptualize counting errors on an additive scale, under which, deviations of k above or below a target count (y) (in other words, a model prediction of $y + k$ or $y - k$) are treated as errors of equal magnitude but opposite direction. However, a consequence of this design is that miscount deviations



(a) Miscount rate (b) Relative errors (c) Directional bias

Figure 2: **Evaluating non-contextual counting tasks across models.** (a) Miscount rate: fraction of trials where the model's predicted count differs from the true count. (b) Relative error: Median relative errors (%) computed over miscounts only, with bands representing the IQR (25th–75th percentile); groups with no miscounts are omitted. (c) Directional asymmetry of errors: frequency bias ($\mathrm{M_{FB}}$) and magnitude bias ($\mathrm{M_{MB}}$, green) measure the directional preferences in counting failures. Diverging bars indicate the directional discrepancy between the frequency and strength of the miscounts.

are bounded when the model undercounts (we only consider non-negative target and model counts), and unbounded when it overcounts, rendering it a skewed evaluation metric. However, we argue that this asymmetry is inherent to counting: a model can only undercount by at most 100% but can overcount by an arbitrarily large amount. We therefore use a multi-dimensional evaluation framework to capture the frequency, relative magnitude, and directional asymmetry of counting errors.

Metrics. Let $\mathcal{Y} = \{(y_i, \hat{y}_i)\}_{i=1}^{\mathrm{N}}$ denote a set of N observations requiring the model to produce a target count $y_i > 0$ and $\hat{y}_i \geq 0$ is the model-generated count. Then, let $e_i = \hat{y}_i - y_i$ be the signed error and the subsets of miscounting, overcounting, and undercounting predictions as

$$\mathcal{Y}^{\neq} = \{(y_i, \hat{y}_i) \in \mathcal{Y} : \hat{y}_i \neq y_i\}, \quad \mathcal{Y}^{+} = \{(y_i, \hat{y}_i) \in \mathcal{Y} : \hat{y}_i > y_i\}, \quad \mathcal{Y}^{-} = \{(y_i, \hat{y}_i) \in \mathcal{Y} : \hat{y}_i < y_i\},$$

respectively. We evaluate counting performance using

$$\mathrm{M_{rate}}(\mathcal{Y}) = \frac{|\mathcal{Y}^{\neq}|}{|\mathcal{Y}|}, \tag{1}$$

$$\mathrm{M_{RE}}(\mathcal{Y}) = \underset{(y_i,\hat{y}_i)\in\mathcal{Y}^{\neq}}{\mathrm{median}} \left(\frac{100 \times |e_i|}{y_i} \right), \qquad \text{if } |\mathcal{Y}^{\neq}| \neq 0, \tag{2}$$

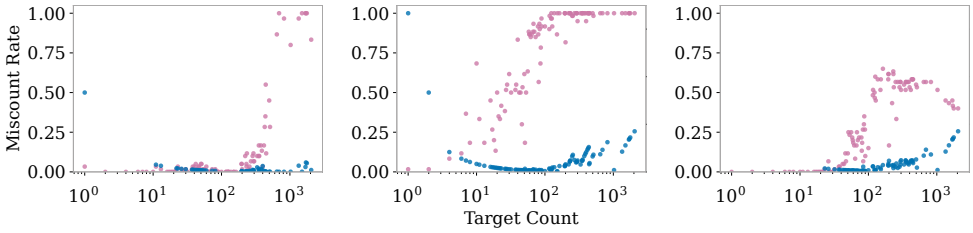
$$\mathrm{M_{FAI}}(\mathcal{Y}) = \frac{|\mathcal{Y}^{+}| - |\mathcal{Y}^{-}|}{|\mathcal{Y}^{+}| + |\mathcal{Y}^{-}|}, \qquad \text{if } |\mathcal{Y}^{\neq}| \neq 0, \tag{3}$$

$$\mathrm{M_{MAI}}(\mathcal{Y}) = \frac{\mathrm{M_{RE}}(\mathcal{Y}^{+}) - \mathrm{M_{RE}}(\mathcal{Y}^{-})}{\mathrm{M_{RE}}(\mathcal{Y}^{+}) + \mathrm{M_{RE}}(\mathcal{Y}^{-})}, \qquad \text{if } |\mathcal{Y}^{\neq}| \neq 0. \tag{4}$$

For $s \in \{+, -\}$, we define the direction-conditional relative error as

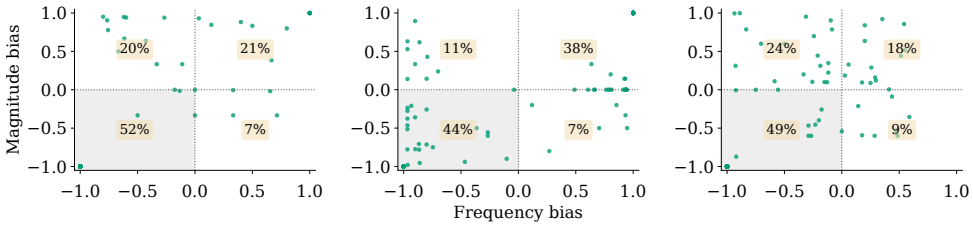
$$\mathrm{M_{RE}}(\mathcal{Y}^{s}) = \begin{cases} \underset{(y_i,\hat{y}_i)\in\mathcal{Y}^{s}}{\mathrm{median}} \left(\frac{100 \times |e_i|}{y_i} \right), & \text{if } |\mathcal{Y}^{s}| > 0, \\ 0, & \text{if } |\mathcal{Y}^{s}| = 0. \end{cases}$$

Here, miscount rates ($\mathrm{M_{rate}}$) measures the fraction of miscounts, while relative miscount errors $\mathrm{M_{RE}}$ measures the relative error from the target magnitude (in %). Miscount frequency bias ($\mathrm{M_{FB}}$) and miscount magnitude bias ($\mathrm{M_{MB}}$) capture the directional asymmetry in error frequency and magnitude, respectively, and range from -1 to 1, with negative values indicating an undercounting bias and positive values indicating an overcounting bias. For $\mathrm{M_{MB}}$, -1 and 1 indicate exclusively undercounting and overcounting errors, respectively. Both are undefined when no miscounts occur.



(a) Claude Opus 4.5 (b) GPT 5.2 (c) Gemini 2.5 Pro

Figure 3: **Miscount rates and relative errors by model.** Pink points (left axis) indicate the proportion of incorrect counts at each target count and blue points (right axis) indicate the median relative error among miscounts. Target count is logarithmically scaled.



(a) Claude Opus 4.5 (b) GPT 5.2 (c) Gemini 2.5 Pro

Figure 4: **Frequency vs. magnitude bias across models.** The x-axis (frequency bias) measures *how often* undercounts dominate, while the y-axis (magnitude bias) measures *how much* they dominate. Each dot represents a target count (that was miscounted by the model at least once) and the shaded area indicates the dominant directional preferences for the model. Percentages denote the share of target counts per quadrant (excluding counts with errors in only one direction).

4 Results

4.1 Non-contextual counting

Figure 2 summarizes the performance of the three models on the non-contextual counting task. Claude Opus 4.5 is the most likely to produce accurate counts, followed by Gemini 2.5 Pro, and then GPT 5.2 (Figure 2a). Miscount rates (Figure 3) rise sharply with the target count, reaching nearly 100% for GPT-5.2 and Claude Opus 4.5, while relative errors remain low for most counts and increase only modestly, remaining within 20% of the target count.

4.2 Contextual Counting

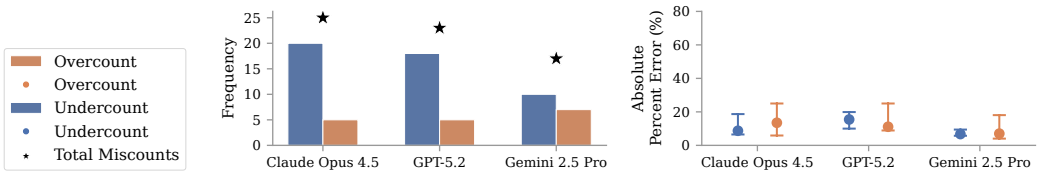
Figure 5 summarizes model performance on the contextual counting task. Undercounting still prevails as the dominant failure mode (Figure 5a). However, the relative error distributions are much closer across models and miscount direction (Figure 5b). Gemini 2.5 Pro outperforms the other two models for this task, with Claude Opus 4.5 performing worst, sharply contrasting its performance on the non-contextual counting task.

5 Input Interactions

Model performance in counting tasks often relies on context-specific cues. We perturb the target and its surrounding context separately to examine how their interaction affects counting performance.

5.1 List Composition

Our experiments evaluating LLM counting behavior under the non-contextual regime revealed some valuable insights. However, the input mode was strictly limited to repeated instances of a single-



(a) Miscount rate (b) Miscount magnitude

Figure 5: **Miscount frequency and magnitude for contextual counting.** *(a)* Frequencies of undercounting and overcounting errors, with black stars indicating the total number of miscounts for each model. When counting entities within a context, Claude Opus 4.5 miscounts most frequently with undercounting being a more common failure mode than overcounting for all three models. *(b)* Median absolute percentage error, reported separately for undercounts and overcounts; vertical intervals indicate the interquartile range (25th–75th percentiles). The error magnitudes for all three models stay roughly the same, with Gemini 2.5 Pro producing the least absolute errors.

