Masked Semantic Priming in GPT-4o Mini: Investigating LLMs Through a Cognitive Psychology Lens

Filippo Colombi
MSc student at CiMEC
University of Trento

April 22, 2025

Contents

1 Introduction

Framework Semantic priming, a fundamental phenomenon in psycholinguistics and cognitive neuroscience, provides critical insights into how the human brain organizes and retrieves semantic knowledge. It refers to the facilitation of a target word's recognition or processing when it is preceded by a semantically related prime. This effect was first empirically demonstrated by Meyer and Schvaneveldt in 1971 [5] using the lexical decision task where participants identified words more quickly when followed related primes (e.g. bread-butter) compared to unrelated pairs (e.g. guitar-butter). This finding suggested that related concepts in the mental lexicon are interconnected, enabling more efficient retrieval. Building on this, Collins and Loftus [1] proposed the spreading activation model of semantic memory in 1975. According to this model, the mental lexicon is structured as a network of interconnected nodes representing concepts. When a prime word is processed, activation spreads to related nodes, reducing the activation threshold required to recognize semantically connected targets. This framework accounts for the graded nature of semantic priming, where more closely related concepts exhibit stronger priming effects. Furthermore, Neely [6] differentiated between automatic and controlled semantic priming processes in 1977. Automatic priming occurs rapidly and unconsciously at short stimulus onset asynchronies (SOAs), reflecting the passive spread of activation within the semantic network. In contrast, controlled priming involves conscious, strategic processes that emerge at longer SOAs, where participants anticipate certain responses based on contextual cues. The neural correlate of semantic priming was clarified by Kutas and Hillyard in 1980 [4] with the discovery of the N400 event-related potential (ERP) component. It is a negative deflection of the brain electrical activity that peaks approximately 400 ms after the presentation of a semantically incongruent stimulus. In their study, unexpected sentence endings elicited larger N400 responses compared to congruent completions, providing neurophysiological evidence that semantic priming modulates brain activity during language comprehension.

Motivations This foundational framework informs the present study, which investigates whether similar semantic priming effects manifest in large language models (LLMs) like GPT-4o. By comparing the probabilistic output of the model in related and unrelated prime-target conditions, this research explores whether LLMs exhibit cognitive-like patterns of semantic association, bridging computational modeling with traditional psycholinguistic paradigms. The motivation behind this study stems from a broader interest in cognitive modeling using AI. These systems offer a convenient starting point for modeling and exploring human language processing due to their architecture and training on vast amounts of linguistic data. A critical question is whether the behaviors they exhibit are unique to their training processes or if they mirror transferable cognitive mechanisms inherent to human language processing. Understanding this could contribute to the debate of whether LLMs merely reflect statistical learning or if they approximate the cognitive structure that govern human semantic memory. Neural networks like GPT are trained on massive datasets, capturing statistical regularities, co-occurrence patterns and semantic relationships present in human language. While these models are not biological in nature, the structured statistical patterns they learn often mimic human-like associations. This raises intriguing questions: do these models, through exposure to language data, develop semantic networks akin to those observed in the human brain? And if so, can they serve as valid proxies for studying cognitive processes like semantic priming? Beyond theoretical interests, there are significant practical applications to this line of inquiry. These systems could be employed to predict and model human behavior in various linguistic tasks, providing a new tool for psycholinguistic research. Moreover, understanding how closely they align with human cognitive processes could inform the refinement of AI architectures, enabling the development of models that better capture human-like semantic organization. GPT-4o is a state-of-the-art (SOTA) model in numerous linguistic domains, including natural language understanding, text generation, translation and dialogue systems. Its ability to produce highly coherent, human-like linguistic artifacts makes it an ideal candidate for investigating semantic priming effects. Beyond the mere scarciity of experiments on priming, there remains a broader and more fundamental question: To what extent do LLMs, particularly closed-source models, exhibit semantic processing mechanisms that align with human psycholinguistic assessments? While extensive research has been conducted on model performance and generative capabilities, little is known about whether their response to such assessments parallel those reported in human. This is particularly relevant given GPT-4o's autoregressive nature, where each word is predicted based on the preceding context. This mechanism inherently mirrors aspects of the human predictive processing in language comprehension, making it a suitable ground for examining whether priming emerges from the model's output.

