Resource-Rational Noisy-Channel Language Processing: Testing the Effect of Algorithmic Constraints on Inferences

Thomas Hikaru Clark, Jacob Hoover Vigly, Edward Gibson, Roger Levy,

MIT Department of Brain and Cognitive Sciences

Correspondence: thclark@mit.edu

Abstract

Human language use is robust to errors: comprehenders can and do mentally correct utterances that are implausible or anomalous. How are humans able to solve these problems in real time, picking out alternatives from an unbounded space of options using limited cognitive resources? And can language models trained on next-word prediction for typical language be augmented to handle language anomalies in a human-like way? Using a language model as a prior and an error model to encode likelihoods, we use Sequential Monte Carlo with optional rejuvenation to perform incremental and approximate probabilistic inference over intended sentences and production errors. We demonstrate that the model captures previously established patterns in human sentence processing, and that a trade-off between human-like noisy-channel inferences and computational resources falls out of this model. From a psycholinguistic perspective, our results offer a candidate algorithmic model of rational inference in language processing. From an NLP perspective, our results showcase how to elicit human-like noisy-channel inference behavior from a relatively small LLM while controlling the amount of computation available during inference. Our model is implemented in the Gen.jl probabilistic programming language, and our code is available at https://github.com/thomashikaru/noisy_channel_model.

1 Introduction

A fundamental question in psycholinguistics is how comprehenders form interpretations of utterances that they hear or see. Of particular interest are cases where comprehenders form an interpretation despite the presence of errors or anomalies; these instances showcase the robustness of human language comprehension to noise, while simultaneously posing a puzzle — when a comprehender observes an ill-formed or implausible utterance, but still derives a meaning from it, how exactly are these alternative interpretations generated and evaluated?

(1) a. The storyteller could turn any incident into an amusing antidote.

b. The test of the devices were carried out before packaging.

In Example 1a, from Ryskin et al. (2021), the word *antidote* is incongruous in context, but is a possible typo or malapropism for a more plausible alternative, *anecdote*. In Example 1b, from Qian and Levy (2023), there is an agreement mismatch between subject and verb, but there is uncertainty about what the correct intended message was because either the subject or verb could be corrected. In all of these cases, comprehenders carry out some form of error correction under uncertainty.

The noisy-channel theory of language processing provides an explanation for human behavior in terms of rational inference (Gibson et al., 2013; Levy, 2008). According to this account, comprehenders have a probabilistic model of how noise can intervene on intended messages, and thus use both the prior probability of messages and the error likelihood when forming interpretations $\mathbf{s}$ from a noisy utterance $\mathbf{u}$, in line with Bayes' Rule:

$$P(\mathbf{s} \mid \mathbf{u}) = \frac{P(\mathbf{u} \mid \mathbf{s})P(\mathbf{s})}{\sum_{\mathbf{s}'} P(\mathbf{u} \mid \mathbf{s}')P(\mathbf{s}')}.$$

However, marginalizing over the space of possible intended messages (the denominator) is typically intractable, inviting the question of how humans may form approximations to this probability distribution. In general, there has been a lack of implemented computational models that simulate noisy-channel processing at the incremental, word-by-word level, and for an open-ended space of alternatives interpretations (as opposed to evaluating some limited set of alternatives). In Section 5, we

Proceedings of the 2025 Conference on Empirical Methods in Natural Language Processing, pages 23660–23673

elaborate on differences between our model and some existing models from the literature (Li and Futrell, 2024b; Li and Ettinger, 2023; Nour Eddine et al., 2024; Meylan et al., 2023).

Some prior work has considered whether large language models (LLMs), given their strong performance at language tasks in general, may exhibit human-like "noisy-channel inference" behavior (Cai et al., 2024). However, it is unclear whether language models trained on next-word prediction are the right model of this behavior; in particular, humans differ from autoregressive LMs in their ability to a) reanalyze previous material in light of new observations (Hanna and Mueller, 2024), b) explicitly model error operations to reason about alternative interpretations of utterances, and c) vary the amount of mental computation devoted to inference in a resource-rational way (Hoover et al., 2023).

In this work, we model language comprehension as solving a probabilistic inference problem: given some noisy utterance $\mathbf{u}$ possibly containing errors, what is the probability distribution over intended sentences $\mathbf{s}$ and the errors that may have intervened on it? We leverage the existing framework of Sequential Monte Carlo (SMC), which provides an incremental and approximate inference algorithm that is well suited to modeling the processing of sentences one word at a time. At the same time, motivated by non-linear, regressive reading behavior in humans (Frazier and Rayner, 1982; Wilcox et al., 2024), we implement a mechanism for reanalysis of previously processed material using MCMC rejuvenation within SMC. We investigate the relationship between noisy-channel inferences and algorithmic constraints, specifically computational resources (number of particles in SMC) and algorithmic inductive biases (the location and type of rejuvenation strategies). In the following sections, we introduce our model and inference algorithm, report two experiments where our model shows a trade-off between algorithmic constraints and noisy-channel behavior, and discuss implications for both cognitive science and NLP.

2 Model

Our model consists of a generative model, subdivided into a language model prior and an error model, and an inference algorithm. The generative model (Figure 1) describes how "noisy" sentences may be generated, and places probability distributions over relevant random variables, while the inference algorithm solves the problem of inverting the generative model (Tenenbaum et al., 2011; Griffiths et al., 2010; Kersten et al., 2004).

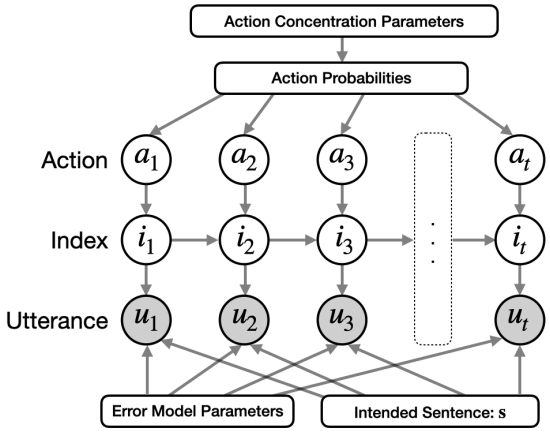


Figure 1: Overview of random variables in the generative model.

2.1 Language Model

This module consists of an autoregressive language model (LM) whose role is to sample words from a vocabulary according to the statistics of typical language usage (i.e., without explicitly modeling errors). In this paper, we report results using the `GPT-2` model (Radford et al., 2019) from HuggingFace (Wolf et al., 2020). Within this framework, the LM expresses a prior $P(\mathbf{s})$ over intended sentences but is not expected to capture noisy-channel behavior on its own, thus we can use a relatively small LM without specialized mechanisms aimed at eliciting reasoning-like behavior (Wei et al., 2023), so long as the LM captures the statistics of typical language well. `GPT-2` has been shown to encode predictability in a way that correlates more strongly with human reading times than larger models (Shain et al., 2024; Kuribayashi et al., 2022; Oh and Schuler, 2023).[1]

The LM module assumes a fixed-size vocabulary $\mathcal{V}$. Since `GPT-2` uses subword tokens, we create custom functions to sample and score words using the LM, subject to the constraint of membership in $\mathcal{V}$. This is achieved by iteratively extracting logits from `GPT-2`, zeroing out the logits of tokens incompatible with any word in $\mathcal{V}$, renormalizing the probability distribution over tokens, and repeating until a valid vocabulary word has been generated.

