Generative Semantic Transformation Process: A Case Study in Goal Prediction via Online Bayesian Language Inference

Lorenss Martinsons*, John Muchovej*, Ilker Yildirim
Department of Psychology, Yale University, New Haven, CT 06520
{lorenss.martinsons,john.muchovej,ilker.yildirim}@yale.edu
* Denotes equal contribution

Abstract

Language understanding in the real-world occurs through noise — often, lots of noise. What makes language understanding so robust? Here, we address this challenge with a new approach. We cast language understanding as Bayesian inference in a generative model of how world states arise and project to utterances. We develop this model in a case study of action understanding from language input: inferring the goal of an agent in 2D grid worlds from utterances. The generative model provides a prior over agents' goals, a planner that maps these goals to actions, and a 'language-renderer' that creates utterances from these actions. The generative model also incorporates GPT-2 as a noisy language production model. We invert this process with sequential Monte Carlo. In a behavioral experiment, the resulting model, called the Generative Semantic Transformation Process, explains evolving goal inferences of humans as utterances unfold.

Keywords: Bayesian inference; natural language processing; large language models; generative models; goal-directed behavior; semantics; cognitive modeling

Introduction

Imagine, while walking your dog, you run into a friend who recently moved to town. This friend has a keen interest in sampling local eateries. So you give them directions to a cafe you recently visited, but throughout these directions, you also intersperse commands to direct your dog's behavior. Despite this noisy transmission, your friend manages to correctly locate this cafe. In a setting where one knows about the basic layout of their environment and actions they could take, even without an extended familiarity, it's easy to imagine navigating it in spite of noisy utterances. So, how was your friend able to successfully reach the cafe, despite your noisy utterances?

Robust language understanding has long been a core challenge in cognitive science, linguistics, and artificial intelligence. One possibility is that the answer is in distributional semantics, and with the advent of Large Language Models (LLMs), this has become an empirically testable possibility (Zhang et al., 2023). Another possibility is the "translation hypothesis" (Wong et al., 2023), in which language understanding is formalized by causal generative models (instead of purely distributional semantics), and language models are used as a black-box map from utterances onto these generative models. Unfortunately, these proposals will be only as robust as the underlying LLMs used for mapping.

Here, we seek a different approach. We hypothesize that robust language understanding arise from embedding language production within comprehension, by using causal generative models as the glue that holds these two systems together. This proposal builds on key ideas in neuroscience, cognitive science, and artificial intelligence. Recent work in the neural basis of language comprehension provide evidence that hindering production also hinders comprehension, potentially pointing to production playing a critical role in comprehension (Martin, Branzi, & Bar, 2018; Scott, McGettigan, & Eisner, 2009; Silbert, Honey, Simony, Poeppel, & Hasson, 2014; Schomers, Kirilina, Weigand, Bajbouj, & Pulvermüller, 2015; Bonhage, Mueller, Friederici, & Fiebach, 2015). Our proposal to embed the production system within comprehension is inspired by the Rational Speech Act (RSA) framework (Goodman & Frank, 2016); but crucially, we suggest that RSA-like computations operate with respect to representations common with the rest of cognition, formalized using causal generative models. Finally, we take advantage of LLMs as capable production systems, while recognizing that they can also be adapted to model noise in utterances that cannot be captured by just the causal generative models.

In this paper, we introduce the Generative Semantic Transformation Process (GSTP) – a probabilistic architecture for grounding language in "worldly content" by casting this process as Bayesian inference in a generative model of how world states arise and project to utterances in natural language. The model incorporates LLMs within this generative process to mimic a language production system capable of producing humanlike language with occasional noisy utterances. The GSTP model jointly captures the semantic properties of real-world state transitions in a causal generative model and the syntactic generalizations of large pre-trained language models.