Research Question and Hypotheses The present work proposes to investigate whether LLMs, such as GPT-4o[1], exhibit semantic priming effects similar to those observed in human cognition, exploring if semantic associations emerging from their probabilistic outputs reflect transferable cognitive mechanisms. This research is situated within a growing field that compares AI to human cognition, exploring parallels and divergences. The aim is to assess whether the model not only reflects simple statistical learning but also develops semantic structures resembling human semantic networks. In other words, the goal is to determine whether the autoregressive behavior of the model generates priming effects comparable to those observed in traditional psychological paradigms. Therefore, the research question I propose is the following: **Does GPT-4o mini model exhibit a significant difference in the probability values of target words when presented in related priming conditions compared to unrelated conditions?**

Expected Outcomes It is hypothesized that targets will exhibit higher probabilities values in the related condition compared to those presented in unrelated conditions. This structure allows for the investigation of whether the emergent cognitive traits of LLMs can be considered analogous to the dynamics of human semantic memory and whether traditional psycholinguistic paradigms can be employed to evaluate the validity of these models as devices for cognitive research.

[1]The experiment was run with GPT-4o mini. However, I will often refer to it as GPT-4o or GPT throughout the text. This is just to make reading as smooth as possible.

2 Methodology

In autoregressive systems as GPT-4o, text generation is fundamentally modeled as a conditional probability problem. The model predicts the next word in a sequence based on the preceding context, represented mathematically as

$$P(w_t \mid w_1, w_2, ..., w_{t-1}) \tag{1}$$

where $P(w_t)$ is the probability of generating a word given the previous ones. This probabilistic framework underpins how the model processes language and generates outputs, making it a suitable foundation for investigating semantic priming effects. In the context of this experiment, the target word is presented after a prime that is either semantically related or unrelated. To assess whether GPT-4o exhibits priming effects, the following contrast was applied:

$$P(target \mid related\_context) \text{ vs. } P(target \mid unrelated\_context)$$

If semantic priming is present, the model should assign a higher probability to the target word in the *related* condition, reflecting an internal representation of semantic association similar to those of humans. GPT models output not only the predicted tokens but also the log-probabilities (logprobs) associated with each token:

$$logprob(w_t) = log[P(w_t \mid w_1, w_2, ..., w_{t-1})] \tag{2}$$

A logprob closer to 0 indicates a higher predicted probability, while more negative values indicate lower confidence in the prediction. In this experiment, I use logprobs to quantify the model's confidence in predicting the target word. Thus, semantic priming is operationalized as:

$$logprob_{related}(target) > logprob_{unrelated}(target) \tag{3}$$

2.1 The Experiment

In this experiment, GPT-4o mini was presented with prime-target pairs, where the prime word was either semantically related or unrelated to the masked target word embedded within a sentence. For each trial, the model received a prompt consisting of the prime followed by a sentence with the target word omitted, and was instructed to generate a single word to fill the blank.

Stimuli Presentation The stimuli were presented to GPT through 500 structured API calls designed to simulate an experimental paradigm of cognitive psychology. Each stimulus consisted of a prime word (semantically related or unrelated to the target) and a sentence containing a masked target word. The API was configured to prompt the model with both the prime and the incomplete sentence as input text: [Prime Word]. [Sentence with the target masked as "..."]

The prompt used during experimentation

```
(model=gpt-4o-mini,
messages=[
    {"role": "system", "content": "You do text-completion.
    I will provide you a sentence with a blank '...',
    your task is to return a single word to complete it."},
    {"role": "user", "content": input_text}
],
temperature=0
)
```

For example, in a related condition, the prime *"below"* may precede the sentence *"The Ferrari finished six places ... the Mercedes"*, where the target is *"above"*. In the unrelated condition, the same sentence would be preceded by an unrelated prime such as *"postage"*. This structure allowed for direct comparison of the model's predictions across priming conditions. To ensure controlled responses, the model was provided with a system instruction to return a single-word completion for the masked portion of the sentence. The temperature was set to zero to minimize randomness and enforce deterministic outputs, and finally logprobs were requested for the predicted token, together with the top 15 alternatives.