[1]We use the `llamppl` library (Lew et al., 2023) for language model caching to speed up inference.

This allows words (as delimited by whitespace) of any number of subword tokens to be generated, as long as the words are within $\mathcal{V}$. For the experiments reported below, we set $\mathcal{V}$ to be the intersection of all words in the test suites with the top 5000 most frequent words from the SUBTLEX-US word frequency corpus (Brysbaert and New, 2009). This method, known as locally constrained decoding (LCD), distorts the original `GPT-2` distribution over strings (see Lipkin et al. (2025); Loula et al. (2025) for a discussion). Empirically, the correlation between `GPT-2` surprisal with and without LCD was 0.95 in a set of 500 sentences (see Appendix A for details).

2.2 Error Model

Given a sentence $\mathbf{s}$ sampled from the LM, the error model generates a possibly noisy utterance $\mathbf{u}$ one word at a time. At each time step t, an action a_t is sampled independently from a categorical probability distribution π over the following 6 actions: **normal** production, **insertion**, **skip**, and **form-based**, **semantic**, and **morphological** substitutions. This probability distribution over actions is drawn from a Dirichlet prior with concentration parameter 10 for **normal** and 1 for each of the 5 errors. Because of insertions and deletions, the index of the current intended word within $\mathbf{s}$ may not be equal to t; we use the notation $\text{idx}(t)$ to denote the index in $\mathbf{s}$ that should be produced at time t under **normal**. The model additionally samples a binary `lookahead` random variable from a Bernoulli distribution; this governs whether or not the sentence $\mathbf{s}$ generates 0 or 1 intended words beyond the number of utterance words, and is necessary to allow inferring a **skip** action, which would imply that $\mathbf{s}$ is longer than $\mathbf{u}$.

At time t, given $\mathbf{s}_{\text{idx}(t)}$ and a_t, the error model generates the output word by applying symbolic rules. For the **normal** action, the output word will simply be $\mathbf{s}_{\text{idx}(t)}$ itself. For **skip**, the output word will be $\mathbf{s}_{\text{idx}(t)+1}$. For **form-based** substitutions, the output word is sampled from a probability distribution over $\mathcal{V}$ where each word's probability is monotonic decreasing in its Levenshtein distance, denoted $\text{Lev}(\cdot,\cdot)$, from $\mathbf{s}_{\text{idx}(t)}$ (Levenshtein, 1965): $P(a \mid b) \propto \beta_1^{\text{Lev(a, b)}}$, where $\beta_1 \sim \text{Beta}(2, 11)$ is a latent variable quantifying how peaked or flat the distribution is, and where $P(a \mid b)$ is clamped to 0 for pairs where $\text{Lev}(a, b) > 5$ or if $a = b$. For **semantic** substitutions, the output word is sampled from a probability distribution over $\mathcal{V}$ where each token's probability is monotonically decreasing in its cosine distance from $\mathbf{s}_{\text{idx}(t)}$ in the `GloVe` semantic embedding space (Pennington et al., 2014): $P(a \mid b) \propto \text{cosineSim(a, b)}^{\beta_2}$, where $\beta_2 \sim \text{Gamma}(6, 1)$ is another latent variable governing the distribution's peakedness, and where $P(a \mid b)$ is clamped to 0 for items outside the 20 closest neighbors or if $a = b$. For insertions, the output word is sampled randomly from the unigram frequency distribution over $\mathcal{V}$, independently of context. For **morphological** substitutions, we apply a grammatical number change to $\mathbf{s}_{\text{idx}(t)}$, changing it from singular to plural or vice versa, assuming both forms are in $\mathcal{V}$, e.g. *kick* $\rightarrow$ *kicks*.

2.3 Inference Algorithm: Sequential Monte Carlo

Given an utterance $\mathbf{u}$, we perform inference on latent variables using Sequential Monte Carlo (Naesseth et al., 2024) with custom rejuvenation proposals (see Appendix B: Algorithms 1, 2, 3). We maintain a set of K particles, $\{x_t^{(i)}\}, i = 1 \ldots K$, each corresponding to a hypothesis about the model state, i.e. the values of all latent random variables in the generative model, denoted as $x_t = (\mathbf{s}_{1:t+\text{lookahead}}, \mathbf{a}_{1:t}, \mathbf{idx}_{1:t}, \pi, \beta_1, \beta_2)$, up to the current time step. Each particle is associated with a weight $w_t^{(i)}$, which, when normalized across particles, serves as an approximation to the probability of the particle's state given the observations (Chopin and Papaspiliopoulos, 2020). We use the set of particles to infer the posterior distribution over states, given a set of observations: $P(x_t \mid \mathbf{u}_{1:t})$. At time t, the algorithm samples a new extended state for each particle, which expresses a hypothesis about $\mathbf{s}_{\text{idx}(t)}$ and a_t. In principle, each particle can now be scored in terms of how well it explains the new observation $\mathbf{u}_t$.

However, due to the symbolic rules in the error model, new particle states randomly sampled from the generative model are likely to be incompatible with the observation, resulting in particles with a probability of zero. We thus use a custom proposal function $q(\cdot)$, which assigns $\mathbf{s}_{\text{idx}(t)}$ heuristically, by either setting it equal to $\mathbf{u}_t$, sampling a form-based or semantic neighbor of $\mathbf{u}_t$, or sampling from the LM-induced next-word distribution given the context $\mathbf{s}_{1:\text{idx}(t)-1}$. Intuitively, this heuristic combines three sources of information that a rational comprehender might use during inference: the linguistic

context, the observation itself, and set of items that resemble the observation. The proposal function then samples an action from the set of actions with non-zero probability of generating $\mathbf{u}_t$. We then apply an importance weight correction in the weight update to offset the bias introduced by this proposal function. The new weight $w_t^{(i)}$ for particle $x_t^{(i)}$ at time t is:

$$w_t^{(i)} = P(\mathbf{u}_t \mid x_t^{(i)}) \frac{P(x_t^{(i)} \mid x_{t-1}^{(i)})}{q(x_t^{(i)} \mid x_{t-1}^{(i)}, \mathbf{u}_t)}$$

This is calculated automatically in Gen based on the specification of the generative function. Particles are resampled at each time step, which resets their weights to a uniform distribution.

We define surprisal as the negative log of the mean particle probability, which itself approximates the conditional probability of an observation in context:

$$P(\mathbf{u}_t \mid \mathbf{u}_{1:t-1}) = \int P(\mathbf{u}_t \mid x_t) P(x_t \mid \mathbf{u}_{1:t-1}) dx_t$$
$$\approx \frac{1}{K} \sum_{i=1}^{K} w_t^{(i)}$$

Intuitively, surprisal is lowest when the current observation is explainable as a high-probability continuation in normal production.

2.4 Rejuvenation

While incremental processing is the default in our model, we also optionally include *rejuvenation* as an algorithmic operationalization of the reanalysis of earlier commitments. Rejuvenation for SMC refers to modifying the random choices of a particle in light of new observations (Gilks and Berzuini, 2001; Doucet et al., 2001; Andrieu et al., 2010). Without rejuvenation, each particle's random choices are never revised; this is problematic in a setting with finite particles, where globally promising particles may be filtered out in favor of locally higher-scoring ones. We speculate that there is a cognitive significance to rejuvenation in the context of rational models of cognition — rejuvenation can bring the inferred posterior distribution closer to the target distribution, but comes at the cost of additional computation, thus providing a way to model a trade-off between the quality of inferences and cognitive effort.