We implement an instance of GSTP in a case-study of action understanding from language input and find striking correspondence from our behavioral experiment between the predictions of GSTP and humans, qualitatively and quantitatively. The correspondence indicates that GSTP can leverage pragmatic reasoning and semantic similarities to make accurate estimations despite linguistic noise, in ways similar to humans. In contrast, alternative models, like GPT-3.5 and GPT-4 (tested in January 2024), cannot perform these inferences accurately, nor in human-like ways. We discuss outstanding challenges and outline future directions to scale up the complexity of the modeled environment, language,

and inference mechanisms. Overall, GSTP provides a strong proof-of-concept for action understanding from realistic, but bounded, language which we believe can be straightforwardly extrapolated to a wide range of domains.

Computational Model

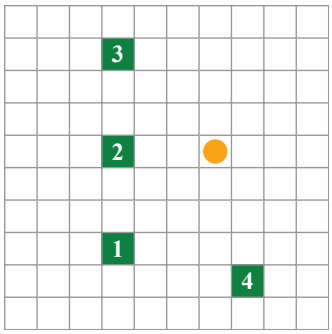

Figure 1: Illustration of the grid world task. The agent is marked as the circle and the potential goal locations are represented by green squares. The task of the observer is to infer the goal location by reading utterances describing the movements of the agent.

We instantiate GSTP using a grid world domain, depicted in Figure 1. Concretely, we simulate an agent navigating a grid world according to an unknown policy. An observer is tasked with inferring the agent's goal, moment-by-moment, from a language signal which noisily captures the agent's actions.

GSTP formalizes meaning as probabilistic states of the agent, including a posterior estimate of its goal location. Taking inspiration from studies of social inference in perceptual contexts (Baker, Saxe, & Tenenbaum, 2009; Baker, Jara-Ettinger, Saxe, & Tenenbaum, 2017), we do so by placing a prior over the agent's goals (locations in the grid world) and embed a Markov Decision Process (MDP) to causally map these goals to actions (through a policy). GSTP then projects the actions of the agent into phrases (e.g., ... *she went 3 south,* ...), and in doing so, incorporates a large language model, Generative Pre-trained Transformer 2 (GPT-2), to account for the noisy nature of natural language production and transmission. Overall, GSTP's architecture combines relatively recent advances in neural generative language modeling (GPT-2) and Bayesian inverse planning (Jara-Ettinger, 2019).

Given this model and an observed unfolding sentence describing the actions of an agent, GSTP employs approximate Bayesian inference to maintain a posterior distribution over the goals of the agent. Inference is implemented using a sequential Monte Carlo algorithm (particle filtering) (Doucet, De Freitas, & Gordon, 2001). This "grounding" of language in a causal generative model of agents produces a robust and flexible framework for making accurate goal predictions even when utterances contain errors or omissions, much like the robust language understanding we observe in humans.

Generative Model

Formalizing meaning via generative models of agents In our task, we formalize the targets of language understanding as generative agent models using the MDP framework. In particular, the MDP is a 10×10 grid world with an agent that must select one of four goal states, as in Figure 1. In the current instantiation of the model, there is only one goal state which has a positive reward, and all others have a negative reward. We place a uniform prior over all possible goal locations. Given the grid world and their goal, there are four actions (N, E, S, W) that the agent may deterministically take to arrive at their goal.

The MDP allows us to causally relate goals to actions by computing an optimal policy through Value Iteration. The model then simulates this policy resulting in a sequence of "atomic-level" actions (e.g., transitioning one tile in the grid world).

Critically, a hierarchical summary of this sequence of actions ultimately informs the utterances generated by GSTP (see Figure 2, "agent model" pane). This hierarchical summary is simply the "chunked actions", obtained by grouping repeated action sequences into tuples of action direction and count, $\text{seq}_i = \langle a_d, a_c \rangle$. For example, if an agent took the following sequence $[N,W,N,N,N,W,W]$, the chunked representation would have four tuples $[\langle N,1\rangle, \langle W,1\rangle, \langle N,3\rangle, \langle W,2\rangle]$, with a chunk length L of 4. These tuples then drive semantic transformation, which in turn conditions utterance generation as described next.