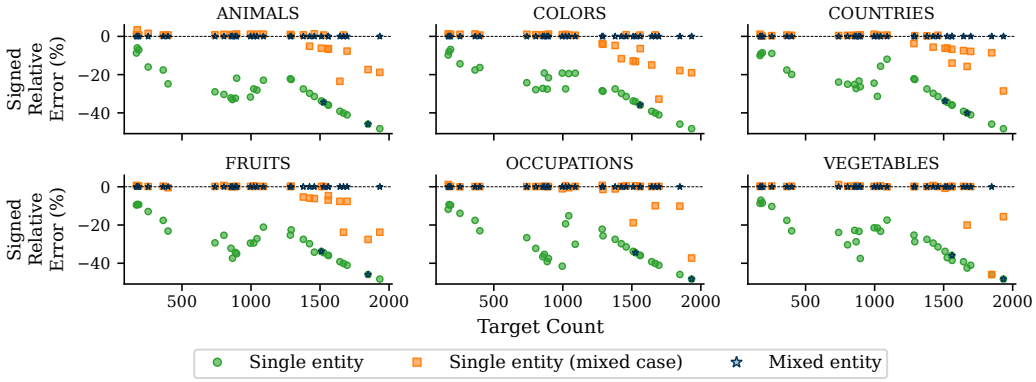


Figure 6: **List composition effects.** In GPT 5.2, lists comprising of single entity repetitions produces larger deviations from the target count than lists with more diverse items.

entity, which raises an important question: *"Do error patterns change with list constitution?"*. To answer this question, we construct a comprehensive dataset of nouns generated by ChatGPT-5.4 (OpenAI, 2026) (Appendix A) from six manually identified and semantically distinct groups (animals, colors, countries, fruits, occupations, and vegetables). We sample 30 target counts under 2000, uniformly at random, without replacement. We then design a paired variation of the non-contextual counting experiment, where the inputs are matched on the target count, with the only distinguishing feature between them being the number of unique entities presented. Specifically, for a given target count (n), the paired non-contextual counting inputs are: (1) *single-entity* lists, consisting of a singular entity repeated n times; and (2) *multi-entity* lists, drawn from the generated list of entities. We find that list composition substantially affects performance, with multi-entity lists being less likely to be miscounted than single-entity lists, consistent with the literature (Zhang et al., 2024; Ríos-García et al., 2026). These findings led us to a slightly different, yet related question: *"Can minimal superficial differences also enhance performance?"*. We answer this by introducing case variations (e.g. ('apple', 'apPLe', 'ApPlE')) to 20% of the entities at random positions within an otherwise identical list of items. This experiment revealed that case variations in a subset of entities also led to a drop in miscount rates and median relative errors across categories (Figure 6). The effects are consistent across models (Figures 12 and 13), indicating that superficial case variations can also improve performance in non-contextual counting.

String type	Description	Example
Nonce	Lexically plausible but semantically meaningless strings.	*prepotist*
Cipher	Character-level substitution using a cipher-mapping.	*dldgefxad*
Random hash	Random hexadecimal sequence.	*7385a0c64*

Table 1: **Target entity ablations.** Target entities are drawn from WordNet, and we apply ablations matched on the string length. Cipher-mapping is obtained by mapping the English alphabet onto a random permutation of itself. The example strings correspond to the target entity *"paperclip"*.

Original Target	Cipher-Ablated Target	Context Ablation Rate	Cipher-Ablated Context
bear	hgle	0%	*[TARGET]s are carnivoran mammals of the family ursidae ..*
		10%	*[TARGET]s are carnivoran mammals of the family* ***retailg*** *..*
		25%	*[TARGET]s are* ***flenawqeln*** *mammals of the family* ***retailg*** *..*
		50%	*[TARGET]s are* ***flenawqeln ylyylxt qb pog*** *family* ***retailg*** *..*
		75%	*[TARGET]s are* ***flenawqeln ylyylxt qb pog blyaxv retailg*** *..*
		90%	*[TARGET]s* ***leg flenawqeln ylyylxt qb pog blyaxv retailg*** *..*
		100%	*[TARGET]****t leg flenawqeln ylyylxt qb pog blyaxv retailg*** *..*

Table 2: **Context ablation at different cipher rates.** Increasing the context ablation rate incrementally replaces words in the original context with their corresponding ciphered version.

5.2 Contextual Interference

To examine which features of the input text passage influence contextual counting, we systematically perturb the target entity and its surrounding context separately while preserving the number and positions of target occurrences. We apply perturbations along the following dimensions.

Target Entity. Table 1 presents the ablation functions applied on the target noun entity. All exact matches of the target string are replaced with the ablated entity.

Input Context. We apply a character-level cipher mapping of the English alphabet to the input context in which the target string appears, while retaining any substrings matching the target entity. We incrementally cipher $0\%, 10\%, 25\%, 50\%, 75\%, 90\%$, and 100% of the context words and evaluate both original and ciphered targets (Table 2).

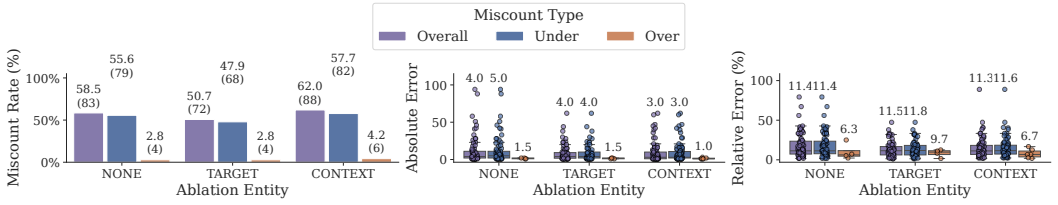


Figure 7: **Target vs. context ablation (GPT 5.2).** Context and target ablations produce relatively minor differences in miscount rate, error magnitude and relative error compared to no ablation.

We find that robustness to input perturbations varies across models. For example, security guardrails on the Claude Opus 4.5 model lead to an overwhelming number of refusals when 25% or more of the context is corrupted. In contrast, both GPT 5.2 and Gemini 2.5 Pro remain fairly robust to both target and context manipulations, despite their lower performance on the non-contextual single-entity task. Figures 7 and 14 show the impact of target and (total) context ablations on GPT 5.2 and Gemini 2.5 Pro. In GPT 5.2, target ablation leads to a decrease in the frequency of miscounts, whereas context ablation leads to higher miscount rates than no ablation. We also observe a drop in relative error magnitudes with target ablation, whereas context ablation leads to a minor increase

in the relative error magnitudes (though statistically insignificant). In Gemini 2.5 Pro, neither ablations lead to meaningful changes in error distributions. Detailed statistical analysis can be found in Appendix C (Tables 8 to 13).

5.2.1 Target Entity Substitution

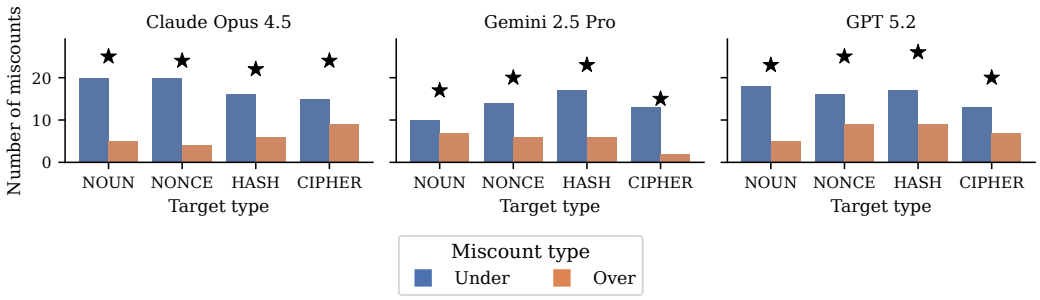


Figure 8: **Effects of target substitution within context.** Replacing the target entity in the contextual counting task does not significantly change the total number of miscounts across models. The stars represent total number of miscounts for the corresponding target entity type.

To identify potential target-context entanglement effects on counting behavior, we present the model with Wikipedia articles within which T_s occurs (as described in Section 3.1). We control for context-target interactions by repeating the same experiment wherein the T_s is replaced by each of its ablated versions described in Table 1. Paired statistical analysis suggests that both Claude Opus 4.5 and Gemini 2.5 Pro do not exhibit significant changes in miscount directionality regardless of the target entity's ablation type. However, GPT 5.2 exhibits a statistically significant drop in the number of miscounts for the cipher ablation type.

5.2.2 Context Degradation

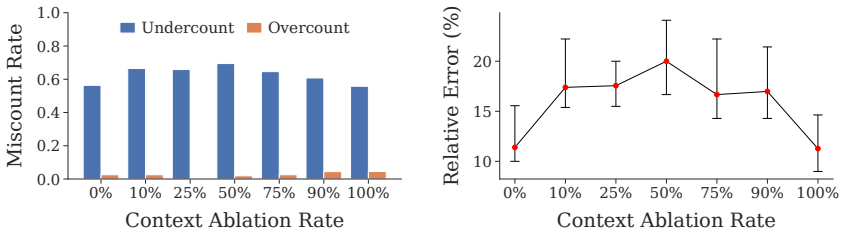


Figure 9: **Progressive context degradation results (GPT 5.2).** Progressively replacing $x\%$ of words in the context leads to a non-monotonic change in performance, with relative errors peaking at 50%. Compared to the clean context, no significant differences in relative error was observed after total context corruption.

Figure 9 shows the impact of progressive context degradation on the counting performance for GPT 5.2. We find that the miscount rates are modestly higher when the ablation is partial (that is, not 0% or 100%) and the median $\mathrm{M_{RE}}$ is significantly higher for partial context ablation than either intact (0%) or fully ciphered (100%) versions with the highest $\mathrm{M_{RE}}$ (median = 13.33, 95% CI: $[8.70, 16.67]$) observed at an ablation rate of 50%. In short, partial context degradation produces more frequent and dispersed errors than either intact or fully ciphered context.

6 Discussion and Limitations

Joint Interpretation of Asymmetry Biases. By analyzing $\mathrm{M_{FB}}$ and $\mathrm{M_{MB}}$ in tandem, we decouple the structural bias of LLM counting errors into independent dimensions of frequency and magnitude on a shared $[-1, 1]$ scale. When aligned (e.g., $\mathrm{M_{FB}} > 0, \mathrm{M_{MB}} > 0$), they expose a systematic

inflation bias in both rate and severity. Conversely, divergence uncovers more nuanced pathologies. For instance, a highly negative $\mathrm{M_{FB}}$ with a near-zero $\mathrm{M_{MB}}$ indicates frequent but low-magnitude undercounting (e.g., missing by exactly one). More critically, an opposing divergence—such as a negative $\mathrm{M_{FB}}$ alongside a highly positive $\mathrm{M_{MB}}$—reveals that while the model undercounts most of the time, its rare overcounting errors are catastrophically large hallucinations. Tracking both indices ensures severe tail-end errors are not obscured by sheer error frequency.

Inflectional forms and context masking. The contextual counting task requires models to count only exact surface-form matches of the target entity. Despite detailed instructions and in-context examples with chain-of-thought reasoning, the presence of inflectional forms was associated with higher miscount rates and relative errors (Figure 15). Masking the surrounding context while preserving the target and its inflectional forms modestly reduced the number of errors from 85 to 73 ($p = 0.073$) and shifted their direction from undercounting toward overcounting ($p < 0.001$). Error magnitudes decreased more often than they increased (59 vs. 30 instances), accompanied by a significant reduction in relative error from 5.53% to 1.26% ($p = 0.023$).