Retrieval of Log-Probabilities Logprobs provide an exhaustive measure of the model's confidence in predicting a given token because they reflect the probability distribution over multiple possible continuations, rather than just the most likely one. They allow for a nuanced comparison of how strongly the model favors certain predictions, making them particularly useful for assessing semantic priming effects. However, retrieving logprobs for the intended target posed a computational challenge due to the tokenization structure of GPT outputs, requiring a sophisticated reconstruction algorithm. When GPT generates a response, it predicts the single most likely token (i.e., the actual completion), but it can also return logprob values for multiple alternative predictions—if explicitly requested in the API call. These values are stored in a structure that contains the predicted token along with a ranked set of alternatives, each associated with its probability. An additional complication arose because GPT often predict subword units, meaning that a target word might be split into multiple tokens[2]. Such level of complexity necessitated a reconstruction system capable of piecing together each "brick" to retrieve the log-probability of the intended word. The retrieval system operated by matching the original target word against the set of alternative completions of the model. If the target appeared in its entirety among the predictions, its associated logprob was directly extracted. Conversely, when the model provided sub-word tokens, a beam search strategy was employed to reconstruct the word step-by-step. At each stage, candidate sequences were expanded by adding predicted tokens, ensuring that only those maintaining a valid morphological match with the target were retained. Once a valid reconstruction was found, the sum of the probabilities of constituent tokens was

[2]All GPT models leverage a Byte Pair Encoding (BPE) tokenizer, which allows for flexible and semantically complete processing of linguistic data

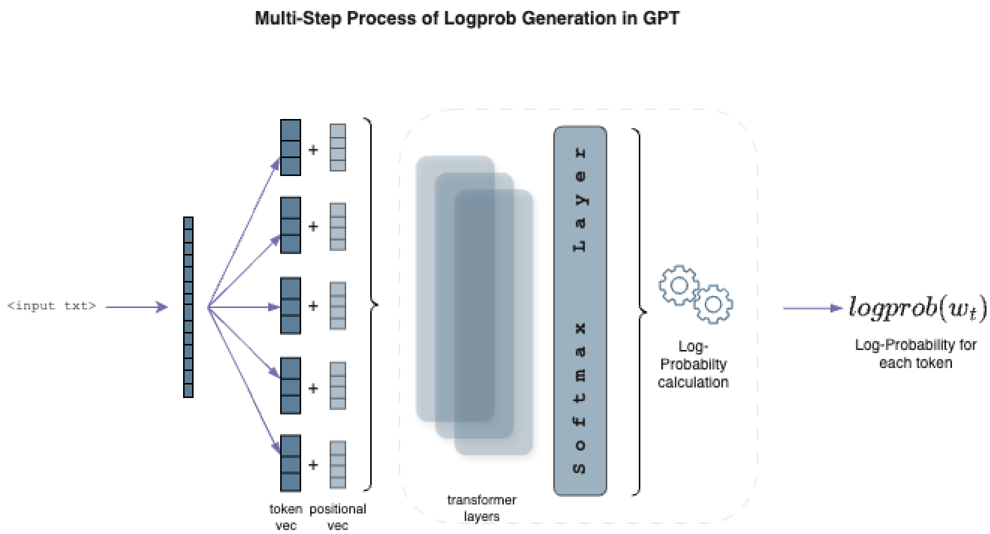

Figure 1: High-level schematic of GPT's multi-step token-prediction pipeline, where token and positional embeddings flow through the transformer architecture. In the final stage, the model's output logits are normalized through a softmax to produce probabilities over the entire vocabulary, which are then transformed into log-probabilities for each potential next token

computed, and the least negative candidate (i.e., the most probable one) was selected as the best match. Where no reconstruction matched the original target, no logprob was assigned (NaN), leaving its interpretation for later stages of analysis. .