A given rejuvenation proposal function takes an existing particle x_t and returns a modified particle $x_t{}'$, which has different choices for some of the random variables in the particle. One such proposal function is the **Form-based Neighbor Proposal**, which takes an existing particle and a specific index t in $\mathbf{u}$, and proposes a different intended word $s_{\text{idx}(t)}$. We sample this word from the form-based substitution distribution as defined in the error model. The model also proposes a change to the corresponding action a_t, flipping it from **normal** to **form-sub** or vice versa. For example, given the utterance in **??** and a particle which assigns the normal action to all words, the algorithm may propose a new particle which designates the intended word *kicked* in place of *licked*, and whose value for a_3 is **form-sub**. Once the proposal function has generated x'_t, we employ the Metropolis-Hastings algorithm to accept or reject this new particle with the following probability:

$$\frac{P\left(x_t{}'\right)}{P\left(x_t\right)} \cdot \frac{g\left(x_t \mid x_t{}'\right)}{g\left(x_t{}' \mid x_t\right)}$$

Thus rejuvenation moves which result in particles with better scores under the generative model are more likely to be accepted, but rejuvenation proposal functions must be carefully designed to be reversible so that they assign non-zero probability to both transitions $x_t \rightarrow x_t{}'$ and $x_t{}' \rightarrow x_t$ (Neklyudov et al., 2020; Cusumano-Towner et al., 2020).

In addition to the Form-based Neighbor Proposal, we additionally employ the analogous **Semantic Neighbor Proposal** and **Morphological Error Proposal** (these operate identically to the Form-based Neighbor Proposal, except that alternatives are sampled from the distribution over semantic neighbors and morphological errors, respectively), and the **Insertion/Deletion Proposal** (which proposes a different intended sentence that contains either one additional or one fewer word).

We implement two distinct rejuvenation strategies, **second-pass rejuvenation** and **conditional rejuvenation**. Second-pass rejuvenation is performed on all words in the utterance after the entire utterance has been observed, and is parametrized by an `iters` parameter, governing how many iterations of all possible rejuvenation proposals to perform. Meanwhile, conditional rejuvenation is initiated probabilistically, with the probability of rejuvenation depending on the surprisal of the most recently observed word in context relative to its

unigram surprisal $-\log P_{\text{uni}}$:

$$\delta_t = \log P_{\text{uni}}(\mathbf{u}_t) - \log \frac{1}{K} \sum_{i=1}^{K} \exp(w_t^{(i)})$$

$$P_{\text{rejuv}} = \exp(\delta_t)/(1 + \exp(\delta_t))$$

The term δ_t above can be interpreted as an estimate of the negative pointwise mutual information[2] between the context and the observation $\mathbf{u}_t$: positive values mean that it is more surprising in this particular context than would be expected based only on its unigram frequency. We posit this as a plausible signal that there may be an error somewhere in the sentence. Conditional rejuvenation is parametrized by a lookback parameter λ, which governs how far back in the sentence to consider for reanalysis. At each time step and for each particle, conditional rejuvenation is triggered with probability P_{rejuv}, and the observations from time $t - \max(1, \lambda)$ to t are targeted for rejuvenation. Higher values of λ make it more likely that regions farther back in the sentence are reanalyzed.

Intuitively, second-pass rejuvenation uses computation indiscriminately, considering all parts of a sentence for reanalysis; conditional rejuvenation, meanwhile, aims to avoid unnecessary computation when utterance words already make sense in context, focusing effort instead on regions preceding likely errors.

3 Experiment 1: The role of particle count in purely incremental inference

What is the relationship between computational resources and the quality of inference, as measured by the ability to handle anomalous words in a human-like way?

A context c induces some next-word probability distribution $P_{\text{LM}}(w \mid c)$ under some language model LM. Under a LM trained on typical language, a word w_A having high probability $P_{\text{LM}}(w_A \mid c)$ does not imply that, on average, word w_B with high error probability $P_{\text{err}}(w_A \rightarrow w_B)$ will have an elevated probability $P_{\text{LM}}(w_B \mid c)$ compared to other low-probability words, except insofar as such errors $w_A \rightarrow w_B$ are well-attested in the training data. However, there is evidence that humans are less surprised by such errors, compared to completely unrelated or unexplainable errors (Ryskin et al., 2021; Li and Futrell, 2024a). In particular, Ryskin et al. (2021) found a neural index of error correction using EEG data from participants reading linguistic stimuli belonging to one of four conditions (Table 1), in terms of the N400 (Kutas and Hillyard, 1980; Kutas and Federmeier, 2011) and P600 (Osterhout and Holcomb, 1992; Kaan et al., 2000; van Herten et al., 2005) event-related potentials. Errors from which recovery was possible showed a small N400 effect and high P600 effect, while unrelated, difficult-to-repair errors induced a large N400 but smaller P600 effect.

The storyteller could turn any incident into an amusing [BLANK]	
Condition	**Completion**
Normal	*anecdote*.
Ungrammatical	*anecdotes*.
Neighbor	*antidote*.
Unrelated	*hearse*.

Table 1: Experiment 1 materials from Ryskin et al. (2021).

Our generative model explicitly models errors, thereby decomposing the probability of observing such an error into the probability of the intended word in context and the probability of the error taking place. Our inference algorithm approximates the distribution over these latent variables, such as the intended word and type of error, sampled using a set of K particles. Crucially, although SMC performs asymptotically correct inference as $K \rightarrow \infty$, in practice the constraint on number of particles impacts whether the choices that best explain the observation are sampled. Thus there may be cases where inference using a small K leads to qualitatively different results than a larger K.

We pass each sentence of the N=504 sentences from Ryskin et al. (2021) to our inference model. We perform SMC inference with K ranging from 4 to 128 in powers of 2. For this experiment, we do not apply any rejuvenation, in order to evaluate purely incremental inference. For each sentence, we compute incremental surprisal from the noisy-channel model and from the baseline LM, which uses the same restricted-vocabulary generative process as the noisy-channel model, but lacks an error model or SMC inference.

3.1 Results

Figure 2 shows the average noisy-channel surprisal of the critical word across items as a function of

[2]Pointwise mutual information is defined as $\text{pmi}(x; y) = \log \frac{p(x,y)}{p(x)p(y)} = \log \frac{1}{p(x)} - \log \frac{1}{p(x|y)}$

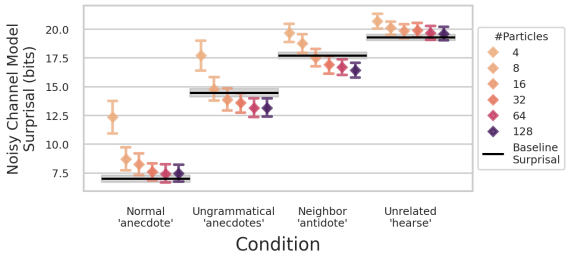


Figure 2: Surprisal from noisy-channel model compared against baseline surprisal, as a function of condition and number of particles. Small colored points denote individual negative particle weights. Error bars and shaded bands denote 95% confidence intervals. In the resolvable error conditions only, given sufficient computation, noisy-channel surprisal is lower than baseline surprisal.