Semantic transformation GSTP links the chunked subtrajectories to semantic representations, that then are projected to natural language utterances. To do so, we convert the hierarchical movements to their latent semantic representations (see Figure 2, "semantic transformation" pane). We define the the semantic value of an utterance, as the Verb Phrase (VP) which concatenates the verb v, count c, and direction d:

$$\text{VP} \rightarrow v + c + d$$

GSTP transforms each action chunk onto this generative VP. In particular, we directly map the direction and count tuple for each chunked action i, $\text{seq}_i = \langle a_d, a_c \rangle$, onto a direction indicator d, a counter indicator c, and a separate movement indicator, verb v, which acts as a prior to indicate what action is taken. This transformation yields a VP that describes the full extent of information within the MDP framework as a latent semantic representation of the behavior of the agent.

Language utterance generation The final step in the generative model is to map semantic representations (signified) to noisy natural language utterances (signifier, or simply, sign; see Figure 2, "language generation" pane)[1].

To do so, we define set of signs: signifier verbs V, signifier counts C, and signifier directions D, that render the semantic representations of verbs, counts and directions, into

[1] Here we borrow the terms "signifier" and "signified" from semiotics

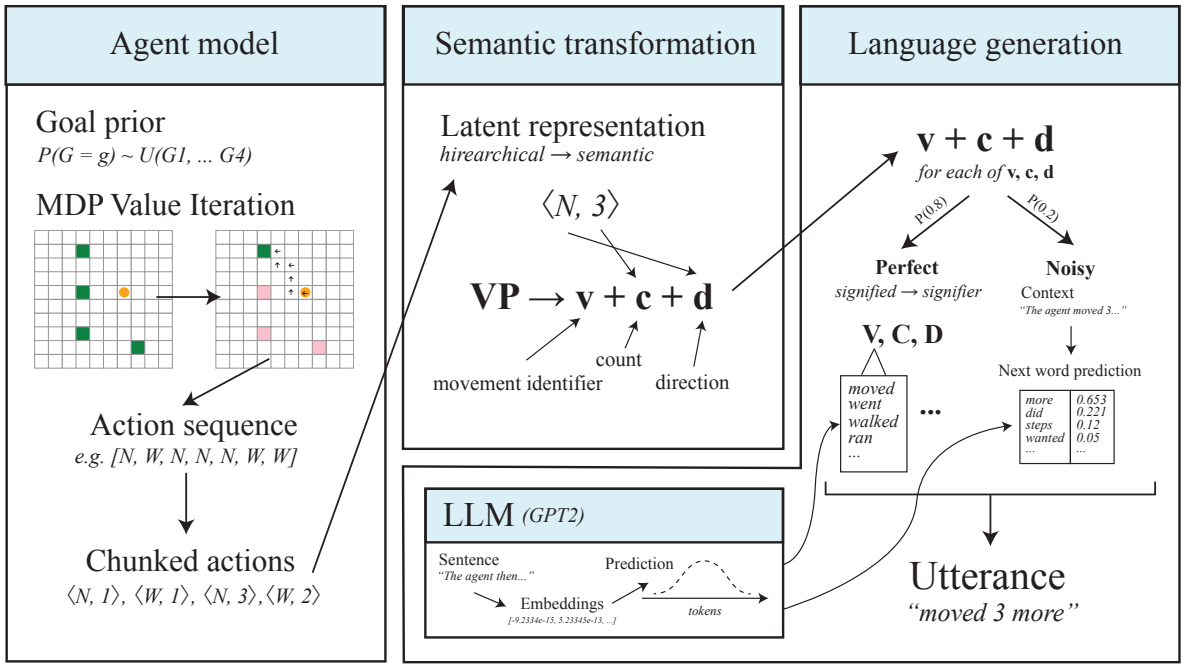

Figure 2: GSTP architecture. The MDP policy samples actions based on a goal prior, generating a chunked action sequence. The sequence gets transformed into semantic and then linguistic representations. The LLM (GPT-2) modulates both the correct signs and the "autocompleted" version of the utterances.