Similarities with human behavior. LLM counting errors resemble patterns in human numerical estimation. Specifically, error magnitude grows with target size (Whalen et al., 1999), large sets are systematically undercounted (Izard & Dehaene, 2008), and numerosity judgments are often attributed to approximate magnitude representations (Feigenson et al., 2004; Marjieh et al., 2025). These parallels raise the possibility that LLMs acquire human-like estimation biases from patterns in human-generated training data.

Limitations. To ensure consistent comparisons across models, we use a fixed prompt template and restrict the non-contextual counting experiments to a single delimiter (the newline character: "\n"). This design isolates performance variations arising from the input distribution rather than prompt formulation. Consequently, our findings characterize model behavior within this controlled evaluation regime and do not establish robustness across prompt phrasings or delimiters. Evaluating such robustness would require substantially expanding an already compute-intensive evaluation of frontier models and is beyond the scope of this work. Evaluating how our results generalize across different prompt distributions and input formats is necessary; however, we leave this systematic analysis to future work.

7 Conclusions

We characterize LLM counting errors across two regimes: non-contextual and contextual counting. While miscounts are more frequent at higher target counts in the non-contextual setting, many incorrect predictions are not that far off from the target count, demonstrating that exact-match metrics alone do not adequately explain LLM counting failures. We find that errors have asymmetric tendencies with a bias towards undercounting, in both frequency and magnitude and are significantly affected by list diversity, whether lexical or semantic, findings consistent with those of past work. Finally, we observe that contextual interference induces more model-specific error patterns, with GPT 5.2 being surprisingly robust to total context corruption but not partial context corruption, Gemini 2.5 Pro outperforming the other two models in the absence of any context corruption but degrading in performance as the context is increasingly corrupted, and Claude Opus 4.5 refusing to perform the task when even a fraction of the context is corrupted. These findings show that LLM counting errors have a behavioral signature that is a function of their frequency, direction, and magnitude. By mapping this behavioral signature across varying context structures, our findings provide a clear empirical foundation to motivate future mechanistic interpretability studies exploring whether and how these failure patterns might be internally represented within models.

AI use statement

In this work, we used generative AI tools to generate synthetic datasets, refine hypotheses, design or provide feedback on research methodology or experiments, implement methods, clean and reformat dataset, support qualitative and thematic data analysis, and interpret results.

We have not used generative AI tools to assist with translation.

Additionally, we used generative AI tools for/to create or modify scientific figures or images, creation of artifacts, summarize or analyse existing literature, discover research topics or identify gaps, brainstorming, sourcing/searching for information, edit a research paper to improve readability, identify relevant literature, and format references. We have reviewed all AI-assisted work. We checked all LLM-generated references and research ideas for potential plagiarism through a manual literature survey, LLM-generated code was verified and tested for correctness by the lead author. We take responsibility for the final content of this work, including text, claims or artifacts produced with the aid of generative AI.

REFERENCES

Anthropic. Claude opus 4.5 system card, 2025. URL https://www.anthropic.com/claude/opus. Available at https://www.anthropic.com.

Thomas Ball, Shuo Chen, and Cormac Herley. Can we count on llms? the fixed-effect fallacy and claims of gpt-4 capabilities, 2024. URL https://arxiv.org/abs/2409.07638.

Federico Barbero, Andrea Banino, Steven Kapturowski, Dharshan Kumaran, João G. M. Araújo, Alex Vitvitskyi, Razvan Pascanu, and Petar Veličković. Transformers need glasses! Information over-squashing in language tasks. In *Advances in Neural Information Processing Systems*, volume 37, 2024. URL https://arxiv.org/abs/2406.04267.

Freya Behrens, Luca Biggio, and Lenka Zdeborová. Counting in small transformers: The delicate interplay between attention and feed-forward layers. In *Proceedings of the 42nd International Conference on Machine Learning*. PMLR, 2025. URL https://arxiv.org/abs/2407.11542.

Satwik Bhattamishra, Kabir Ahuja, and Navin Goyal. On the Ability and Limitations of Transformers to Recognize Formal Languages. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pp. 7096–7116, Online, 2020. Association for Computational Linguistics. doi: 10.18653/v1/2020.emnlp-main.576. URL https://aclanthology.org/2020.emnlp-main.576.

Yingshan Chang and Yonatan Bisk. Language models need inductive biases to count inductively. In *The Thirteenth International Conference on Learning Representations (ICLR)*, 2025. URL https://arxiv.org/abs/2405.20131.

Karl Cobbe, Vineet Kosaraju, Mohammad Bavarian, Mark Chen, Heewoo Jun, Lukasz Kaiser, Matthias Plappert, Jerry Tworek, Jacob Hilton, Reiichiro Nakano, Christopher Hesse, and John Schulman. Training verifiers to solve math word problems. *arXiv preprint arXiv:2110.14168*, 2021.

Gheorghe Comanici, Eric Bieber, Mike Schaekermann, Ice Pasupat, Noveen Sachdeva, Inderjit Dhillon, Marcel Blistein, Ori Ram, Dan Zhang, Evan Rosen, Luke Marris, Sam Petulla, et al. Gemini 2.5: Pushing the Frontier with Advanced Reasoning, Multimodality, Long Context, and Next Generation Agentic Capabilities, December 2025. URL http://arxiv.org/abs/2507.06261. arXiv:2507.06261 [cs.CL].

Lisa Feigenson, Stanislas Dehaene, and Elizabeth Spelke. Core systems of number. *Trends in Cognitive Sciences*, 8(7):307–314, 2004. doi: 10.1016/j.tics.2004.05.002.

Michael C. Frank. Baby steps in evaluating the capacities of large language models. *Nature Reviews Psychology*, 2:451–452, 2023. doi: 10.1038/s44159-023-00211-x.

Tairan Fu, Raquel Ferrando, Javier Conde, Carlos Arriaga, and Pedro Reviriego. Why Do Large Language Models (LLMs) Struggle to Count Letters?, December 2024. URL http://arxiv.org/abs/2412.18626. arXiv:2412.18626 [cs].

Rochel Gelman and Charles R. Gallistel. *The Child's Understanding of Number*. Harvard University Press, Cambridge, MA, 2022. ISBN 978-0-674-03753-3. doi: 10.4159/9780674037533.

Mor Geva, Ankit Gupta, and Jonathan Berant. Injecting Numerical Reasoning Skills into Language Models. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pp. 946–958, Online, 2020. Association for Computational Linguistics. doi: 10.18653/v1/2020.acl-main.89. URL https://www.aclweb.org/anthology/2020.acl-main.89.

Siavash Golkar, Alberto Bietti, Mariel Pettee, Michael Eickenberg, Miles Cranmer, Keiya Hirashima, Geraud Krawezik, Nicholas Lourie, Michael McCabe, Rudy Morel, Ruben Ohana, Liam Holden Parker, Bruno Régaldo-Saint Blancard, Kyunghyun Cho, and Shirley Ho. Contextual Counting: A Mechanistic Study of Transformers on a Quantitative Task, May 2024. URL http://arxiv.org/abs/2406.02585. arXiv:2406.02585 [cs.LG].

Michael Hahn. Theoretical limitations of self-attention in neural sequence models. *Transactions of the Association for Computational Linguistics*, 8:156–171, 2020. doi: 10.1162/tacl_a_00306. URL https://aclanthology.org/2020.tacl-1.11/.

Hosein Hasani, Amirmohammad Izadi, Fatemeh Askari, Mobin Bagherian, Sadegh Mohammadian, Mohammad Izadi, and Mahdieh Soleymani Baghshah. Understanding Counting Mechanisms in Large Language and Vision-Language Models, April 2026. URL http://arxiv.org/abs/2511.17699. arXiv:2511.17699 [cs.CV].

Cheng-Ping Hsieh, Simeng Sun, Samuel Kriman, Shantanu Acharya, Dima Rekesh, Fei Jia, Yang Zhang, and Boris Ginsburg. ÌRULER: What's the Real Context Size of Your Long-Context Language Models? In *Proceedings of the First Conference on Language Modeling (COLM)*, 2024.

Véronique Izard and Stanislas Dehaene. Calibrating the mental number line. *Cognition*, 106(3): 1221–1247, 2008. doi: 10.1016/j.cognition.2007.06.004.

Takeshi Kojima, Shixiang Shane Gu, Machel Reid, Yutaka Matsuo, and Yusuke Iwasawa. Large language models are zero-shot reasoners. In *Advances in Neural Information Processing Systems*, volume 35, pp. 22199–22213, 2022. URL https://arxiv.org/abs/2205.11916.

Mathieu Le Corre and Susan Carey. One, two, three, four, nothing more: An investigation of the conceptual sources of the verbal counting principles. *Cognition*, 105(2):395–438, 2007. doi: 10.1016/j.cognition.2006.10.005.

Mosh Levy, Alon Jacoby, and Yoav Goldberg. Same task, more tokens: The impact of input length on the reasoning performance of large language models. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 15339–15353, Bangkok, Thailand, 2024. Association for Computational Linguistics. doi: 10.18653/v1/2024.acl-long.818. URL https://aclanthology.org/2024.acl-long.818/.

Aitor Lewkowycz, Anders Andreassen, David Dohan, Ethan Dyer, Henryk Michalewski, Vinay Ramasesh, Ambrose Slone, Cem Anil, Imanol Schlag, Theo Gutman-Solo, Yuhuai Wu, Behnam Neyshabur, Guy Gur-Ari, and Vedant Misra. Solving quantitative reasoning problems with language models. *Advances in Neural Information Processing Systems*, 35:3843–3857, 2022.

Nelson F. Liu, Kevin Lin, John Hewitt, Ashwin Paranjape, Michele Bevilacqua, Fabio Petroni, and Percy Liang. Lost in the middle: How language models use long contexts. *Transactions of the Association for Computational Linguistics*, 12:157–173, 2024. doi: 10.1162/tacl_a_00638. URL https://aclanthology.org/2024.tacl-1.9/.