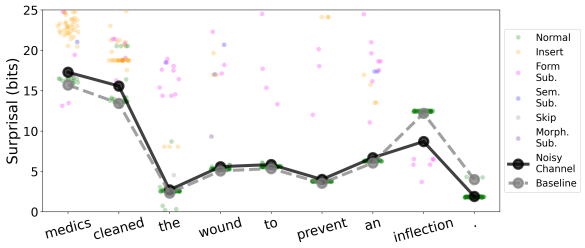


Figure 3: Example sentence containing a Neighbor anomaly comparing noisy-channel and baseline surprisal. Vertical axis is cropped to the range (0, 25) for visual clarity.

condition and number of particles, with baseline surprisal for comparison. As particle count increases, noisy-channel surprisal of the observed word tends to decrease, as expected. More interestingly, comparing the value of noisy-channel and baseline surprisal shows a dissociation between recoverable errors (the Neighbor and Ungrammatical conditions) compared to the other conditions (Normal and Unrelated). For recoverable errors, given sufficient particles (here more than 8), noisy-channel surprisal was on average lower than baseline surprisal, while for the other two conditions, average noisy-channel surprisal asymptotically approached average baseline surprisal but never went below it. For $K = 128$, noisy-channel surprisal was on average 1 to 2 bits lower than baseline surprisal.

An illustrative example of one sentence is shown in Figure 3. For the word "inflection", some particles sample the much more contextually predictable "infection" as the intended word, corresponding to a **form-sub** action. These particles drive down the overall surprisal of this observation in comparison to the baseline LM, whose surprisal is well-approximated by particles that sampled the **normal** action to explain the observed word. The noisy-channel surprisal benefit can also extend to following words or punctuation, as correcting the error can allow better prediction of subsequent material.

We also observe that the it is precisely the recoverable error conditions that exhibit a high posterior probability of an error at the critical word (Figure 4, top panel). While it might initially seem surprising that the Unrelated condition does not induce a high posterior of the action being an error, this is because critical words in the Unrelated condition are not explainable within the generative model of errors; therefore, the model must simply treat them as low-probability continuations of the sentence. Interestingly, this pattern is analogous to that seem in EEG data from Ryskin et al. (2021), in which a strong P600 signal was observed in the recoverable error conditions but not the normal or unrelated error conditions. Turning to model inferences about the intended word, the mean posterior placed on the target word (i.e., the critical word in the Normal condition) increased monotonically as a function of particle count for the two recoverable error conditions, while it remained at zero in the Unrelated condition (Figure 4, bottom panel). This indicates that greater computational resources help the approximate inference algorithm to discover high-probability explanations for noisy sentences, but only if the error is explainable.

4 Experiment 2: The role of algorithmic constraints in reanalysis of potential errors

What is the role of algorithm parameters, in particular those governing rejuvenation, on the similarity of model and human inferences, for sentences which invite reanalysis (as opposed to purely incremental processing)? We address this question using the materials of Qian and Levy (2023), where participants were asked to correct items with agreement errors (Table 2), such that either the subject of the verb could be edited to form a grammatical sentence. We considered a subset of N=120 items with singular subjects and plural verbs. We quantify the verb-edit preference for an item as the ratio of the probability of a verb edit to the probability of an edit at either subject or verb. Human participants made edits to the verb 60% of the time, and edits to the subject 29% of the time, which could

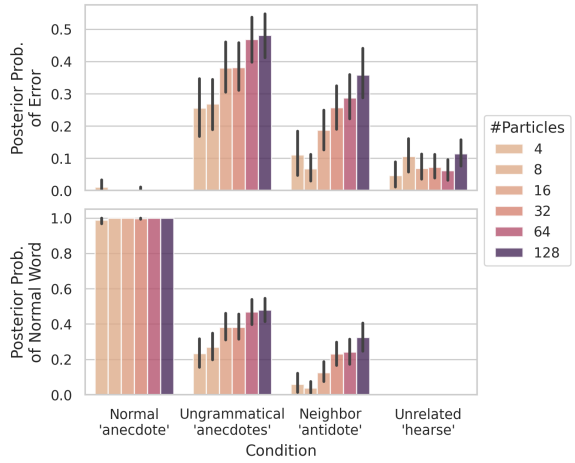


Figure 4: Mean posterior probability placed by the model on the critical word being an error (top) and on the intended word being the word from the Normal condition (bottom), as a function of condition and number of particles. Error bars denote the 95% confidence interval, across items. For the resolvable error conditions, more computation is associated with greater model confidence that the critical word is an error, and higher accuracy at retrieving the 'correct' intended word.

potentially indicate a bias towards editing more recently processed material. Yet they a displayed fine-grained sensitivity to items, making edits that were broadly consistent across individuals, with a mean split-half correlation of 0.81 (95% CI: 0.80-0.81), computed across 500 random 50-50 splits of participants.[3]

Condition	Sentence
Sg Sg Pl	The test of the device were carried out before packaging.
Sg Pl Pl	The test of the devices were carried out before packaging.

Table 2: Experiment 2 materials from Qian and Levy (2023). The condition name denotes whether each of the subject, intervening noun, and verb are singular (Sg) or plural (Pl).

We use our model to run inference on the experimental items using either second-pass rejuvenation or conditional rejuvenation, while systematically varying two key algorithmic parameters. For second-pass rejuvenation, the `iters` parameter controls how many iterations of rejuvenation are performed after the first incremental pass through the sentence. We interpret model results for each sentence as follows: at each of the subject and verb, we compute the posterior probability that the word is an error (i.e., a non-**normal** action). For conditional rejuvenation, we vary the lookback parameter λ, which controls how far back the algorithm proposes rejuvenation moves. For both rejuvenation strategies, we define the model verb-edit preference as $P(a_{\textbf{verb}} = \text{error} \mid \mathbf{u})/(P(a_{\textbf{verb}} = \text{error} \mid \mathbf{u}) + P(a_{\textbf{subject}} = \text{error} \mid \mathbf{u}))$, and compare this to the verb-edit preference across participants for the same item.

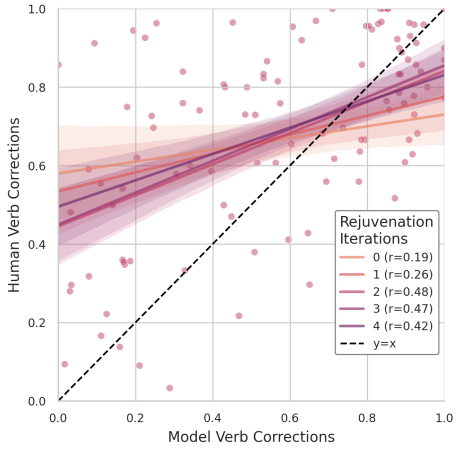
[3]The items with plural subjects and singular verbs had considerably lower split-half reliability, at 0.65 (95% CI: 0.64-0.65). These were not part of our analysis.

4.1 Results

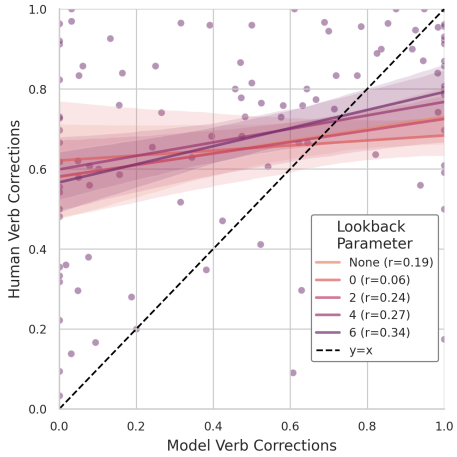
Figure 5a shows model verb-edit preferences using second-pass rejuvenation, plotted against human verb-edit preferences. Pearson's r is shown for each value of `iters`. Compared to the baseline of 0 rejuvenation iterations, adding rejuvenation consistently improved the fit to the human data, with the greatest correlation when `iters` $= 2$. This indicates that up to a point, performing additional iterations of rejuvenation (at the cost of computational resources) yields inferences about error location that more closely resemble those of humans. Figure 5b compares different values of λ within conditional rejuvenation. Pearson's r is shown for each value of λ. Values of $\lambda \geq 2$ fit the human data better than purely incremental inference. However, our results also indicate that even the best model correlation with human inferences is lower than the mean split-half human correlation of 0.81, thus the model does not fully capture all features that humans may use to infer intended meanings (see Limitations).