Data Construction The stimuli set was built following previous research[3] and was designed to ensure that semantic associations were robustly controlled. A total of 250 triplets (*target, related prime, unrelated prime*) were selected from the semantic priming project, a widely used database containing highly validated prime-target association from human behavioral studies. The rationale behind using SPP was its empirical grounding—these prime-target have been extensively tested in psycholinguistic experiments, making them an ideal starting point for evaluating whether LLMs, like GPT, exhibit cognitive processes akin to those observed in human behavioral tests. Given that GPT is trained on massive linguistic corpora, it has probably internalized complex semantic structures, making it a suitable model for priming-based investigations.

To construct the experimental dataset, the following procedure was applied:

1. Selection of prime-tareget pairs:
   - A randomly chosen prime-target pair was selected from SPP in the related condition.
   - The corresponding prime-target pair was selected to contrast with the related condition.
   - Only first-associate (most common) target were considered, ensuring strong semantic links for the related condition.
2. Pairing process:

- Each related and unrelated prime was paired with the same target word, creating a contrastive pair.

3. Contextual sentence construction:
   - A sentence was invented to serve as a contextual frame for the target word.
   - The target word was removed from the sentence and replaced with a placeholder ("...") creating a fill-in-the-blank format for the model.

4. Tabular data representation:
   - The entire dataset was stored in a structured tabular format, with each stimulus set organized as follows

ID	**Type**	**Word (Prime)**	**Target**	**Sentence**
001	Related	below	above	"The Ferrari finished six places ... the Mercedes"
002	Unrelated	postage	above	"The Ferrari finished six places ... the Mercedes."

Table 1: Example of Prime-Target Stimuli: Each two consecutive rows represent a contrastive pairs. The model was prompted with one row at time, following the logic described in the *Stimuli Presentation* paragraph

2.2 Statistical Testing

To determine whether GPT-4o exhibits semantic priming effects, a statistical approach was designed to compare the log-probabilities of target words across related vs. unrelated priming conditions. Since logprobs are continuous numerical values, they provide a measure of the model's confidence in predicting a given word, making them suitable for inferential statistical analysis. The key objective of this analysis was to assess whether logprobs were significantly higher (closer to 0) in the related condition compared to the unrelated condition, mirroring the facilitatory mechanism observed in human priming studies. Given the paired nature of the data—where each target word appears in both conditions with the same sentence context—the statistical analysis was designed to compare logprobs at the within-item level. Statistical tests often require that data distribution meets certain assumptions. Specifically, normality was a key consideration: if the distribution of logprobs followed a normal pattern, a paired t-test would be appropriate; if not, a Wilcoxon signed-rank test, a popular non-parametric alternative, would be used instead. Following this strategy, an initial assessment of normality was planned, ensuring that the choice of statistical test was applied ad-hoc, rather than arbitrary. This decision was crucial because logprobs are inherently skewed measures, often concentrated around certain thresholds, and the dataset was expected to contain NaN values where the model failed to predict (or the retrieval algorithm failed to recompose) the target word. To maintain statistical rigor, missing values would be handled through imputation, but this step also had the potential to affect normality, requiring a flexible approach.

Multiple Imputation Approach The first strategy involved multiple imputation, a statistical technique that estimates missing logprobs based on the distribution of observed data. Imputation is considered a reasonable approach to retain a larger dataset while minimizing bias. Here, an assumption of near-random data missingness had been adopted, although similar hypotheses are often difficult to verify.

Complete-Case Analysis Precisely because it is difficult to determine with certainty whether the data is missing for largely random reasons, it is also useful to perform the test on the dataset without imputation. Therefore, the second approach involved analyzing the subset of the results where log-probabilities for each condition were reconstructed.

Both approaches were then tested following the aforementioned statistical decision tree: if normality was preserved, a paired t-test would be applied; if not, the Wilcoxon signed-rank would be used instead.

3 Results

The aim of this results section is to determine whether GPT-4o mini exhibits semantic priming effects, measured as differences in log-probabilities of target words in related vs. unrelated priminig conditions. Given the presencce of missing—cases where the experiment failed to generate the expected target word—two complementary analytical approaches were adopted. Summarizing from the previous section: (a) Mulitple Imputation, which estimates missing values to maintain the statistical power, and (b) Complete-Case Analysis, which restricts the dataset to instances where logprobs were successfully retrieved in both conditions, ensuring pairwise comparisons.