Raja Marjieh, Veniamin Veselovsky, Thomas L. Griffiths, and Ilia Sucholutsky. What is a number, that a large language model may know it?, 2025. URL https://arxiv.org/abs/2502.01540. arXiv:2502.01540.

R. Thomas McCoy, Shunyu Yao, Dan Friedman, Matthew Hardy, and Thomas L. Griffiths. Embers of Autoregression: Understanding Large Language Models Through the Problem They are Trained to Solve, September 2023. URL http://arxiv.org/abs/2309.13638. arXiv:2309.13638 [cs].

William Merrill and Ashish Sabharwal. The parallelism tradeoff: Limitations of log-precision transformers. *Transactions of the Association for Computational Linguistics*, 11:531–545, 2023. doi: 10.1162/tacl_a_00562. URL https://aclanthology.org/2023.tacl-1.31/.

George A. Miller. WordNet: A lexical database for English. In *Speech and Natural Language: Proceedings of a Workshop Held at Harriman, New York, February 23-26, 1992*, 1992. URL https://aclanthology.org/H92-1116/.

Maxwell Nye, Anders Johan Andreassen, Guy Gur-Ari, Henryk Michalewski, Jacob Austin, David Bieber, David Dohan, Aitor Lewkowycz, Maarten Bosma, David Luan, Charles Sutton, and Augustus Odena. Show your work: Scratchpads for intermediate computation with language models, 2021. URL https://arxiv.org/abs/2112.00114. arXiv:2112.00114.

OpenAI. Chatgpt (gpt-5.4). https://openai.com, 2026. Accessed: 2026-02-26.

OpenAI. Introducing GPT-5.2, March 2026. URL https://openai.com/index/introducing-gpt-5-2/.

OpenRouter. OpenRouter: Unified API for LLMs, 2025. URL https://openrouter.ai.

Jing Qian, Hong Wang, Zekun Li, Shiyang Li, and Xifeng Yan. Limitations of language models in arithmetic and symbolic induction. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 9285–9298, Toronto, Canada, 2023. Association for Computational Linguistics. doi: 10.18653/v1/2023.acl-long.516. URL https://aclanthology.org/2023.acl-long.516/.

Yasaman Razeghi, Robert L Logan IV, Matt Gardner, and Sameer Singh. Impact of pretraining term frequencies on few-shot numerical reasoning. In *Findings of the Association for Computational Linguistics: EMNLP 2022*, pp. 840–854, Abu Dhabi, United Arab Emirates, 2022. Association for Computational Linguistics. doi: 10.18653/v1/2022.findings-emnlp.59. URL https://aclanthology.org/2022.findings-emnlp.59/.

Martiño Ríos-García, Nawaf Alampara, and Kevin Maik Jablonka. Semantic Content Determines Algorithmic Performance, January 2026. URL http://arxiv.org/abs/2601.21618. arXiv:2601.21618 [cs.AI].

Barbara W. Sarnecka and Susan Carey. How counting represents number: What children must learn and when they learn it. *Cognition*, 108(3):662–674, 2008. doi: 10.1016/j.cognition.2008.05.007.

Raj Shah, Vijay Marupudi, Reba Koenen, Khushi Bhardwaj, and Sashank Varma. Numeric Magnitude Comparison Effects in Large Language Models. In Anna Rogers, Jordan Boyd-Graber, and Naoaki Okazaki (eds.), *Findings of the Association for Computational Linguistics: ACL 2023*, pp. 6147–6161, Toronto, Canada, July 2023. Association for Computational Linguistics. doi: 10.18653/v1/2023.findings-acl.383. URL https://aclanthology.org/2023.findings-acl.383/.

Freda Shi, Xinyun Chen, Kanishka Misra, Nathan Scales, David Dohan, Ed Chi, Nathanael Schärli, and Denny Zhou. Large language models can be easily distracted by irrelevant context. In *Proceedings of the 40th International Conference on Machine Learning*, volume 202 of *Proceedings of Machine Learning Research*, pp. 31210–31227. PMLR, 2023. URL https://proceedings.mlr.press/v202/shi23a.html.

Mingyang Song, Mao Zheng, and Xuan Luo. Counting-Stars: A Multi-evidence, Position-aware, and Scalable Benchmark for Evaluating Long-Context Large Language Models. In Owen Rambow, Leo Wanner, Marianna Apidianaki, Hend Al-Khalifa, Barbara Di Eugenio, and Steven Schockaert (eds.), *Proceedings of the 31st International Conference on Computational Linguistics*, pp. 3753–3763, Abu Dhabi, UAE, January 2025. Association for Computational Linguistics. URL https://aclanthology.org/2025.coling-main.253/.

Lena Strobl, William Merrill, Gail Weiss, David Chiang, and Dana Angluin. What formal languages can transformers express? A survey. *Transactions of the Association for Computational Linguistics*, 12:543–561, 2024. doi: 10.1162/tacl_a_00663. URL https://aclanthology.org/2024.tacl-1.30/.

Eric Wallace, Yizhong Wang, Sujian Li, Sameer Singh, and Matt Gardner. Do NLP Models Know Numbers? Probing Numeracy in Embeddings, September 2019. URL http://arxiv.org/abs/1909.07940. arXiv:1909.07940 [cs].

Xilong Wang, Hao Fu, Jindong Wang, and Neil Zhenqiang Gong. STRINGLLM: UNDERSTANDING THE STRING PRO- CESSING CAPABILITY OF LARGE LANGUAGE MODELS. In *International Conference on Learning Representations (ICLR)}*, 2025.

Jason Wei, Xuezhi Wang, Dale Schuurmans, Maarten Bosma, Brian Ichter, Fei Xia, Ed Chi, Quoc Le, and Denny Zhou. Chain-of-thought prompting elicits reasoning in large language models. In *Advances in Neural Information Processing Systems*, volume 35, pp. 24824–24837, 2022. URL https://arxiv.org/abs/2201.11903.

John Whalen, C. R. Gallistel, and Rochel Gelman. Nonverbal counting in humans: The psychophysics of number representation. *Psychological Science*, 10(2):130–137, 1999. doi: 10.1111/1467-9280.00120.

Karen Wynn. Children's understanding of counting. *Cognition*, 36(2):155–193, August 1990. ISSN 00100277. doi: 10.1016/0010-0277(90)90003-3. URL https://linkinghub.elsevier.com/retrieve/pii/0010027790900033.

Nan Xu and Xuezhe Ma. LLM The Genius Paradox: A Linguistic and Math Expert's Struggle with Simple Word-based Counting Problems. In Luis Chiruzzo, Alan Ritter, and Lu Wang (eds.), *Proceedings of the 2025 Conference of the Nations of the Americas Chapter of the Association for Computational Linguistics: Human Language Technologies (Volume 1: Long Papers)*, pp. 3344–3370, Albuquerque, New Mexico, April 2025. Association for Computational Linguistics. ISBN 979-8-89176-189-6. doi: 10.18653/v1/2025.naacl-long.172. URL https://aclanthology.org/2025.naacl-long.172/.

Andy Yang and David Chiang. Counting Like Transformers: Compiling Temporal Counting Logic Into Softmax Transformers. In *Proceedings of the First Conference on Language Modeling (COLM)*, 2024.

Gilad Yehudai, Haim Kaplan, Guy Dar, Royi Rassin, Asma Ghandeharioun, Mor Geva, and Amir Globerson. When can transformers count to n?, 2024. URL https://arxiv.org/abs/2407.15160. arXiv:2407.15160.

Xiang Zhang, Juntai Cao, and Chenyu You. Counting Ability of Large Language Models and Impact of Tokenization, October 2024. URL http://arxiv.org/abs/2410.19730. arXiv:2410.19730 [cs.CL].

Pál Zsámboki, Ádám Fraknói, Máté Gedeon, András Kornai, and Zsolt Zombori. Do Attention Heads Compete or Cooperate during Counting?, February 2025. URL http://arxiv.org/abs/2502.06923. arXiv:2502.06923 [cs.LG].

A Appendix

This section provides details of the prompts we use for operationalizing counting under different complexity settings.

We use the system prompts described in Table 3 to elicit the counting behavior of language models.

Task Type	**User Prompt**
Isolated Repetition Counting	apple apple apple apple apple

Task Type	User Prompt
Non-contextual counting	"You will be given a list of fruits. Your task is to count the TOTAL number of items in the list. Here is the list: <list> monkey bread ackee cupuacu cherry peach noni starfruit baobab fruit Important: You should count the total number of items in the list, NOT the number of distinct/unique items. For example, if the list is <list> apple banana apple orange apple </list> , the total count is 5 (not 3), because "apple" appears 3 times and each occurrence should be counted separately. If the list structure is complex or unclear, use a <scratchpad> to work through your counting process before providing your final answer. Provide your final answer as a single number inside <answer> tags. Your answer should contain only the total count, nothing else.
Contextual occurrence counting	"You will be counting how many times a target word appears in a text blob. Here is the text to analyze: <text> TEXT </text> Here is the target word to count: <target_word> {TARGET_WORD} </target_word> Your task is to count how many times the target word appears in the text as a complete, standalone word. Follow these rules carefully: - Only count exact matches of the target word (the text is already lower-cased, so case doesn't matter) - Only count whole word occurrences - the target word must be bounded by word boundaries (spaces, punctuation, or the start/end of text) - Do NOT count inflections or variations of the word (e.g., if the target is "run", do not count "running", "runs", or "runner") - Do NOT count instances where the target string appears as part of a larger word (e.g., if the target is "cat", do not count "category" or "concatenate") - There may or may not be leading or trailing spaces around occurrences Before providing your final count, use the scratchpad below to work through the text systematically and identify each valid occurrence of the target word. <scratchpad> Identify each occurrence of the target word and verify it is a whole word match. </scratchpad> After working through the text, provide only the numerical count inside <count> tags. Do not include any explanation or additional text. {EXAMPLES} Now, please count the occurrences of the target word in the provided text."

Table 5: **Example user prompts used for eliciting counting behavior in models.**

Counting	System Prompt
Non-contextual (single-entity)	"You are a helpful assistant that can count the number of times a word occurs."
Non-contextual (single-entity with mixed case)	"You are a helpful assistant that counts the total number of items in a list."
Non-contextual (mixed-entity)	"You are a helpful assistant that can count the number of times a word occurs."
Contextual	"You are a helpful assistant that counts the number of times a specific word appears in a given context. Output a number within <answer> and </answer> tags and nothing else."