5 Discussion

Noisy-channel language processing refers to how comprehenders may interpret anomalous utterances inferentially, rather than literally. While this phenomenon is well-studied empirically, there are open questions surrounding what algorithms people may use to arrive at noisy-channel inferences, what effect constraints on computational resources may have on these inferences, and how biases such as incrementality (Altmann and Mirković, 2009; Williams, 2006; Cho et al., 2017; Kamide et al., 2003), recency (Gibson, 1990; Bartek et al., 2011), or resource-rationality (Griffiths et al., 2015; Lieder and Griffiths, 2020) may explain patterns of human comprehension. For example, in Experiment 2,



(a) Second-pass rejuvenation



(b) Conditional rejuvenation

Figure 5: Model verb-edit preference plotted against human verb-edit preference, across items, for second-pass rejuvenation (a) and conditional rejuvenation (b). Darker hues indicate more iterations of rejuvenations. Shaded bands denote 95% confidence intervals. Scatterplot shows datapoints when `iters` $= 3$ (a) and when $\lambda = 6$ (b).

purely incremental inference would be biased towards correcting subject-verb agreement errors by editing the verb, which is when the error becomes apparent, while reanalysis of earlier parts of the sentence might find better edits.

From the perspective of cognitive science and psycholinguistics, our framework provides an implemented algorithmic model of resource-rationality in noisy-channel language processing. Our results demonstrate that qualitatively different patterns of surprisal and inferences emerge by changing the value of parameters that govern computational limitations and reanalysis scope. Previous work has considered computational models of the time-course and neural correlates of noisy-channel inferences (Li and Futrell, 2024b; Li and Ettinger, 2023), or Bayesian models of word recognition under noise for children's speech (Meylan et al., 2023). Work in the predictive coding paradigm has also modeled differences between predictable words, neighbors of predicted words, and other errors (Nour Eddine et al., 2024; Laszlo and Federmeier, 2009), while the effect of memory constraints on the processing of syntactic garden paths has been modeled with approximate SMC inference with varying numbers of particles (Levy et al., 2008). Our model complements such work by showing how an approximate sampling-based algorithm can discover and evaluate alternative interpretations of an utterance; under this model, qualitatively different patterns emerge for recoverable and non-recoverable errors (Experiment 1), similar to the dissociation found by Ryskin et al. (2021). Additionally, our work extends earlier models by incorporating a plausible algorithmic account of reanalysis of earlier material, which we show increases the fit of model inferences to human inferences compared to purely incremental inference (Experiment 2). Finally, our model provides a way to instantiate the notion of resource-rationality in noisy-channel processing at a fine-grained level by varying particle count and iterations of rejuvenations. Future work may consider processing policies where computational resources can dynamically adapt to the difficulty of inference (Hoover et al., 2023), and can evaluate whether experimental manipulations such as speeded judgments or incentives for accuracy can elicit human behavioral profiles that match inference with varying computational resources.

From the perspective of NLP practitioners, our framework of constructing a generative model of errors and performing approximate inference yields a method for eliciting human-like noisy-channel inference behavior from relatively small LLMs like `GPT-2`. By implementing an error model as a generative function, this approach allows for customizing the error model based on domain-specific prior knowledge about the types of errors one expects in the world. Our framework also implements customizable inference, where the amount of computation can be scaled using parameters for the number of particles and the amount of lookback during rejuvenation; these parameters allow a user to navigate the tradeoff between computational resources and the exactness of inference. Previous work has considered the role of SMC algorithms in controlled

generation from language models (Lew et al., 2023; Lipkin et al., 2025; Loula et al., 2025), but here we show how such approaches, combined with an error model informed by domain knowledge, can model human rational inferences and provide robustness against noise. Other work bridging NLP and cognitive science has shown how probabilities from LLMs can be adapted based on alternatives to better model human cognitive processes (Giulianelli et al., 2025; Meister et al., 2024).

6 Conclusion

In this work, we introduce an implemented model of noisy-channel language comprehension using generative functions, probabilistic programming, and Sequential Monte Carlo inference. The model is modular and customizable, allowing different assumptions to be encoded via choice of language model, implementation of the error model, and parameters of the inference algorithm. This allows our model to instantiate varying hypotheses about the computational resource constraints available during inference, which we can manipulate to assess their influence on noisy-channel inferences. Our results indicate that resource constraints can affect whether or not an inferential interpretation of a given anomalous utterance is discovered, and show that augmenting a purely incremental processing algorithm with reanalytical rejuvenation moves can improve fit . Our model offers a candidate algorithmic-level account of rational inference in language processing, and can be used to interrogate open questions in the field, such as what explains the variation between individuals and between items in whether inferential interpretations are formed.

Limitations

We acknowledge some limitations of this work.

Our proposed error model is limited in its expressive power, leaving out some purported basic error operations such as word exchanges (Poppels and Levy, 2016). The implementation of skip errors is currently limited by the `lookahead` random variable, which allows a maximum of one skipped word per sentence to avoid needing to generate multiple words beyond what is needed to explain an utterance. This reflects an inherent tension between incremental inference, which builds up latent variables word by word based on observed utterances, and more flexible global inferences where the number of words in $\mathbf{s}$ may be quite different from the number of words in $\mathbf{u}$. Additionally, the current model assumes that form-based subsitution errors always still generate a vocabulary word (as opposed to a non-word). Thus, the model is at present ill-suited for modeling inferences for degraded language containing non-word errors. We note that humans do a show a lexical bias effect during speech production errors – errors thar result in real words (e.g. "darn bore" $\rightarrow$ "barn door") are more likely than errors that result in non-words (Baars et al., 1975). Future work will extend the error model to also handle non-words (this would imply that any non-vocabulary word in $\mathbf{u}$ will have zero probability of being assigned the **normal** action). While it can generate a wide range of plausible transformations of a given intended sentence, the probabilities it assigns to these transformations are not calibrated to the actual statistics of production errors (for example, our model treats the erroneous pluralization of a singular word as equally likely as the singularization of a plural word). Some sources of uncertainty are encoded as latent variables and included in the inference problem (e.g., the parameters β_1 and β_2 governing the distributions over form-based and semantic substitutions). Other model choices, such as the use of GloVe embeddings or the concentration parameters for the Dirichlet prior, are fixed properties of the model. We leave further exploration of the space of error models, and calibration of its free parameters, to future work.

Another limitation is our language model. We use a single LM as our prior $P(\mathbf{s})$ in our model, but have not thoroughly investigated the sensitivity of inference to different choices of language model or different prompts given to the model. Additionally, we employ token masking to restrict the model vocabulary to a predefined set of frequent words, so the LM does not assign probability mass to the potentially long tail of low-probability utterances in English (Loula et al., 2025; Lipkin et al., 2025). The iterative process of token masking and sampling at each step also creates a slowdown, which could in theory be addressed by utilizing an LM which natively produces probability distributions over words, rather than tokens. The choice of English as the language of our experiments is also a limitation – while English is relatively morphologically simple, thus making it amenable to inference over discrete words, it would be non-trivial to adapt our model to morphologically complex languages where errors might be more readily analyzed at the

morpheme level.