Multiple Imputation Results Before conducting hypothesis testing, missing values in logprobs were addressed using multiple imputation (MI). Out of 500 total observations, 201 (40%) were missing, requiring imputation to allow for a complete dataset. Five imputed datasets were generated using a multivariate imputer that estimates each value from all the others. Pooled estimates were finally derived. To asses how imputation affected the distribution of logprobs, summary statistics were calculated before and after imputation. The only relevant variation is over standard deviation (std). To determine whether a parametric test or a non-parametric alternative was appropriate, normality of the imputed logprobs was assessed using the Shapiro-Wilk test. This evidenced a significant departure from normality ($W = 0.891$, $p = 0.00024$) indicating that a non-parametric test was required for hypothesis testing. A Wilcoxon signed-rank test showed that there is no strong evidenced that GPT-4o mini assigned significantly higher logprobs to targets in the related condition vs the unrelated condition ($T = 441.0$, $p = 0.088$). This contrasts with expectations, as human studies typically show a clear priming effect in reaction times and lexical decision tasks.

Complete-Case Results The Complete-Case analysis was conducted using only full retrieved prime-target pairs, ensuring that all statistical comparisons were based on directly observed data. Out of 500 total trials, 298 logprob values were successfully retrieved, but only 127 contrastive pairs could be reconstructed for direct comparison. This represents a substantial reduction in sample size, which affects statistical power but ensures that no assumptions were made about missing values. Congruently to what was done with imputed data, a normality assessment was conducted to confirm a strong deviation from normality ($W = 0.789$, $p = 3.17 \times 10^{-12}$). Since normality assumption was violated, a Wilcoxon signed-rank test was conducted to compare the survived logprobs. Unlike multiple imputation, the complete-case yielded a significant result ($T = 1793.0$, $p = 4.64 \times 10^{-8}$). This provides evidence that GPT-4o mini exhibits a semantic priming effect, with significantly higher log-probabilities for target words in related conditions than in unrelated conditions.

4 Discussion

The findings of this study offer an interesting perspective on the challenges of using LLMs in cognitive modeling. While complete-case analysis detected a significant priming effect, the multiple imputation approach did not, raising important methodological and conceptual inquiries. The discussion is divided into two sections: (a) methodological considerations, focusing on missing data challenges, tokenization artifacts, statistical sensitivity, and potential imputation biases that may have influenced the results and (b) conceptual implications, addressing whether LLMs exhibit cognitive-like priming, how predictive mechanisms compare to biological semantic encoding and retrieval and what these findings mean for cognitive modeling.

4.1 Methodological Considerations

Handling Missing Data In this experiment, a critical methodological challenge was posed by missing data—40% of the logprob values—requiring the use of multiple imputation to reconstruct a complete dataset. MI is generally preferred over list-wise deletion, as it preserves statistical power by estimating missing values based on the observed distribution. However, when such a substantial portion of data is missing, MI may not fully recover the real distribution, raising questions about representativeness. One consequence is the arousal of variance compression in logprobs values, testified by a shrink in standard deviation. This phenomenon likely occurs predicting missing values based on observed ones, pulls extreme values toward the mean. While this can stabilize estimates in smaller datasets, it may have unintentionally smoothed meaningful variability in the logprobs, affecting true distribution. Indeed, normality test showed a significant departure from normality after imputation was performed. Since semantic priming effects are often subtle, any reduction in variance could have diminished the contrast between related and unrelated conditions, thereby weakening the observable effects. This is consistent with the Wilcoxon test result in the MI dataset, whereas the complete-case analysis did detect a significant effect. The divergence between imputed and complete-case results raises an important methodological question: *did MI impoverished the priming effect, preventing statistical detection, rather than recover lost information?* If the missing data was missing not at random (MNAR)[3] but instead systematic then MI could have incorrectly smoothed meaningful distinctions, masking an effect that was present in the raw data.