Table 3: **System prompts used.**

Target Entity Category	List of Target Entities
Special Characters (SPL_CHARS)	*",", ".", ";", "-", "_", "+", "*", "@", "#", "$"*
Numbers (NUMS)	*"0", "1", "2", "4", "5", "6", "8", "9", "14", "17"*
Characters (CHARS)	*"a", "b", "e", "m", "p", "q", "s", "t", "x", "z"*
Concrete Nouns (CONC_NOUNS)	*"door", "banana", "watermelon", "bat", "cat", "ice cream", "chair", "spoon", "knife", "box"*
Abstract Nouns (ABS_NOUNS)	*"idea", "happiness", "success", "fashion", "greed", "love", "life", "strength", "frustration", "beauty"*
Random Strings (RAND_STR)	*"sdfeg", "fgergerjk", "gsggh", "dsdfte", "sfefwefwg", "sdsewew", "travjkx", "uuhwra", "qyerrv", "awsd"*

Table 4: **Target entities for the non-contextual counting task.**

Task Type	Target Counts
Non-contextual (single-entity)	`POWERS OF 2`: [1, 2, 4, 8, 16, 32, 64, 128, 256, 512, 1024, 2048] `MULTIPLES OF 10`: [10, 20, 30, 40, 50, 60, 70, 80, 90] `MULTIPLES OF 50`: [50, 100, 150, 200, 250, 300, 350, 400, 450] `POWERS OF 2 + OFFSET`: [7, 10, 11, 17, 18, 22, 27, 28, 47, 58, 84, 87, 123, 124, 152, 196, 200, 232, 498, 644, 694, 820, 1344, 1503, 1688, 1715, 1768] `MULTIPLES OF 10 + OFFSET`: [6, 8, 13, 19, 21, 23, 26, 27, 32, 37, 38, 42, 46, 47, 53, 56, 58, 59, 67, 68, 73, 79, 81, 83, 86, 87, 89] `MULTIPLES OF 50 + OFFSET`: [44, 51, 62, 81, 116, 117, 126, 137, 163, 195, 197, 222, 231, 235, 269, 285, 293, 294, 332, 351, 356, 376, 389, 417, 428, 438, 443]
Non-contextual (mixed-entity) counting	`UNDER 500`: [328, 58, 13, 380, 141, 126, 115, 72, 378, 53, 347, 457, 280, 45, 303, 217, 17, 16, 48, 112, 120, 259, 309, 14, 288, 102, 367, 333, 360, 215] `OVER 1500`: [1827, 1557, 1512, 1879, 1640, 1625, 1614, 1571, 1877, 1552, 1846, 1956, 1779, 1544, 1802, 1716, 1516, 1515, 1547, 1611, 1619, 1758, 1808, 1513, 1787, 1601, 1866, 1832, 1859, 1714]
Contextual counting	Frequency of the target noun occurring within the Wikipedia article.

Table 6: **Target counts across regimes.**

Category	Entities
Fruits (N=114)	'abiu', 'acai berry', 'ackee', 'apple', 'apricot', 'atemoya', 'avocado', 'bacuri', 'banana', 'baobab fruit', 'black currant', 'black sapote', 'blackberry', 'blood orange', 'blueberry', 'boysenberry', 'breadfruit', 'cactus pear', 'canary melon', 'canistel', 'cantaloupe', 'cape gooseberry', 'carambola', 'casaba melon', 'cherimoya', 'cherry', 'clementine', 'cloudberry', 'coconut', 'cranberry', 'crenshaw melon', 'cupuacu', 'currant', 'date', 'dragon fruit', 'durian', 'elderberry', 'feijoa', 'fig', 'goji berry', 'golden berry', 'gooseberry', 'grape', 'grapefruit', 'ground cherry', 'guava', 'honeydew', 'horned melon', 'huckleberry', 'ice cream bean', 'jabuticaba', 'jackfruit', 'jujube', 'kiwano', 'kiwi', 'kumquat', 'langsat', 'lemon', 'lime', 'lingonberry', 'loganberry', 'longan', 'loquat', 'lychee', 'mamey', 'mandarin', 'mango', 'mangosteen', 'marionberry', 'marula', 'meyer lemon', 'miracle fruit', 'monkey bread', 'mulberry', 'muscadine', 'nance', 'nectarine', 'noni', 'olive', 'orange', 'papaya', 'passion fruit', 'pawpaw', 'peach', 'pear', 'persimmon', 'physalis', 'pineapple', 'plantain', 'plum', 'pomegranate', 'pomelo', 'prickly pear', 'quince', 'rambutan', 'raspberry', 'red currant', 'salak', 'santa claus melon', 'sapodilla', 'soursop', 'starfruit', 'strawberry', 'sugar apple', 'surinam cherry', 'tamarillo', 'tamarind', 'tangerine', 'ugli fruit', 'watermelon', 'white currant', 'white sapote', 'yellow sapote', 'yuzu'

Category	Entities
vegetables ($N = 93$)	'acorn squash', 'anaheim pepper', 'artichoke', 'arugula', 'asparagus', 'banana pepper', 'bean sprouts', 'beet', 'bell pepper', 'bok choy', 'broccoli', 'broccoli rabe', 'broccolini', 'brussels sprouts', 'butterhead lettuce', 'butternut squash', 'button mushroom', 'carrot', 'cassava', 'cauliflower', 'celeriac', 'celery', 'cherry tomato', 'collard greens', 'corn', 'cremini mushroom', 'cucumber', 'daikon radish', 'delicata squash', 'edamame', 'eggplant', 'endive', 'english cucumber', 'fava beans', 'fennel', 'garlic', 'green beans', 'green cabbage', 'green leaf lettuce', 'habanero pepper', 'horseradish', 'iceberg lettuce', 'jalapeno', 'jerusalem artichoke', 'jicama', 'kabocha squash', 'kale', 'kohlrabi', 'leek', 'lima beans', 'mini sweet pepper', 'mustard greens', 'napa cabbage', 'okra', 'oyster mushroom', 'parsnip', 'pattypan squash', 'peas', 'poblano pepper', 'portobello mushroom', 'potato', 'pumpkin', 'radicchio', 'radish', 'red cabbage', 'red leaf lettuce', 'red onion', 'rhubarb', 'roma tomato', 'romaine lettuce', 'rutabaga', 'savoy cabbage', 'scallion', 'serrano pepper', 'shallot', 'shiitake mushroom', 'snow peas', 'spaghetti squash', 'spinach', 'sugar snap peas', 'sweet potato', 'swiss chard', 'tomatillo', 'tomato', 'turnip', 'water chestnuts', 'watercress', 'wax beans', 'white onion', 'yam', 'yellow onion', 'yellow squash', 'zucchini'
animals ($N = 128$)	'alligator', 'alpaca', 'ant', 'armadillo', 'axolotl', 'baboon', 'badger', 'bat', 'bear', 'beaver', 'bee', 'bison', 'boa constrictor', 'butterfly', 'camel', 'cat', 'chameleon', 'cheetah', 'chicken', 'chimpanzee', 'clownfish', 'cobra', 'cougar', 'cow', 'coyote', 'crab', 'crocodile', 'crow', 'deer', 'dog', 'dolphin', 'donkey', 'duck', 'eagle', 'elephant', 'flamingo', 'fox', 'frog', 'gecko', 'giraffe', 'goat', 'goldfish', 'goose', 'gorilla', 'great white shark', 'grizzly bear', 'hamster', 'hawk', 'hedgehog', 'heron', 'hippopotamus', 'horse', 'hummingbird', 'hyena', 'iguana', 'jaguar', 'jellyfish', 'kangaroo', 'koala', 'komodo dragon', 'leopard', 'lion', 'lizard', 'llama', 'lobster', 'lynx', 'manatee', 'meerkat', 'monkey', 'moose', 'mouse', 'octopus', 'orangutan', 'orca', 'ostrich', 'otter', 'owl', 'panda', 'parrot', 'peacock', 'pelican', 'penguin', 'pig', 'pigeon', 'piranha', 'polar bear', 'porcupine', 'python', 'rabbit', 'raccoon', 'rat', 'rattlesnake', 'rhinoceros', 'robin', 'salamander', 'salmon', 'scorpion', 'sea lion', 'sea turtle', 'seagull', 'seahorse', 'seal', 'shark', 'sheep', 'skunk', 'sloth', 'snail', 'sparrow', 'spider', 'squid', 'squirrel', 'starfish', 'stingray', 'swan', 'swordfish', 'tiger', 'toad', 'tortoise', 'toucan', 'tuna', 'turkey', 'vulture', 'walrus', 'weasel', 'whale', 'wolf', 'woodpecker', 'zebra'