actual utterances. Crucially, the GSTP architecture assumes that sometimes the signifier is imperfectly communicated. To accommodate such imperfections in language transmission, with a small Bernoulli probability (0.2 in our simulations), the generative model produces a noisy sign purely based on the context that has been generated so far (instead of the signified semantic content). We implement such noise using an LLM which yields a sign via next-word prediction given the context generated so far.

LLM Prompt
Here are 4 possible goal positions in a 10x10 grid world given in (x,y) coordinates, where positive y is up, and positive x is right:
1: (4, 3)
2: (4, 6)
3: (4, 9)
4: (8, 2)

The agent started at position (7, 6), and did the following set of movements described by an observer: "It went opened down, walked immediately follows, jumped two went, went 2 west"

Which goal was the agent going to? Give me an answer, the number 1 to 4 corresponding to the goal. Answer:

We also use the same LLM to define the distributions over non-noisy verb, count, and direction signifiers. We now describe this procedure for verbs. We first prompt the LLM with a context as unambiguous as possible ("LLM Prompt", above), and retrieve the distribution over next tokens it predicts. Specifically, we queried GPT-2 (temperature parameter set to 0.4) with the "LLM Prompt" and recorded the distribution over next tokens it predicted. The token with highest probability was " moved". We then computed the Euclidean distance over the vocabulary from " moved" to retrieve the 20 closest tokens based on the premise that the embedding space has semantic coherence (Şenel, Utlu, Yücesoy, Koc, & Cukur, 2018). This yields a set of 20 semantically similar words to the word that GPT-2 has encoded as the signifier for an action. We pruned this set of 20 for anomalous tokens, yielding a set of 15 tokens in V[2]. We then computed the relative log probabilities for these tokens under GPT-2 and created a categorical distribution for the non-noisy signifiers of V.

We repeated a similar procedure to compute the members of count signs C and direction signs D and their corresponding distributions. Crucially, the distributions for C and D are mixtures of categorical distributions. For example the distribution over C is defined over the sets {" one", " two", " three", ...} and {" 1", " 2", " 3", ...}. Similarly, the distribution over D is defined over the sets {" up", " left", " down", " right",} and

[2]There are known tokens in GPT-2 with abnormal behavior acting as particular centroids within the embedding space, see more here.

Start Phrase	VPs				Goal g	Num. VP L
The agent	went 3 left.				2	1
The agent	went one right,	drove 4 south.			4	2
The agent	went 1 up,	seemed three west,	drove 2 up.		3	3
The agent	moved 1 down,	went one more,	walked 3 south.		4	3
The agent	moved two wanted,	went 1 down,	submitted initial west,	seemed two down.	1	4
The agent	walked two there,	ran one up,	walked one left,	moved 2 north.	3	4
The agent	went 2 left,	walked one south,	moved one west,	went two down.	1	4
The agent	went 1 randomly,	walked 1 hoped,	walked two up,	seemed two left.	3	4
The agent	went undercover left.				2	1
The agent	moved to left,	jumped 2 up,	drove 1 west,	drove one up.	3	4
The agent	went 1 top,	moved three west,	did two south.		1	3
The agent	proceeded and sailed,	drove one east,	walked 3 south.		4	3
The agent	went opened down,	walked immediately follows,	jumped two went,	went 2 west.	1	4

Table 1: Samples from GSTP. Red denotes noisy signifiers. These samples are stimuli in the behavioral experiment.

{" north", " west", " south", " east"}. These sets are sampled from a Bernoulli distribution with $p = 0.5$.