Finally, our approach to rejuvenation contains limitations. Our inference algorithm uses heuristics to propose reanalyses of earlier material. However, it still resembles brute force search in that it proposes changes across a wide range of word positions, dependent on the algorithmic parameter λ. An alternative would be to first identify the most likely positions of errors, then focus rejuvenation effort on those locations, thus reducing unnecessary computation. In either case, our inference algorithm is an approximate inference algorithm – in the limit of infinite particles, the inferred distribution approaches the target distribution, but running inference to convergence may require impractical numbers of particles or iterations of rejuvenation. Future work can consider the relationship between computational resources available to the model and humans places under differing cognitive loads (e.g. under time pressure or attentional demands).

We do not foresee any novel risks introduced by our work, due to our use of existing and publicly accessible datasets and models.

References

Gerry T. M. Altmann and Jelena Mirković. 2009. Incrementality and Prediction in Human Sentence Processing. *Cognitive Science*, 33(4):583–609. _eprint: https://onlinelibrary.wiley.com/doi/pdf/10.1111/j.1551-6709.2009.01022.x.

Christophe Andrieu, Arnaud Doucet, and Roman Holenstein. 2010. Particle Markov chain Monte Carlo methods. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, 72(3):269–342. _eprint: https://onlinelibrary.wiley.com/doi/pdf/10.1111/j.1467-9868.2009.00736.x.

Bernard J. Baars, Michael T. Motley, and Donald G. MacKay. 1975. Output editing for lexical status in artificially elicited slips of the tongue. *Journal of Verbal Learning & Verbal Behavior*, 14(4):382–391. Place: Netherlands Publisher: Elsevier Science.

Brian Bartek, Richard L. Lewis, Shravan Vasishth, and Mason R. Smith. 2011. In search of on-line locality effects in sentence comprehension. *Journal of Experimental Psychology: Learning, Memory, and Cognition*, 37(5):1178–1198. Place: US Publisher: American Psychological Association.

Marc Brysbaert and Boris New. 2009. Moving beyond Kučera and Francis: A critical evaluation of current word frequency norms and the introduction of a new and improved word frequency measure for American English. *Behavior Research Methods*, 41(4):977–990.

Zhenguang G. Cai, Xufeng Duan, David A. Haslett, Shuqi Wang, and Martin J. Pickering. 2024. Do large language models resemble humans in language use? *arXiv preprint*. ArXiv:2303.08014 [cs].

Pyeong Whan Cho, Matthew Goldrick, and Paul Smolensky. 2017. Incremental parsing in a continuous dynamical system: sentence processing in Gradient Symbolic Computation. *Linguistics Vanguard*, 3(1). Publisher: De Gruyter Mouton Section: Linguistics Vanguard.

Nicolas Chopin and Omiros Papaspiliopoulos. 2020. Particle Filtering. In Nicolas Chopin and Omiros Papaspiliopoulos, editors, *An Introduction to Sequential Monte Carlo*, pages 129–165. Springer International Publishing, Cham.

Marco Cusumano-Towner, Alexander K. Lew, and Vikash K. Mansinghka. 2020. Automating Involutive MCMC using Probabilistic and Differentiable Programming. *arXiv preprint*. ArXiv:2007.09871 [stat].

Arnaud Doucet, Nando Freitas, and Neil Gordon, editors. 2001. *Sequential Monte Carlo Methods in Practice*. Springer, New York, NY.

Lyn Frazier and Keith Rayner. 1982. Making and correcting errors during sentence comprehension: Eye movements in the analysis of structurally ambiguous sentences. *Cognitive Psychology*, 14(2):178–210.

Edward Gibson. 1990. Recency Preference and Garden-Path Effects. *Proceedings of the Annual Meeting of the Cognitive Science Society*, 12(0).

Edward Gibson, Leon Bergen, and Steven T. Piantadosi. 2013. Rational integration of noisy evidence and prior semantic expectations in sentence interpretation. *Proceedings of the National Academy of Sciences*, 110(20):8051–8056. Publisher: Proceedings of the National Academy of Sciences.

Walter R. Gilks and Carlo Berzuini. 2001. Following a moving target—Monte Carlo inference for dynamic Bayesian models. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, 63(1):127–146. _eprint: https://onlinelibrary.wiley.com/doi/pdf/10.1111/1467-9868.00280.

Mario Giulianelli, Sarenne Wallbridge, Ryan Cotterell, and Raquel Fernández. 2025. Incremental Alternative Sampling as a Lens into the Temporal and Representational Resolution of Linguistic Prediction.

Thomas L. Griffiths, Nick Chater, Charles Kemp, Amy Perfors, and Joshua B. Tenenbaum. 2010. Probabilistic models of cognition: exploring representations and inductive biases. *Trends in Cognitive Sciences*, 14(8):357–364.

Thomas L. Griffiths, Falk Lieder, and Noah D. Goodman. 2015. Rational Use of Cognitive Resources: Levels of Analysis Between the Computational

and the Algorithmic. *Topics in Cognitive Science*, 7(2):217–229.

Michael Hanna and Aaron Mueller. 2024. Incremental Sentence Processing Mechanisms in Autoregressive Transformer Language Models. *arXiv preprint*. ArXiv:2412.05353 [cs].

Jacob Louis Hoover, Morgan Sonderegger, Steven T. Piantadosi, and Timothy J. O'Donnell. 2023. The Plausibility of Sampling as an Algorithmic Theory of Sentence Processing. *Open Mind*, 7:350–391.

Edith Kaan, Harris , Anthony, Gibson , Edward, , and Phillip Holcomb. 2000. The P600 as an index of syntactic integration difficulty. *Language and Cognitive Processes*, 15(2):159–201. Publisher: Routledge _eprint: https://doi.org/10.1080/016909600386084.

Yuki Kamide, Gerry T. M Altmann, and Sarah L Haywood. 2003. The time-course of prediction in incremental sentence processing: Evidence from anticipatory eye movements. *Journal of Memory and Language*, 49(1):133–156.

Daniel Kersten, Pascal Mamassian, and Alan Yuille. 2004. Object Perception as Bayesian Inference. *Annual Review of Psychology*, 55(Volume 55, 2004):271–304. Publisher: Annual Reviews.

Tatsuki Kuribayashi, Yohei Oseki, Ana Brassard, and Kentaro Inui. 2022. Context Limitations Make Neural Language Models More Human-Like. In *Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing*, pages 10421–10436, Abu Dhabi, United Arab Emirates. Association for Computational Linguistics.

Marta Kutas and Kara D. Federmeier. 2011. Thirty Years and Counting: Finding Meaning in the N400 Component of the Event-Related Brain Potential (ERP). *Annual Review of Psychology*, 62(Volume 62, 2011):621–647. Publisher: Annual Reviews.

Marta Kutas and Steven A. Hillyard. 1980. Reading Senseless Sentences: Brain Potentials Reflect Semantic Incongruity. *Science*, 207(4427):203–205. Publisher: American Association for the Advancement of Science.

Sarah Laszlo and Kara D. Federmeier. 2009. A Beautiful Day in the Neighborhood: An Event-Related Potential Study of Lexical Relationships and Prediction in Context. *Journal of memory and language*, 61(3):326–338.

V. Levenshtein. 1965. Binary codes capable of correcting deletions, insertions, and reversals. *Soviet physics. Doklady*.