Category	Entities
countries ($N = 195$)	'Afghanistan', 'Albania', 'Algeria', 'Andorra', 'Angola', 'Antigua and Barbuda', 'Argentina', 'Armenia', 'Australia', 'Austria', 'Azerbaijan', 'Bahamas', 'Bahrain', 'Bangladesh', 'Barbados', 'Belarus', 'Belgium', 'Belize', 'Benin', 'Bhutan', 'Bolivia', 'Bosnia and Herzegovina', 'Botswana', 'Brazil', 'Brunei', 'Bulgaria', 'Burkina Faso', 'Burundi', 'Cambodia', 'Cameroon', 'Canada', 'Cape Verde', 'Central African Republic', 'Chad', 'Chile', 'China', 'Colombia', 'Comoros', 'Costa Rica', 'Croatia', 'Cuba', 'Cyprus', 'Czech Republic', 'Democratic Republic of the Congo', 'Denmark', 'Djibouti', 'Dominica', 'Dominican Republic', 'Ecuador', 'Egypt', 'El Salvador', 'Equatorial Guinea', 'Eritrea', 'Estonia', 'Eswatini', 'Ethiopia', 'Fiji', 'Finland', 'France', 'Gabon', 'Gambia', 'Georgia', 'Germany', 'Ghana', 'Greece', 'Grenada', 'Guatemala', 'Guinea', 'Guinea-Bissau', 'Guyana', 'Haiti', 'Honduras', 'Hungary', 'Iceland', 'India', 'Indonesia', 'Iran', 'Iraq', 'Ireland', 'Israel', 'Italy', 'Ivory Coast', 'Jamaica', 'Japan', 'Jordan', 'Kazakhstan', 'Kenya', 'Kiribati', 'Kuwait', 'Kyrgyzstan', 'Laos', 'Latvia', 'Lebanon', 'Lesotho', 'Liberia', 'Libya', 'Liechtenstein', 'Lithuania', 'Luxembourg', 'Madagascar', 'Malawi', 'Malaysia', 'Maldives', 'Mali', 'Malta', 'Marshall Islands', 'Mauritania', 'Mauritius', 'Mexico', 'Micronesia', 'Moldova', 'Monaco', 'Mongolia', 'Montenegro', 'Morocco', 'Mozambique', 'Myanmar', 'Namibia', 'Nauru', 'Nepal', 'Netherlands', 'New Zealand', 'Nicaragua', 'Niger', 'Nigeria', 'North Korea', 'North Macedonia', 'Norway', 'Oman', 'Pakistan', 'Palau', 'Palestine', 'Panama', 'Papua New Guinea', 'Paraguay', 'Peru', 'Philippines', 'Poland', 'Portugal', 'Qatar', 'Republic of the Congo', 'Romania', 'Russia', 'Rwanda', 'Saint Kitts and Nevis', 'Saint Lucia', 'Saint Vincent and the Grenadines', 'Samoa', 'San Marino', 'Saudi Arabia', 'Senegal', 'Serbia', 'Seychelles', 'Sierra Leone', 'Singapore', 'Slovakia', 'Slovenia', 'Solomon Islands', 'Somalia', 'South Africa', 'South Korea', 'South Sudan', 'Spain', 'Sri Lanka', 'Sudan', 'Suriname', 'Sweden', 'Switzerland', 'Syria', 'São Tomé and Príncipe', 'Tajikistan', 'Tanzia', 'Thailand', 'Timor-Leste', 'Togo', 'Tonga', 'Trinidad and Tobago', 'Tunisia', 'Turkey', 'Turkmenistan', 'Tuvalu', 'Uganda', 'Ukraine', 'United Arab Emirates', 'United Kingdom', 'United States', 'Uruguay', 'Uzbekistan', 'Vanuatu', 'Vatican City', 'Venezuela', 'Vietnam', 'Yemen', 'Zambia', 'Zimbabwe'

Category	Entities
colors ($N = 233$)	'adobe', 'alice blue', 'amber', 'amethyst', 'antique white', 'apricot', 'aqua', 'aquamarine', 'army green', 'ash', 'ash gray', 'avocado', 'azure', 'baby blue', 'baby pink', 'banana', 'beige', 'bisque', 'black', 'blood red', 'blue', 'blush', 'bone', 'brick red', 'bronze', 'brown', 'bubblegum', 'burgundy', 'burnt orange', 'burnt sienna', 'burnt umber', 'butter', 'canary', 'caramel', 'carmine', 'carolina blue', 'cerulean', 'champagne', 'charcoal', 'chartreuse', 'cherry', 'chestnut', 'chocolate', 'clay', 'cobalt', 'cobalt blue', 'coffee', 'copper', 'coral', 'cornflower blue', 'cornsilk', 'cream', 'cream yellow', 'crimson', 'cyan', 'dark blue', 'dark gray', 'dark green', 'dark orange', 'dark purple', 'dark red', 'dark slate gray', 'dark yellow', 'deep pink', 'deep purple', 'denim', 'dim gray', 'dune', 'dusty rose', 'ebony', 'ecru', 'eggplant', 'electric blue', 'electric yellow', 'emerald', 'espresso', 'fern', 'flamingo', 'flax', 'fluorescent green', 'forest green', 'fuchsia', 'gainsboro', 'ghost white', 'gold', 'golden', 'goldenrod', 'grape', 'grass green', 'gray', 'green', 'grey', 'gunmetal', 'heather', 'honeydew', 'hot pink', 'hunter green', 'ice blue', 'indigo', 'ivory', 'jade', 'jet black', 'kelly green', 'khaki', 'latte', 'lavender', 'lavender blush', 'lemon', 'lemon chiffon', 'light blue', 'light goldenrod', 'light gray', 'light green', 'light orange', 'light pink', 'light purple', 'light red', 'light slate gray', 'light yellow', 'lilac', 'lime', 'lime green', 'linen', 'magenta', 'mahogany', 'marigold', 'maroon', 'mauve', 'medium gray', 'metallic gray', 'midnight blue', 'mint', 'mint green', 'misty rose', 'moccasin', 'mocha', 'moss', 'moss green', 'mulberry', 'mustard', 'navy', 'navy blue', 'neon blue', 'neon green', 'neon orange', 'neon pink', 'neon purple', 'neon red', 'neon yellow', 'nude', 'oatmeal', 'obsidian', 'ocean blue', 'ochre', 'off-white', 'olive', 'olive green', 'onyx', 'orange', 'orchid', 'pale yellow', 'papaya whip', 'parchment', 'pastel blue', 'pastel coral', 'pastel green', 'pastel lavender', 'pastel mint', 'pastel orange', 'pastel peach', 'pastel pink', 'pastel purple', 'pastel yellow', 'peach', 'pearl', 'periwinkle', 'periwinkle purple', 'pine green', 'pink', 'platinum', 'plum', 'powder blue', 'pumpkin', 'purple', 'raspberry', 'raw sienna', 'red', 'rose', 'rose gold', 'rose pink', 'royal blue', 'royal purple', 'ruby', 'rust', 'sage', 'salmon', 'sand', 'sandstone', 'sapphire', 'scarlet', 'seafoam', 'seafoam green', 'seashell', 'sepia', 'sienna', 'silver', 'sky blue', 'slate blue', 'slate gray', 'smoke', 'snow', 'steel blue', 'sunshine', 'tan', 'tangerine', 'taupe', 'tawny', 'teal', 'teal green', 'terracotta', 'thistle', 'tomato', 'turquoise', 'umber', 'vermilion', 'violet', 'walnut', 'wheat', 'white', 'white smoke', 'wine', 'wisteria', 'yellow'

Category	Entities
occupations (197)	'accountant', 'actor', 'actuary', 'advertising manager', 'aerospace engineer', 'agricultural scientist', 'air traffic controller', 'anesthesiologist', 'animator', 'anthropologist', 'archaeologist', 'architect', 'art director', 'astronomer', 'auditor', 'author', 'auto mechanic', 'banker', 'barber', 'bartender', 'biologist', 'biomedical engineer', 'border patrol agent', 'botanist', 'bus driver', 'business analyst', 'carpenter', 'cashier', 'chef', 'chemical engineer', 'chemist', 'chiropractor', 'civil engineer', 'clergy', 'concierge', 'construction manager', 'content creator', 'cook', 'copywriter', 'corporate trainer', 'correctional officer', 'court interpreter', 'court reporter', 'curator', 'customer service representative', 'customs officer', 'cybersecurity analyst', 'data analyst', 'data scientist', 'database administrator', 'delivery driver', 'dental hygienist', 'dentist', 'detective', 'devops engineer', 'dietitian', 'diplomat', 'dock worker', 'economist', 'editor', 'electrical engineer', 'electrician', 'elementary school teacher', 'entrepreneur', 'environmental engineer', 'environmental scientist', 'epidemiologist', 'event planner', 'executive assistant', 'farmer', 'fashion designer', 'film director', 'financial advisor', 'financial analyst', 'firefighter', 'fisherman', 'flight attendant', 'forensic scientist', 'forester', 'funeral director', 'geologist', 'glazier', 'government administrator', 'graphic designer', 'groundskeeper', 'hvac technician', 'hair stylist', 'heavy equipment operator', 'high school teacher', 'horticulturist', 'hotel manager', 'housekeeper', 'human resources manager', 'it support specialist', 'illustrator', 'insurance agent', 'interior designer', 'interpreter', 'ironworker', 'janitor', 'journalist', 'judge', 'landscape architect', 'lawyer', 'legal secretary', 'librarian', 'life coach', 'locksmith', 'logistics coordinator', 'machinist', 'management consultant', 'marine biologist', 'marketing manager', 'mason', 'massage therapist', 'mechanical engineer', 'medical laboratory technician', 'medical transcriptionist', 'meteorologist', 'midwife', 'military officer', 'musician', 'nail technician', 'network engineer', 'neuroscientist', 'news anchor', 'nurse', 'occupational therapist', 'oceanographer', 'operations manager', 'optician', 'optometrist', 'painter', 'paralegal', 'paramedic', 'personal trainer', 'pest control technician', 'pharmacist', 'photographer', 'physical therapist', 'physician', 'physicist', 'pilot', 'pipefitter', 'plumber', 'police officer', 'politician', 'postal worker', 'principal', 'product manager', 'professor', 'project manager', 'psychiatrist', 'psychologist', 'public health officer', 'public relations specialist', 'purchasing agent', 'quality assurance analyst', 'radio host', 'radiologist', 'rancher', 'real estate agent', 'respiratory therapist', 'retail buyer', 'roofer', 'sales representative', 'school counselor', 'security guard', 'ship captain', 'social media manager', 'social worker', 'software engineer', 'soldier', 'special education teacher', 'speech therapist', 'statistician', 'store manager', 'supply chain manager', 'surgeon', 'systems analyst', 'tattoo artist', 'taxi driver', 'teaching assistant', 'technical writer', 'train conductor', 'travel agent', 'truck driver', 'tutor', 'ux designer', 'urban planner', 'veterinarian', 'video game designer', 'waiter', 'warehouse worker', 'web developer', 'welder', 'zoologist'

Table 7: **Entities grouped by semantic categories for non-contextual counting.**

B APPENDIX

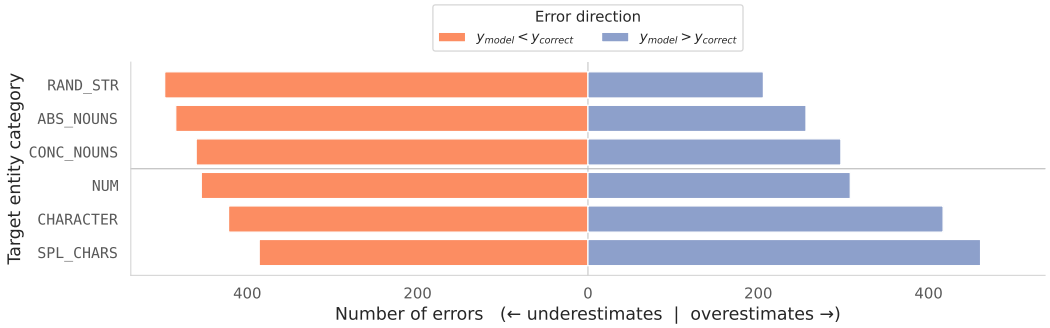


Figure 10: Counting Error Directions in GPT 5.2

C APPENDIX

We ablate the context within which the target entity is embedded (by cipher-mapping all English alphabets that do not match the target entity) and compare its effects to target ablation (by cipher-mapping the target is isolation). These two settings are designed to capture how robustly models perform string matching and counting tasks under minimal conceptual overlap between the target and context.