Finally, for a noisy signifier in the case of an imperfect transmission, we use the distribution over next-tokens generated by GPT-2 given the constructed sentence so far. Hence, the GSTP model accounts for generating a syntactically relevant, yet semantically, imprecise word in noisy language production or comprehension.

Posterior

The posterior over goals given a sequence of L utterances, $P(G|U_{1:L})$, can be factorized by the utterance tuples $U_i = \langle v, c, d \rangle$ where each sub-utterance is explained either by the distributions of non-noisy signs or "autocomplete" signifiers from the LLM.

$$P(G|U_{1:L}) = \Pi_{g \in G} \Pi_{l=1}^{L} P(U_l|G = g, U_{1:l-1}) P(G = g) \quad (1)$$

$$P(G = g) \sim \mathcal{U}(G_1, ..., G_m) \quad (2)$$

$$P(U_l|G = g, U_{1:l-1}) = \text{GSTP}(g, \text{seq}_l, S_l, U_{1:l-1}) \quad (3)$$

where $g \in G$ are the possible goal locations, $P(G = g)$ is a uniform distribution over these goals (Equation 2), $P(U_l|G = g, U_{1:l-1})$ is the likelihood function induced by the generative process of GSTP (Equation 3), seq_l is the chunked action tuple from the MDP, and S_l is the sentence accumulated so far.

Inference

We approximate the posterior in Equation 1 using particle filtering. This maintains a posterior over the goal location as each utterance is observed. We use 200 particles and return 20 unweighted samples of the posterior over goal locations at each time step. We compare the average of these samples to behavior. We illustrate this posterior at each time point in Figure 3 for an example sentence, showing GSTP's online goal inference through language. We implemented GSTP, including the generative model and inference procedure, using Gen.jl, a state-of-the-art probabilistic programming system (Cusumano-Towner, Saad, Lew, & Mansinghka, 2019), which incorporated GPT-2 using Transformers.jl (Cheng, 2023).

Simulation details For simulating the generative model, we set the Boltzmann parameter for the MDP policy to 1.5, the temperature for GPT-2 to 0.4, language imperfection probability of 0.2, and add a resampling step after each step with a threshold of 0.5.

Alternative models To critically evaluate GSTP, we also tested GPT-3.5 and GPT-4 on the same utterances, given the "LLM Prompt" (above). To do so, we experimentally prompted the models on the stimuli, sampling 5 outputs per each additional utterance, and calculating the posterior distribution over these samples.

Behavioral Experiment

To test GSTP, we conduct an experiment where participants are presented with a grid world, like in Figure 1 – where participants are shown the initial location of an agent and possible goal locations. They then observe an unfolding sentence describing the sequence of actions the agent took (Table 1), without a visual presentation of those actions. At each step of the unfolding, participants respond by indicating which of the possible goal locations they believe the agent to be heading towards.

Participants 16 U.S. participants were recruited on Prolific and view each trial (n=16 participants per sentence).

Stimuli The stimuli consisted of an image of the environment, much like the grid world presented in Figure 1 with all goal locations as buttons. This image persisted across trials, while a sentence at the bottom of the screen is incrementally revealed. For each time t, a new utterance was presented. When the sentence finished, a new sentence was presented, again part by part. In total, 13 sentences (sets of utterances) were sampled from the generative model to be used as stimuli.

Figure 3: Probability distributions over goal locations for humans, GSTP, and GPT-4 on two example sentences from our stimuli set. (Note that GSTP uses GPT-2 as its noise model and to estimate its signifier distributions.)

The stimuli were then randomly shuffled for each participant, except for the first sentence that involved a simple sentence "went 3 left" as a primer.

Procedure Each time a new utterance within a sentence is revealed, the participants were instructed to click on the goal they believed to be the agent's desired goal. When they clicked on a goal location, the experiment proceeded to the next utterance of the sentence (alongside the already unfolded portion of the sentence). Participants did not know how many utterances would be presented in a sentence, and the agent remained in the start location throughout the task. When all utterances in a sentence were revealed, participants moved to the next sentence (starting from the first utterance of that sentence). In total, each participant make goal inferences over 40 utterances.