Roger Levy. 2008. A Noisy-Channel Model of Human Sentence Comprehension under Uncertain Input. In *Proceedings of the 2008 Conference on Empirical Methods in Natural Language Processing*, pages 234–243, Honolulu, Hawaii. Association for Computational Linguistics.

Roger Levy, Florencia Reali, and Thomas Griffiths. 2008. Modeling the effects of memory on human online sentence processing with particle filters. In *Advances in Neural Information Processing Systems*, volume 21. Curran Associates, Inc.

Alexander K. Lew, Tan Zhi-Xuan, Gabriel Grand, and Vikash K. Mansinghka. 2023. Sequential Monte Carlo Steering of Large Language Models using Probabilistic Programs. *arXiv preprint*. ArXiv:2306.03081 [cs].

Jiaxuan Li and Allyson Ettinger. 2023. Heuristic interpretation as rational inference: A computational model of the N400 and P600 in language processing. *Cognition*, 233:105359.

Jiaxuan Li and Richard Futrell. 2024a. Decomposition of surprisal: Unified computational model of ERP components in language processing. *arXiv preprint*. ArXiv:2409.06803 [cs].

Jiaxuan Li and Richard Futrell. 2024b. An information-theoretic model of shallow and deep language comprehension. *arXiv preprint*. ArXiv:2405.08223 [cs, math].

Falk Lieder and Thomas L. Griffiths. 2020. Resource-rational analysis: Understanding human cognition as the optimal use of limited computational resources. *Behavioral and Brain Sciences*, 43:e1. Publisher: Cambridge University Press.

Benjamin Lipkin, Benjamin LeBrun, Jacob Hoover Vigly, João Loula, David R. MacIver, Li Du, Jason Eisner, Ryan Cotterell, Vikash Mansinghka, Timothy J. O'Donnell, Alexander K. Lew, and Tim Vieira. 2025. Fast Controlled Generation from Language Models with Adaptive Weighted Rejection Sampling. *arXiv preprint*. ArXiv:2504.05410 [cs].

João Loula, Benjamin LeBrun, Li Du, Ben Lipkin, Clemente Pasti, Gabriel Grand, Tianyu Liu, Yahya Emara, Marjorie Freedman, Jason Eisner, Ryan Cotterell, Vikash Mansinghka, Alexander K. Lew, Tim Vieira, and Timothy J. O'Donnell. 2025. Syntactic and Semantic Control of Large Language Models via Sequential Monte Carlo. *arXiv preprint*. ArXiv:2504.13139 [cs].

Clara Meister, Mario Giulianelli, and Tiago Pimentel. 2024. Towards a Similarity-adjusted Surprisal Theory. *arXiv preprint*. ArXiv:2410.17676 [cs].

Stephan C. Meylan, Ruthe Foushee, Nicole H. Wong, Elika Bergelson, and Roger P. Levy. 2023. How adults understand what young children say. *Nature Human Behaviour*, 7(12):2111–2125. Publisher: Nature Publishing Group.

Christian A. Naesseth, Fredrik Lindsten, and Thomas B. Schön. 2024. Elements of Sequential Monte Carlo. *arXiv preprint*. ArXiv:1903.04797 [stat].

Kirill Neklyudov, Max Welling, Evgenii Egorov, and Dmitry Vetrov. 2020. Involutive MCMC: a unifying framework. In *Proceedings of the 37th International Conference on Machine Learning*, volume 119 of *ICML'20*, pages 7273–7282. JMLR.org.

Samer Nour Eddine, Trevor Brothers, Lin Wang, Michael Spratling, and Gina R. Kuperberg. 2024. A predictive coding model of the N400. *Cognition*, 246:105755.

Byung-Doh Oh and William Schuler. 2023. Why does surprisal from larger transformer-based language models provide a poorer fit to human reading times? *Transactions of the Association for Computational Linguistics*, 11:336–350. Place: Cambridge, MA Publisher: MIT Press.

Lee Osterhout and Phillip J Holcomb. 1992. Event-related brain potentials elicited by syntactic anomaly. *Journal of Memory and Language*, 31(6):785–806.

Jeffrey Pennington, Richard Socher, and Christopher Manning. 2014. GloVe: Global Vectors for Word Representation. In *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 1532–1543, Doha, Qatar. Association for Computational Linguistics.

Till Poppels and Roger P. Levy. 2016. Structure-sensitive Noise Inference: Comprehenders Expect Exchange Errors. *Proceedings of the Annual Meeting of the Cognitive Science Society*, 38(0).

Peng Qian and Roger Philip Levy. 2023. Comprehenders' Error Correction Mechanisms are Finely Calibrated to Language Production Statistics.

Alec Radford, Jeffrey Wu, Rewon Child, David Luan, Dario Amodei, and Ilya Sutskever. 2019. Language Models are Unsupervised Multitask Learners. *OpenAI blog*, 1(8):9.

Rachel Ryskin, Laura Stearns, Leon Bergen, Marianna Eddy, Evelina Fedorenko, and Edward Gibson. 2021. An ERP index of real-time error correction within a noisy-channel framework of human communication. *Neuropsychologia*, 158:107855.

Cory Shain, Clara Meister, Tiago Pimentel, Ryan Cotterell, and Roger Levy. 2024. Large-scale evidence for logarithmic effects of word predictability on reading time. *Proceedings of the National Academy of Sciences*, 121(10):e2307876121. Publisher: Proceedings of the National Academy of Sciences.

Joshua B. Tenenbaum, Charles Kemp, Thomas L. Griffiths, and Noah D. Goodman. 2011. How to Grow a Mind: Statistics, Structure, and Abstraction. *Science*, 331(6022):1279–1285.

Marieke van Herten, Herman H. J. Kolk, and Dorothee J. Chwilla. 2005. An ERP study of P600 effects elicited by semantic anomalies. *Cognitive Brain Research*, 22(2):241–255.

Jason Wei, Xuezhi Wang, Dale Schuurmans, Maarten Bosma, Brian Ichter, Fei Xia, Ed Chi, Quoc Le, and Denny Zhou. 2023. Chain-of-Thought Prompting Elicits Reasoning in Large Language Models. *arXiv preprint*. ArXiv:2201.11903 [cs].

Ethan Gotlieb Wilcox, Tiago Pimentel, Clara Meister, and Ryan Cotterell. 2024. An information-theoretic analysis of targeted regressions during reading. *Cognition*, 249:105765.

John N. Williams. 2006. Incremental interpretation in second language sentence processing. *Bilingualism: Language and Cognition*, 9(1):71–88.

Thomas Wolf, Lysandre Debut, Victor Sanh, Julien Chaumond, Clement Delangue, Anthony Moi, Pierric Cistac, Tim Rault, Rémi Louf, Morgan Funtowicz, Joe Davison, Sam Shleifer, Patrick von Platen, Clara Ma, Yacine Jernite, Julien Plu, Canwen Xu, Teven Le Scao, Sylvain Gugger, and 3 others. 2020. HuggingFace's Transformers: State-of-the-art Natural Language Processing. *arXiv preprint*. ArXiv:1910.03771 [cs].