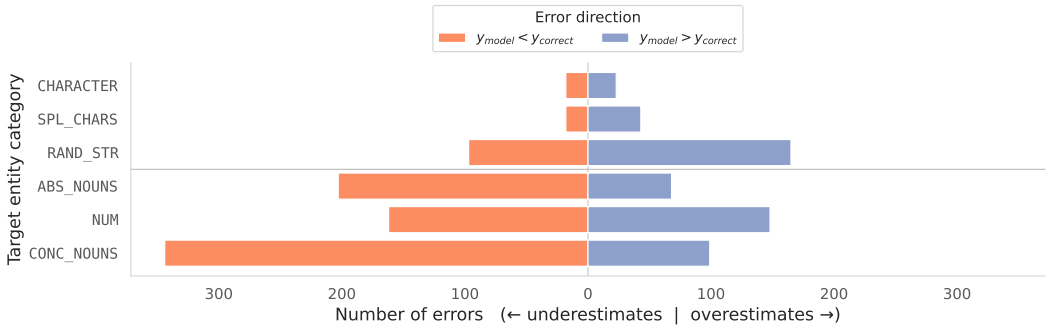


Figure 11: Counting Error Directions in Gemini 2.5 Pro

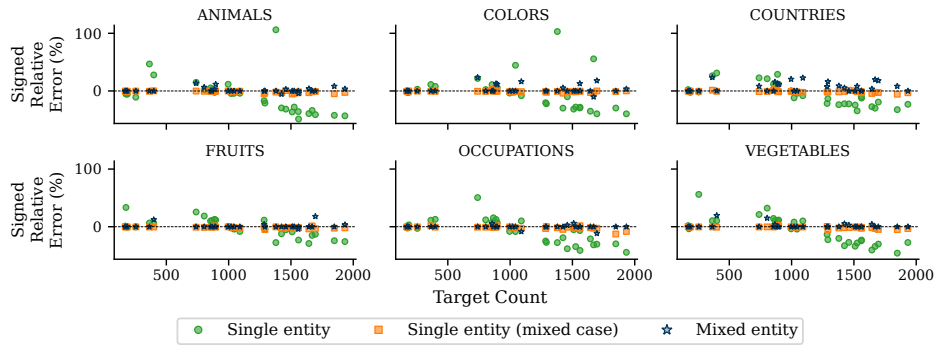


Figure 12: **Non-contextual counting on single- vs mixed-entity lists (Gemini 2.5 Pro).** When presented with lists containing the same number of elements, Gemini 2.5 Pro produces almost no errors on lists sampled from a broader set of entities within the semantic category but is prone to more errors when the list is composed of a single unique element, especially at higher counts.

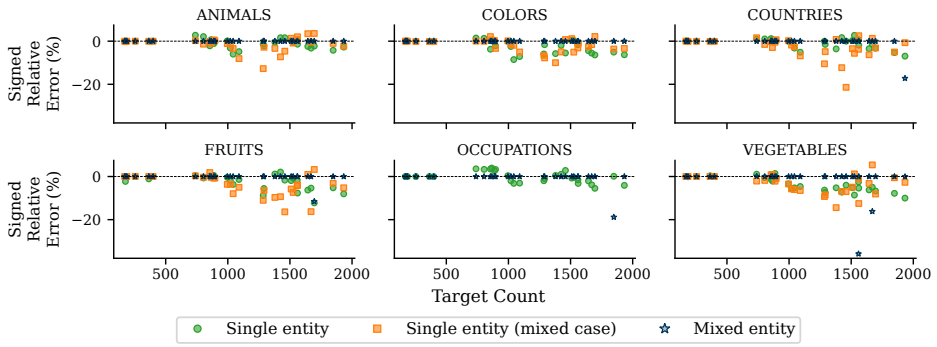


Figure 13: **List diversity effects (Claude Opus 4.5).** When presented with lists containing the same number of elements, Claude Opus 4.5 produces almost no errors on lists sampled from a broader set of entities within the semantic category but is prone to more errors when the list is composed of a single unique element, especially at higher counts. Both settings use medium reasoning effort.

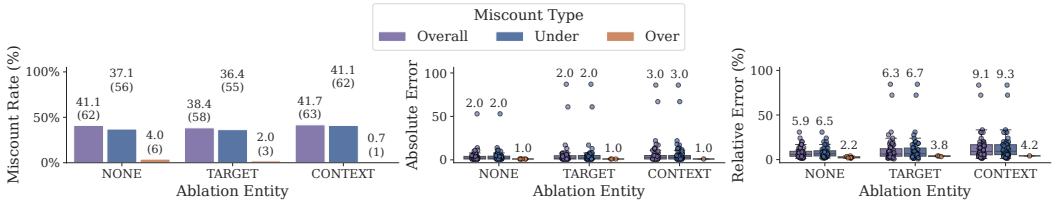
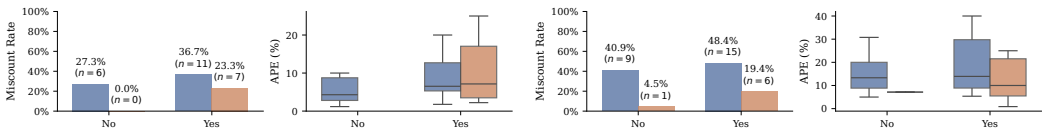


Figure 14: **Target versus context ablation (Gemini 2.5 Pro).** Context and target ablations produce relatively minor differences in the miscount rates, error magnitudes and relative errors compared to no ablation.



(a) Gemini 2.5 Pro

(b) GPT 5.2

Figure 15: **Miscounting in the presence of inflectional forms.**

Statistic	Target Ablation	Context Ablation
N	151	151
Miscount frequency		
# Miscounts	$62 \rightarrow 58$	$62 \rightarrow 63$
Miscount rate	$0.411 \rightarrow 0.384$	$0.411 \rightarrow 0.417$
Rate difference (Δ)	-2.6%	0.7%
Worsened pairs	15	13
Improved pairs	19	12
Both miscount	43	50
Both correct	74	76
Discordant odds ratio	0.795	1.080
McNemar exact p	0.608	1.000
McNemar p_{Holm}	1.000	1.000
Error composition		
# Correct	$89 \rightarrow 93$	$89 \rightarrow 88$
# Undercounts	$56 \rightarrow 55$	$56 \rightarrow 62$
# Overcounts	$6 \rightarrow 3$	$6 \rightarrow 1$
Correct $\rightarrow$ Under	13	13
Correct $\rightarrow$ Over	2	0
Under $\rightarrow$ Correct	16	8
Over $\rightarrow$ Correct	3	4
Under $\rightarrow$ Over	0	0
Over $\rightarrow$ Under	2	1
Stuart–Maxwell χ^2	1.399	6.147
Degrees of freedom	2	2
Stuart–Maxwell p	0.497	0.046
Stuart–Maxwell p_{Holm}	0.497	0.093

Table 8: **Paired comparison of counting outcomes (Gemini 2.5 Pro).** The two panels are nested resolutions of the same paired transition table and each carries its own test. Miscount frequency collapses undercounts and overcounts into a single miscount outcome and is tested with the exact McNemar test; error composition retains all three outcomes and is tested for marginal homogeneity with Stuart–Maxwell. The Holm correction is applied within each test across the two ablations.

Neither target nor context ablation induces statistically significant changes in Gemini 2.5 Pro's miscount frequencies or error magnitudes. Target ablation produces negligible changes across all measures. Context ablation leaves overall accuracy unchanged but shows a nonsignificant tendency to redistribute errors from overcounting toward undercounting. Although aggregate relative error decreases slightly under context ablation, error magnitude does not improve among shared failures, suggesting that the aggregate change is driven primarily by transitions in which examples fail rather than by uniformly more accurate estimates.

For GPT-5.2, target ablation shows a consistent but nonsignificant improvement pattern: it reduces miscount frequency descriptively and lowers both absolute and relative error among all examples and shared failures. Context ablation leaves miscount frequency unchanged but shifts errors from overcounting toward undercounting and descriptively increases error severity among shared failures. None of these effects survives statistical testing or multiple-comparison correction.

We find that cipher-ablating the target can lead to a drop in miscount rates (58.5% to 50.7% of passages for GPT 5.2), whereas, cipher-ablating the context leads to an increase in error rates (58.5% to 62.0% of passages for GPT 5.2). However, neither of them were statistically significant under paired McNemar tests ($p_{\text{Holm}} = 0.198$ and 0.424 for GPT 5.2). Likewise, no statistically significant redistribution of miscounting errors (over/under/correct) was observed under either ablations for GPT 5.2 (Stuart–Maxwell $\chi^2(2) = 3.72$, $p_{\text{Holm}} = .311$; $\chi^2(2) = 1.54$, $p_{\text{Holm}} = .462$). However, under target ablation, the error magnitude of the generated counts increased from 6.4 occurrences

	Target Ablation		Context Ablation	
Statistic	Absolute	Relative	Absolute	Relative
N	151	151	151	151
Mean	1.570 → 2.172	3.102 → 4.169	1.570 → 2.768	3.102 → 5.436
Median	0.000 → 0.000	0.000 → 0.000	0.000 → 0.000	0.000 → 0.000
Mean Δ	0.603	1.067	1.199	2.334
Median Δ	0.000	0.000	0.000	0.000
95% CI	[-0.522, 1.728]	[-0.498, 2.631]	[0.076, 2.321]	[0.734, 3.933]
Paired t	1.058	1.347	2.111	2.882
Paired t p	0.292	0.180	0.036	0.005
Paired t p_{Holm}	0.360	0.360	0.109	0.018
Cohen's d_z	0.086	0.110	0.172	0.235
Wilcoxon W	976.000	929.500	469.500	398.500
Wilcoxon p	0.997	0.742	0.004	0.001
Wilcoxon p_{Holm}	1.000	1.000	0.012	0.003
Rank-biserial correlation	0.001	0.048	0.432	0.518