Results

We now report the performance of the models and how they compare with humans on our task of goal inference with language input. First, we find that both the GSTP model and humans were 100% accurate in inferring the correct goal after the whole sentence was uttered. However, the average probability assigned to the correct target differed, with GSTP being more certain than humans: 0.89 for GSTP, and 0.79 for human participants. We generally saw more variance in the human behavior relative to GSTP.

In Figure 3, we qualitatively evaluate GSTP and GPT-4 against human behavior by providing detailed results from two example sentences in our stimuli set. On the first example, where the reward is actually at location 3 (the top part of Figure 1), as soon as the utterance suggests going up (*jumped 2 up*), GSTP and humans become quite confident about the reward location. This is not the case for GPT-4. A similar pattern emerges in the second example on this figure, with a reversal of the inferred final location observed in humans and GSTP but not in GPT-4.

To quantify these results, we correlate the average human responses and model predictions at the level of individual trials (utterance-level comparisons; Figure 4). We see that the inferred goal locations by GSTP and humans are substantially correlated ($R^2 = .69$), with still more variance to explain. We also see that humans tend to guess location 2 more often than the model. Critically, the model significantly outperforms GPT-3.5 which is decoupled from behavior ($R^2 = 0.0, p < .001$ for pairwise comparison). (We were not able to evaluate GPT-4 to limited access.)

Finally, we also quantify our results by measuring the KL distance between the probability distributions of model predictions on each sentence and human average responses. In Figure 5, we find that GSTP accounts for the goal inferences in certain sentences to a greater degree than others. Overall, GSTP, on the average, significantly outperforms a model that predicts goals at random (average GSTP: 0.15; random: 0.35) and interestingly, both GSTP and the random baseline outperforms the GPT-4 and GPT-3.5, which perform poorly due to idiosyncratic tuning. To quantify the effect of an imbalanced prior over goal 2, we saw that by running a subsequent augmentation of GSTP by placing a 10% higher prior on goal 2, we saw a reduction in the average KL to 0.12.

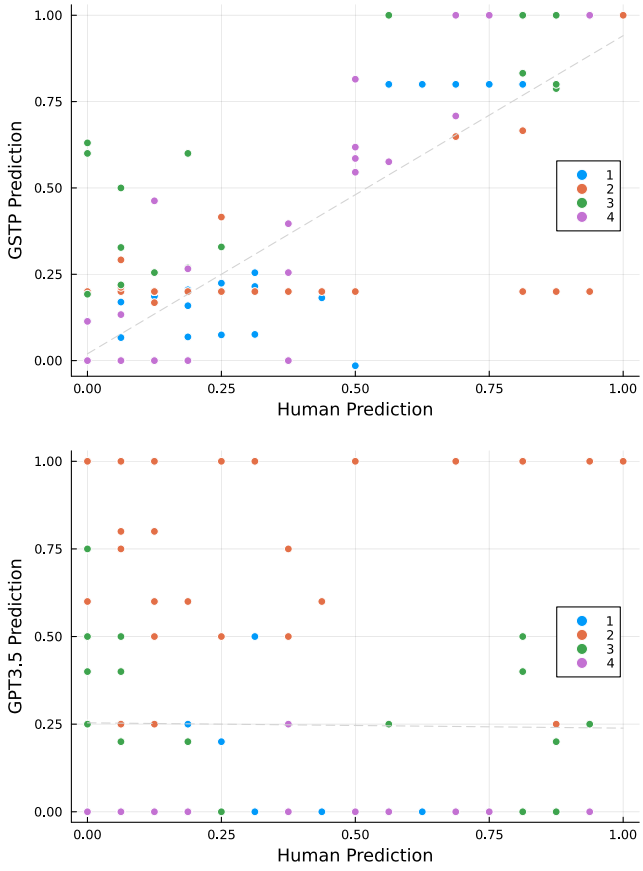

Figure 4: Probability assigned for each goal for each utterance. Top: GSTP ($R^2 = 0.69$), bottom: GPT-3.5 ($R^2 = 0.0$). Color indicates the goal being predicted.