Tokenization and Target Reconstruction Bias A significant challenge in the experiment was retrieving log-probabilities for target words due to GPT's subword tokenization. Like other transformer models, it does not always generate words as units, instead break less frequent or morphologically complex words into multiple subword tokens via BPE. This posed a serious obstacle to probability extraction. Further complicating word retrieval was the format of the model's output, which returns a ranked list of predicted tokens along with their logprobs. In cases where the model generated the target as a single token extraction was straightforward. However, when the model split the target across multiple tokens, its overall logprob had to be reconstructed from its individual components—a process that introduces uncertainty. To tackle this challenge, a beam search algorithm was implemented to iteratively reconstruct multi-token targets from the list of predicted subword tokens. While beam search improved reconstruction, it also introduced potential artifacts: (a) some reconstructions may not have perfectly matched the intended target, leading to incorrect logprob values, and (b) certain targets may have been tokenized inconsistently. If tokenization patterns differed systematically between conditions, this could have biased logprob retrieval, introducing confound in the analysis.

Statistical Sensitivity & Priming Detection That being said, divergent findings in MI and complete-case results likely arise from two interrelated factors: (a) variance compression introduced by imputation, which may have diluted the contrast between related and unrelated conditions, and (b) tokenization and reconstruction inconsistencies, which could have added noise to logprob retrieval, particularly in cases where targets were split into multiple tokens. The takeaway is that priming signal drawn from next-word probability retrieval in LLMs may be relatively weak, making it overtly susceptible to distortions introduced by data preprocessing.

4.2 LLMs and Cognitive Modeling

The methodological considerations discussed so far demonstrated how data preprocessing choices and tokenization can influence statistical sensitivity in LLM cognitive experiments. However, these findings also raise deeper conceptual questions: *To what extent do LLMs exhibit semantic priming effects comparable to those observed in human cognition? And if LLMs capture statistical relationship between words, does this also means that they can replicate the cognitive mechanisms underlying human semantic memory?*. To answer such

[3]Unfortunately, there is no surefire way to determine which category data will fall. Random missingness is an assumption that need to be made based upon direct knowledge of the data and its collection mechanisms.

questions, it is possible to draw insights from the two dominant theoretical frameworks that have shaped our understanding on semantic processing: spreading activation theory, as already presented in the introductory section[1] and predictive coding theory *citation*. These models offer different perspectives on how the brain organizes and retrieves meaning, and comparing findings from present work allows to assess the extent to which LLMs approximate cognitive mechanisms. The rest of this section reflects on these themes.

Spreading Activation, Semantic memory and LLMs The spreading activation theory suggests that semantic memory is structured as a network of interconnected concepts, where activation spreads from one node (a word/concept) to related nodes based on semantic similarity and association strength. This model has been widely supported by human psycholinguistic studies. The priming effects detected in the complete-case analysis seems to align with spreading activation framework. LLMs, much like human semantic memory, links concept by encoding statistical co-occurrence patterns between words—though they do it on a considerably larger scale. However, while human priming effects are driven by neural activation spreading across conceptual networks, GPT does not store explicit semantic structures, it instead predicts word based on learned probability distributions. This distinction is crucial: in human cognition, spreading is dynamically modulated by context, prior experience, and attentional control, whereas LLMs' priming emerges from purely statistical dependencies in language data. Current results suggest that semantic priming effects in GPT do not necessarily indicate cognitive-like concept retrieval. The observed priming effect is likely a byproduct of training, rather than a direct parallel to human conceptual activation. Additionally, the lack of a significant effect in MI dataset further challenges the idea that LLM-based priming mirrors human spreading activation dynamics. According to human experiments, priming effects persist despite noise or missing data because activation propagates through associative memory networks. In contrast, the weakening of priming in the imputed dataset suggest a more fragile mechanism.