A Appendix: Locally Constrained Decoding vs. Base Model

To investigate the degree of distortion of surprisal values introduced by performing locally constrained decoding (LCD) to enforce a restricted vocabulary, we compare word-level surprisal values from `GPT-2` with and without LCD applied. For `GPT-2` without LCD, we sum sub-word token surprisals to calculate word-level surprisals. Surprisals were computed for 504 sentences from Ryskin et al. (2021). The restricted vocabulary was set to be the union of the 5000 most frequent words in the SUBTLEX-US dataset (Brysbaert and New, 2009) and the vocabulary used in the experimental items. Figure 6 shows that surprisal values obtained via LCD have a correlation of 0.95 with the original surprisal values. Qualitatively, LCD has a slight tendency to underestimate surprisal compared to the base model, due to eliminating the long tail of low-frequency possible completions. Based on manual inspection, LCD is most likely to underestimate surprisal for low-frequency words.

B Appendix: Inference Algorithm

Algorithm 1 shows pseudocode for the Sequential Monte Carlo inference algorithm used in our model. The abbreviation MH denotes a Metropolis-Hastings accept-reject step, implemented via the Gen `mh()` function. Algorithm 2 shows pseudocode for the Form-Based Neighbor Proposal. The function formSubProbs() denotes a function

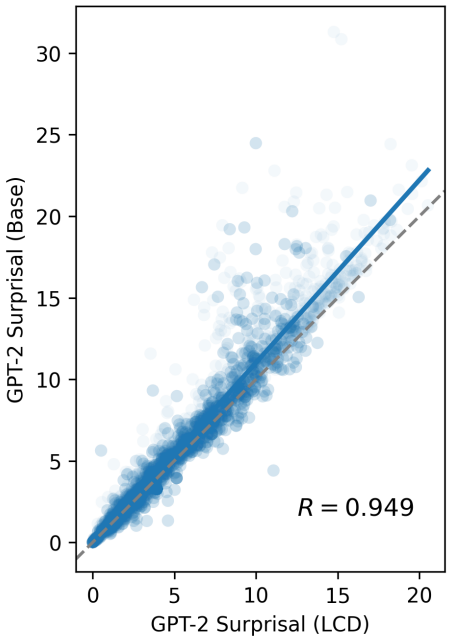


Figure 6: Comparison of surprisal values from locally constrained decoding (LCD) and the GPT-2 base model.

which returns a probability distribution over the vocabulary based on form-based similarity, as described in the main paper. The Semantic Neighbor Proposal and Morphological Error Proposal are highly similar and are thus omitted, simply using functions that return probability distributions based on semantic similarity and morphological similarity, respectively. Algorithm 3 shows pseudocode for the Insertion/Deletion Proposal.

In Algorithms 2 and 3, the notation $x[\cdot]$ denotes accessing the value of a random choice stored by a particle x. Algorithm 3 omits some of the low-level bookkeeping involved in inserting or deleting a word from the intended sentence, which needs to be done carefully to ensure that the resulting sentence still has non-zero probability under the generative model.

C Appendix: Use of Artifacts and Models

We utilize the GPT-2 language model, which has 137M parameters and which is available on Huggingface via the MIT license. Experiments were run on CPUs on our institution's compute cluster.

We also utilize existing datasets from Ryskin et al. (2021) (unknown license) and Qian and Levy (2023) (CC-By Attribution 4.0 International license), which are publicly available via OSF. We use this data purely for evaluating the psycholinguistic explanatory power of our model, and not for training new models or any commercial purposes.

We make our code available to scientific researchers for non-commercial use.

We acknowledge the use of ChatGPT for help with debugging code.

Algorithm 1 Sequential Monte Carlo with Rejuvenation

Inputs: Observations $\mathbf{u}_{1:T}$, number of particles K, Lookback parameter λ, Iterations `iters`
Initialize: For $i = 1, \ldots, K$, sample $x_0^{(i)} \sim P(x_0)$ and set $w_0^{(i)} = 0$
for $t = 1$ to T **do**
  Propagate: $x_t^{(i)} \sim q(x_t \mid x_{t-1}^{(i)}, \mathbf{u}_t)$
  Weight: $w_t^{(i)} \leftarrow P(\mathbf{u}_t \mid x_t^{(i)}) P(x_t^{(i)} \mid x_{t-1}^{(i)}) / q(x_t^{(i)} \mid x_{t-1}^{(i)}, \mathbf{u}_t)$
  Convert weights to normalized probabilities: $\tilde{w}_t^{(i)} \leftarrow \frac{\exp(w_t^{(i)})}{\sum_{j=1}^{K} \exp(w_t^{(j)})}$
  Resample: For $i = 1, \ldots, K$, draw ancestor index $a_t^{(i)} \sim \text{Categorical}(\tilde{w}_{t-1}^{(1)}, \ldots, \tilde{w}_{t-1}^{(K)})$; $x_t^{(i)} \leftarrow x_t^{(a_t^{(i)})}$
  // Conditional Rejuvenation
  $\text{p}_{\text{rejuv}} \leftarrow \sigma(\log \frac{1}{K} \sum_i w_t^{(i)} - P_{\text{uni}}(\mathbf{u}_t))$
  for $i = 1 \ldots K$ **do**
    if not Bernoulli(p_{rejuv}) **then**
      continue
    end if
    for t' **in** shuffle$(\max(1, t - \lambda) \ldots t)$ **do**
      $x_{t'}^{(i)} \leftarrow \text{MH}(x_{t'}^{(i)}, \text{Form Proposal})$
      $x_{t'}^{(i)} \leftarrow \text{MH}(x_{t'}^{(i)}, \text{Semantic Proposal})$
      $x_{t'}^{(i)} \leftarrow \text{MH}(x_{t'}^{(i)}, \text{Morpho Proposal})$
      $x_{t'}^{(i)} \leftarrow \text{MH}(x_{t'}^{(i)}, \text{Ins/Del Proposal})$
    end for
  end for
end for
// Second-Pass Rejuvenation
for $j = 1 \ldots$ `iters` **do**
  for $i = 1 \ldots K$ **do**
    for $t' = 1 \ldots T$ **do**
      $x_{t'}^{(i)} \leftarrow \text{MH}(x_{t'}^{(i)}, \text{Form Proposal})$
      $\ldots$ (*other proposals*)
      $x_{t'}^{(i)} \leftarrow \text{MH}(x_{t'}^{(i)}, \text{Ins/Del Proposal})$
    end for
  end for
end for
Output: Approximate posterior distribution $\{x_t^{(i)}, w_t^{(i)}\}_{i=1}^{K}$ for each $t = 1, \ldots, T$

Algorithm 2 Form-Based Neighbor Proposal

Inputs: Original particle x_t, Target timestep τ
$\text{ps} = \text{formSubProbs}(\mathbf{s}_{\text{idx}(\tau)}, x_t[\beta])$
$v \sim \text{Categorical}(\mathcal{V}, \text{ps})$
$x'_t[\mathbf{s}_{\text{idx}(\tau)}] = v$
Output: New particle x'_t

Algorithm 3 Insertion/Deletion Proposal

Inputs: Original particle x_t, Target timestep τ
$\mathbf{s}_{\text{temp}} \leftarrow x_t[\mathbf{s}]$
$\text{insert} \sim \text{bernoulli}(0.5)$
if insert **then**
  $\text{word} \sim P_{\text{LM}}(\cdot \mid x_t[\mathbf{s}_{1:\text{idx}(\tau)-1}])$
  $\text{insert}(\mathbf{s}_{\text{temp}}, \text{idx}(\tau), \text{word})$
else
  $\text{delete}(\mathbf{s}_{\text{temp}}, \text{idx}(\tau))$
end if
$x'_t[\mathbf{s}] \leftarrow \mathbf{s}_{\text{temp}}$
Output: New particle x'_t