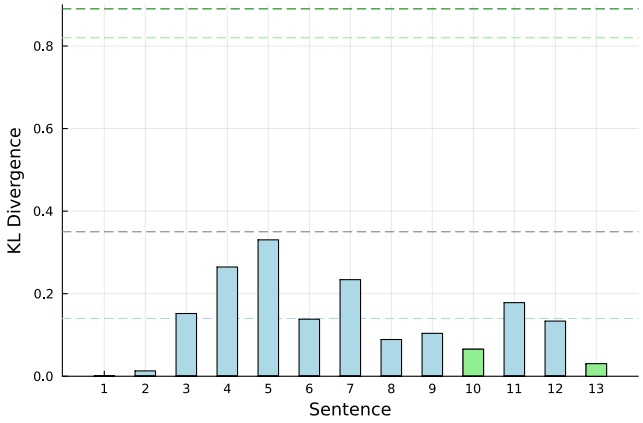

Figure 5: KL divergence of probability distributions for each of the sentences. The gray line is the average KL for a randomized instance of distributions, light blue: GSTP; green: GPT-3.5; light green: GPT-4. Green bars: the two sentences in Figure 3

Discussion

In this work, we presented a probabilistic architecture for grounding language in "worldly content". Our architecture performs Bayesian inverse planning about agents based on language input, and leverages LLMs for imperfect language production. We tested our architecture in the context of a grid world navigation task where participants had to infer goal location from utterances in a piecemeal manner, like GSTP did in simulation. Our model predicted participant judgements significantly, suggesting that people update their goal predictions a similar process – whereby noisy language is ignored/-corrected for by a grounding to "worldly content", much in the spirit of GSTP. However, we did also see areas for improvement of the model, namely, the assumption of a uniform prior over the goals. We also saw that both participant data and the model's predictions were generally highly varied, and it's unclear whether the remaining variance can be explained by other phenomena or are just random in their nature. More participants and testing data might yield other significant insights.

Additionally, the GSTP model represents a significant departure from the performance of the best existing language models, such as GPT-3.5 and GPT-4, all while using the relatively computationally efficient GPT-2 as a baseline. Our approach of a Bayesian inference mechanism grounded in a physical model of an environment allows for a much more rich understanding of language in context, also highlighting the importance of multimodal approaches to worldly language modeling.

Understanding even simple goal-directed language relies on pragmatic inferences about causality, intent, and planning. The GSTP model presents a new approach to formalizing language understanding — as Bayesian inference in generative models of how world states arise and project to language. The success of GSTP in mirroring human inference patterns suggests that similar underlying cognitive processes may be at play in human understanding of goal-directed language. Moreover, the modularity of GSTP allows for a desirable flexibility in many domains of language. We believe that the nature of mapping hierarchical representations to latent semantic content, and then to noisy natural language can explain a lot of domain-specific behavior in language production and understanding. Adapting the model to language understanding in other domains is of immediate interest; developing more efficient inference procedures is also of interest.

We consider our work as a proof-of-concept for a new approach to language understanding, with several significant challenges remaining. These include scaling the model to more complex environments, more sophisticated language use, and considering communicative contexts (Degen, 2023). Overall, our research highlights the profound potential for combining Bayesian methods and language models to bridge the gap between the practical use of language to real-world contexts.

References

Baker, C. L., Jara-Ettinger, J., Saxe, R., & Tenenbaum, J. B. (2017). Rational quantitative attribution of beliefs, desires and percepts in human mentalizing. *Nature Human Behaviour*, *1*(4), 0064.