Predictive Coding and the Mechanisms Underlying Priming in LLMs An alternative perspective fo understanding semantic processing is predictive coding theory [2]. This model suggests that the brain functions as a hierarchical predictive system, continuously generating expectations about incoming sensory input and minimizing prediction errors by adjusting internal models. In this framework, priming occurs because a related prime reduces the uncertainty (prediction error) associated with recognizing the target, leading to faster processing. LLMs, particularly autoregressive models like GPT, operate in a manner structurally similar to predictive coding. They generate words one at a time, updating predictions based on past context. This aligns with the core principle of predictive coding. The log-probabilities extracted in this study measure the system's internal prediction certainty, making them conceptually analogous to prediction error signals in the human brain. The critical difference is that in biological brains, prediction errors lead to adaptive training and belief updating, whereas in LLMs, prediction errors do not modify the model in real-time—they rather influence generation for a short time-window, impacting token selection within the fixed-parameters of the trained model. This means GPT does not actively minimize uncertainty over time. The experimental findings support this distinction. In human coding models, priming effects are expected to persist across different noise conditions because the brain continuously adjust its processing. In contrast, the fragility of GPT's mechanisms suggest that the model's lack a hierarchical learning process that adapts to uncertainty over time. This highlight a fundamental limitations of LLMs: while they approximate prediction driven behavior, they do not engage in error-driven learning during inference, a key component of human cognition. As a result, while priming in LLMs may superficially resembles predictive coding, it does not capture the adaptive mechanisms that govern biological semantic memory. .

5 Conclusion

The results of this study highlight an ongoing debate in cognitive modeling: *to what extent do LLMs exhibit cognitive-like processing?* The presence of a priming effect suggests that LLMs capture meaningful relationships between words, much like spreading activation models, but the disappearance of this effect in the imputed dataset suggests that LLMs' priming is more fragile than human priming. Together, these findings give the impression that LLMs do not simulate human cognition in a mechanistic sense. Instead, they exhibit statistical properties that resemble cognitive processes at the output level but are not necessarily driven by the same underlying computations.

Final Thoughts and Future Directions I firmly believe that while LLMs do not currently replicate human semantic cognition, they offer valuable tools for modeling language-based associations. It is my opinion that the presented approach may be improved and extended:

1. Target predictability: controlling for how predictable a target word is in natural language using frequency norms, surprisals values and entropy-based estimates. This would help disentangle semantic priming from simple word predictability in LLMs.

2. Word frequency effects: since high-frequency words are easily predicted and low-frequency words may be underrepresented in training data, future experiments should systematically control word frequency to determine its impact in priming strength.

3. Contextual influence: LLMs process meaning based on statistical co-occurrence within a fixed context window, which may amplify or suppress subtle priming effects. Future studies should manipulate prime-target distance to assess if context length and structural dependencies influence results. Additionally, future research should explore alternative token-matching strategies, ensuring logprobs reconstruction does not systematically fail with certain word structures. And finally, it should be also considered if modifying LLM architectures—for example, incorporating mechanisms for hierarchical belief updating similar to predictive coding models—would lead to more cognitively plausible representations of meaning. Comparative studies between neural data (e.g. N400 effects) and LLM-based measurements would also clarify whether LLM

priming truly reflect human-like semantic process or remain fundamentally separate in nature.

References

[1] A. M. Collins and E. F. Loftus. A spreading-activation theory of semantic processing. psychological review,. *Psychological Review*, 1975.

[2] K. Friston. A theory of cortical responses. *Philosophical Transactions of the Royal Society B: Biological Sciences*, 2005.

[3] Balota D. A. Neely J. H. Cortese M. J. Cohen-Shikora E. R. Tse C. S. Yap M. J. Bengson J. J. Niemeyer D. Hutchison, K. A. and E. Buchanan. The semantic priming project. *Behavior research methods*, 2013.

[4] M. Kutas and S. A. Hillyard. Reading senseless sentences: Brain potentials reflect semantic incongruity. *Science*, 1980.

[5] D. E. Meyer and R. W. Schvaneveldt. Facilitation in recognizing pairs of words: Evidence of a dependence between retrieval operations. *Journal of Experimental Psychology*, 1971.

[6] J.H. Neely. Semantic priming and retrieval from lexical memory: Roles of inhibitionless spreading activation and limited-capacity attention. *Journal of Experimental Psychology: General*, 1977.