Table 9: **Overall paired error comparison across all valid examples (Gemini 2.5 Pro).**

	Target Ablation		Context Ablation	
Statistic	Absolute	Relative	Absolute	Relative
N	43	43	50	50
Mean	4.744 → 6.628	8.365 → 10.555	4.380 → 7.620	8.049 → 12.365
Median	3.000 → 2.000	7.317 → 6.000	2.000 → 3.000	6.523 → 9.262
Mean Δ	1.884	2.189	3.240	4.316
Median Δ	0.000	0.000	0.000	0.000
95% CI	[-2.031, 5.799]	[-1.749, 6.128]	[-0.083, 6.563]	[0.972, 7.661]
Paired t	0.971	1.122	1.960	2.593
Paired t p	0.337	0.268	0.056	0.012
Paired t p_{Holm}	0.537	0.537	0.167	0.050
Cohen's d_z	0.148	0.171	0.277	0.367
Wilcoxon W	199.500	176.000	113.500	84.000
Wilcoxon p	0.936	0.539	0.005	0.001
Wilcoxon p_{Holm}	1.000	1.000	0.014	0.003
Rank-biserial correlation	0.017	0.133	0.570	0.682

Table 10: **Paired error-severity comparison on shared failures (Gemini 2.5 Pro).**

on average to 4.1 occurrences after ablation ($p_{\text{Holm}} = .007$). Context ablation, on the other hand, produced no such statistically significant shrinkage ($6.4 \rightarrow 5.4$, $p_{\text{Holm}} = .263$).

Statistic	Target Ablation	Context Ablation
N	156	156
Miscount frequency		
# Miscounts	91 → 77	91 → 94
Miscount rate	0.583 → 0.494	0.583 → 0.603
Rate difference (Δ)	-9.0%	1.9%
Worsened pairs	13	15
Improved pairs	27	12
Both miscount	64	79
Both correct	52	50
Discordant odds ratio	0.491	1.240
McNemar exact p	0.038	0.701
McNemar p_{Holm}	0.077	0.701
Error composition		
# Correct	65 → 79	65 → 62
# Undercounts	87 → 73	87 → 87
# Overcounts	4 → 4	4 → 7
Correct → Under	10	15
Correct → Over	3	0
Under → Correct	24	11
Over → Correct	3	1
Under → Over	1	5
Over → Under	1	1
Stuart–Maxwell χ^2	5.521	1.532
Degrees of freedom	2	2
Stuart–Maxwell p	0.063	0.465
Stuart–Maxwell p_{Holm}	0.127	0.465

Table 11: **Paired comparison of counting outcomes (GPT 5.2).** The two panels are nested resolutions of the same paired transition table and each carries its own test. Miscount frequency collapses undercounts and overcounts into a single miscount outcome and is tested with the exact McNemar test; error composition retains all three outcomes and is tested for marginal homogeneity with Stuart–Maxwell. The Holm correction is applied within each test across the two ablations.

	Target Ablation		Context Ablation	
Statistic	Absolute	Relative	Absolute	Relative
N	156	156	156	156
Mean	6.064 → 3.814	10.000 → 6.479	6.064 → 5.096	10.000 → 8.806
Median	1.000 → 0.000	5.534 → 0.000	1.000 → 1.000	5.534 → 5.556
Mean Δ	-2.250	-3.522	-0.968	-1.195
Median Δ	0.000	0.000	0.000	0.000
95% CI	[-3.556, -0.944]	[-5.495, -1.548]	[-2.186, 0.250]	[-3.525, 1.136]
Paired t	-3.402	-3.524	-1.570	-1.013
Paired t p	0.001	0.001	0.118	0.313
Paired t p_{Holm}	0.003	0.002	0.237	0.313
Cohen's d_z	-0.272	-0.282	-0.126	-0.081
Wilcoxon W	1327.500	1471.000	1925.500	1960.500
Wilcoxon p	0.000	0.001	0.186	0.236
Wilcoxon p_{Holm}	0.001	0.003	0.371	0.371
Rank-biserial correlation	-0.441	-0.381	-0.155	-0.140

Table 12: **Overall paired error comparison across all valid examples (GPT 5.2).**

	Target Ablation		Context Ablation	
Statistic	Absolute	Relative	Absolute	Relative
N	64	64	79	79
Mean	13.922 → 8.969	18.578 → 12.803	11.734 → 9.582	17.981 → 13.698
Median	7.000 → 5.000	11.652 → 11.270	5.000 → 3.000	11.765 → 11.111
Mean Δ	-4.953	-5.775	-2.152	-4.283
Median Δ	-2.000	-4.455	-1.000	-0.820
95% CI	[-8.035, -1.872]	[-9.152, -2.398]	[-4.508, 0.204]	[-7.627, -0.938]
Paired t	-3.212	-3.417	-1.818	-2.549
Paired t p	0.002	0.001	0.073	0.013
Paired t p_{Holm}	0.006	0.004	0.073	0.025
Cohen's d_z	-0.402	-0.427	-0.205	-0.287
Wilcoxon W	425.500	446.000	894.000	825.000
Wilcoxon p	0.001	0.003	0.087	0.033
Wilcoxon p_{Holm}	0.006	0.008	0.087	0.067
Rank-biserial correlation	-0.485	-0.460	-0.238	-0.297

Table 13: **Paired error-severity comparison on shared failures (GPT 5.2).**

	Masked Context Ablation	
No Ablation	Correct	Incorrect
Correct	46	13
Incorrect	25	60
Incorrect total	85 (No Ablation)	73 (Masked Context Ablation)
Exact McNemar test		$p = 0.073$

Note. The analysis includes 144 complete pairs. The exact McNemar test uses only discordant pairs and is two-sided.

Table 14: **Context Masking Effects: Correct/Incorrect transitions (GPT 5.2).**

	Masked Context Ablation		
No Ablation	Under	Correct	Over
Under	45	22	14
Correct	10	46	3
Over	1	3	0
Stuart–Maxwell test	$\chi^2(2) = 14.85, p =< 0.001$		

Note. There were 14 under-to-over and 1 over-to-under direction transitions. The exact test comparing direction reversals gave $p =< 0.001$.

Table 15: **Paired treatment analysis: Miscount direction changes (GPT-5.2).**

Measure	No Ablation	Masked Context Ablation	Paired change
Mean absolute value	10.31	8.82	-1.49
Median absolute value	5.53	1.26	0.00
Interquartile range	[0.00, 13.74]	[0.00, 10.13]	—
Mean-change 95% CI	—	—	[-4.14, 1.26]
Increased/decreased/unchanged		30 / 59 / 55	
Wilcoxon signed-rank test		$W = 1447.50$, $p = 0.023$	
Paired t-test		$t(143) = -1.07$, $p = 0.288$	

Note. Paired change is $|\text{Masked Context Ablation}| - |\text{No Ablation}|$; positive values indicate increase in magnitude. The Wilcoxon and paired t-tests are two-sided.

Table 16: **Paired treatment analysis: Unsigned relative errors (GPT 5.2).**

Measure	No Ablation	Masked Context Ablation	Paired change
Mean absolute value	6.69	7.96	1.27
Median absolute value	1.00	1.00	0.00
Interquartile range	[0.00, 6.00]	[0.00, 3.00]	—
Mean-change 95% CI	—	—	[-0.99, 4.19]
Increased/decreased/unchanged		30 / 59 / 55	
Wilcoxon signed-rank test		$W = 1404.50$, $p = 0.014$	
Paired t-test		$t(143) = 0.96$, $p = 0.339$	

Note. Paired change is $|\text{Masked Context Ablation}| - |\text{No Ablation}|$; positive values indicate increase in magnitude. The Wilcoxon and paired t-tests are two-sided.

Table 17: **Paired treatment analysis: Unsigned error magnitudes (GPT 5.2).**

Degradation	N	Miscount rate	Under	Over	Median APE	95% CI
0%	158	58.9%	56.3%	2.5%	5.56	[1.85, 8.82]
10%	158	69.0%	66.5%	2.5%	11.25	[7.77, 14.81]
25%	158	65.8%	65.8%	0.0%	10.00	[6.45, 14.07]
50%	157	71.3%	69.4%	1.9%	13.33	[8.70, 16.67]
75%	158	67.1%	64.6%	2.5%	10.00	[7.69, 13.33]
90%	158	65.2%	60.8%	4.4%	7.14	[5.56, 11.44]
100%	156	60.3%	55.8%	4.5%	5.56	[2.60, 7.42]

Note. Under and Over are the shares of all instances miscounted in each direction, so they sum to the miscount rate. APE is the absolute percentage error over all instances at that degradation rate, including exact counts. The interval is a percentile bootstrap 95% confidence interval for the median.

Table 18: **Context degradation: counting performance by rate (GPT 5.2).**

Outcome	Test	N	Statistic	df	p	Effect size
Miscount	Cochran's Q	156	20.84	6	0.002	—
APE	Friedman	156	78.79	6	< 0.001	0.084

Note. Both tests are restricted to instances observed at every degradation rate. The effect size for the Friedman test is Kendall's W; no effect size is reported for Cochran's Q.

Table 19: **Context degradation: omnibus paired tests (GPT 5.2).**

	Miscount				APE		
Degradation	Worsened	Improved	p	p_{Holm}	Median Δ	p	p_{Holm}
10%	21	5	0.002	0.012	0.00	< 0.001	< 0.001
25%	21	10	0.071	0.213	0.00	< 0.001	< 0.001
50%	27	7	< 0.001	0.005	0.59	< 0.001	< 0.001
75%	26	13	0.053	0.213	0.00	< 0.001	< 0.001
90%	21	11	0.110	0.220	0.00	0.003	0.006
100%	15	12	0.701	0.701	0.00	0.236	0.236

Note. Each rate is compared against the 0% baseline. Worsened and Improved count instances that changed from correct to miscounted and the reverse; their p-values come from two-sided exact McNemar tests. Median Δ is the median within-instance change in absolute percentage error, tested with a two-sided Wilcoxon signed-rank test. Holm correction is applied across the comparisons within each outcome. All tests use the instances observed at every rate.

Table 20: **Context degradation: comparisons against the undegraded baseline (GPT 5.2).**