Baker, C. L., Saxe, R., & Tenenbaum, J. B. (2009). Action understanding as inverse planning. *Cognition*, *113*(3), 329–349.

Bonhage, C. E., Mueller, J. L., Friederici, A. D., & Fiebach, C. J. (2015, July). Combined eye tracking and fMRI reveals neural basis of linguistic predictions during sentence comprehension. *Cortex*, *68*, 33–47. Retrieved from http://dx.doi.org/10.1016/j.cortex.2015.04.011 doi: 10.1016/j.cortex.2015.04.011

Cheng, P. (2023). *Transformers.jl.* Retrieved from https://github.com/chengchingwen/Transformers.jl/

Cusumano-Towner, M. F., Saad, F. A., Lew, A. K., & Mansinghka, V. K. (2019). Gen: A general-purpose probabilistic programming system with programmable inference. In *Proceedings of the 40th acm sigplan conference on programming language design and implementation* (pp. 221–236). New York, NY, USA: ACM. doi: 10.1145/3314221.3314642

Degen, J. (2023). The rational speech act framework. *Annual Review of Linguistics*, *9*, 519–540.

Doucet, A., De Freitas, N., & Gordon, N. (2001). An introduction to sequential monte carlo methods. *Sequential Monte Carlo methods in practice*, 3–14.

Goodman, N. D., & Frank, M. C. (2016, November). Pragmatic language interpretation as probabilistic inference. *Trends Cogn. Sci.*, *20*(11), 818–829. Retrieved from http://dx.doi.org/10.1016/j.tics.2016.08.005 doi: 10.1016/j.tics.2016.08.005

Jara-Ettinger, J. (2019). Theory of mind as inverse reinforcement learning. *Current Opinion in Behavioral Sciences*, *29*, 105–110.

Martin, C. D., Branzi, F. M., & Bar, M. (2018, January). Prediction is production: The missing link between language production and comprehension. *Sci. Rep.*, *8*(1), 1–9. Retrieved from https://www.nature.com/articles/s41598-018-19499-4 doi: 10.1038/s41598-018-19499-4

Schomers, M. R., Kirilina, E., Weigand, A., Bajbouj, M., & Pulvermüller, F. (2015, October). Causal influence of articulatory motor cortex on comprehending single spoken words: TMS evidence. *Cereb. Cortex*, *25*(10), 3894–3902. Retrieved from http://dx.doi.org/10.1093/cercor/bhu274 doi: 10.1093/cercor/bhu274

Scott, S. K., McGettigan, C., & Eisner, F. (2009, April). A little more conversation, a little less action–candidate roles for the motor cortex in speech perception. *Nat. Rev. Neurosci.*, *10*(4), 295–302. Retrieved from http://dx.doi.org/10.1038/nrn2603 doi: 10.1038/nrn2603

Şenel, L. K., Utlu, I., Yücesoy, V., Koc, A., & Cukur, T. (2018). Semantic structure and interpretability of word embeddings. *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, *26*(10), 1769–1779.

Silbert, L. J., Honey, C. J., Simony, E., Poeppel, D., & Hasson, U. (2014, October). Coupled neural systems underlie the production and comprehension of naturalistic narrative speech. *Proc. Natl. Acad. Sci. U. S. A.*, *111*(43), E4687–96. Retrieved from http://dx.doi.org/10.1073/pnas.1323812111 doi: 10.1073/pnas.1323812111

Wong, L., Grand, G., Lew, A. K., Goodman, N. D., Mansinghka, V. K., Andreas, J., & Tenenbaum, J. B. (2023). From word models to world models: Translating from natural language to the probabilistic language of thought. *arXiv preprint arXiv:2306.12672*.

Zhang, Z., Zhang, G., Hou, B., Fan, W., Li, Q., Liu, S., ... Chang, S. (2023, July). Certified robustness for large language models with Self-Denoising. Retrieved from http://arxiv.org/abs/2307.07171