SAMPLED PRAGMATIC ADAPTATION: ZERO-SHOT PRAGMATIC INFERENCE FOR LANGUAGE MODELS

Anonymous authors
Paper under double-blind review

ABSTRACT

Large language models (LLMs) may assign high probability to various responses which are plausible for a query but are also compatible with nearby interpretations of the user's intent. We introduce *Sampled Pragmatic Adaptation* (SPA), a zero-shot inference-time adaptation method that use this ambiguity to improve its output distribution by generating a local neighborhood of query-specific alternative intents. We formulate the adaptation through a minimum-intervention principle: among distributions satisfying query-local marginal constraints, we seek the one that minimally departs from the original distribution in KL divergence. The resulting correction connects pragmatic reasoning under the Rational Speech Act framework, iterative proportional fitting, and prior-shift calibration. We derive utterance- and token-level SPA variants that require neither training nor task-level calibration data, making the approach applicable to open-ended sequential generation. We study the method on tool invocation, a controlled setting in which models must select and construct function calls from conversational context and tool specifications, and introduce StaTI, a static benchmark assembled from existing datasets to evaluate tool invocation independently of downstream execution.

1 INTRODUCTION

Large language models (LLMs) are increasingly used as agents that solve tasks by interacting with an environment (Yao et al., 2022; Qu et al., 2025). In most agentic workflows, the model receives a conversation and the tool documentation as input and it has to decide wether to call a tool, with which arguments and how many times (Patil et al., 2025; Qin et al., 2023; Yao et al., 2024). In this work we refer to this step as *tool invocation* (TI), a task also known as function calling.

A tool call can be seen as the response of the model to the user's *intent* (the *meaning* to be conveyed), which is expressed through a query. The same query can be compatible with more than one tool call, so the model has to resolve the ambiguity. For example, consider a user who asks "Do I need an umbrella in Paris tomorrow?" and an assistant model that has access to two tools, `get_current_weather(city)` and `get_forecast(city,days)`. We hypothesize that the model may give a high probability to the call `get_current_weather(city="Paris")`, since this call is common for questions about the weather in a city. To reduce the chance of such incorrect call, we propose to adapt the LLM's output by generating alternative queries such as "Is it raining in Paris right now?" or "Do I need an umbrella in Paris today?" which are more likely to match the `get_current_weather` function and contrast them with the original one. This contrast, should increase to the probability that the original query "Do I need an umbrella in Paris tomorrow?" results in the call `get_forecast(city="Paris",days=1)`, since such call is likely only for the original query, not for the generated alternative ones.

The above intuition has been studied in collaborative conversation, where participants deal with ambiguous messages by reasoning about each other. In a collaborative scenario, participants should choose messages that make their intent clear to the others participants (H. Paul Grice, 1975). The Rational Speech Act (RSA) model represents this principle based on a Bayesian approach as an iteration between a speaker (in the example, the assistant LLM) and a listener (the user) (Frank & Goodman, 2012). We use this methodology to adapt the output of an LLM at inference time and formalize this adaptation through a minimum-intervention principle: given a prior distribution of the meanings being conveyed, pragmatic reasoning is the consequence of modifying the behavior

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

of the participants only by using the information that alternative intents sampled from that prior provide. Formally, among all joint distributions that quantify the relationship between meanings and responses we select the closest one in KL divergence to the distribution induced by the original LLM and that simultaneously matches the prior distributions of the problem. In this way, this adaptation introduces the least distributional change compatible with the desired priors distribution.

The formalization is motivated from the RSA equations. We show that the RSA iteration can be rewritten both as a scaling of the speaker responses and a shift of the prior distribution (Proposition 3 of Appendix A.1). When the pragmatic scaling parameter of RSA is set to one, this iteration is equivalent to the Iterative Proportional Fitting Procedure (IPFP) (Ireland & Kullback, 1968) which finds the joint distribution that is closest, in Kullback–Leibler divergence, to the one given by the model, under fixed marginal distributions (Fienberg, 1970; Csiszar, 1975). A different rewrite of the iteration was introduced by Estienne (2023), by iteratively finding the prior shift that minimizes the cross-entropy of the adapted model, which is a proper scoring rule (Gneiting & Raftery, 2007) used to calibrate a classifier. The relationship between these iterative adaptation procedures provides a complementary interpretations on the iterative adaptation process, both as an iterative pragmatic reasoning process and as a prior-shift calibration, and has a clear criterion of optimality.

The above adaptation methods assumes access either to the prior distributions of the problem or to a set of unlabeled samples where those priors can be estimated. This information is usually not available for the TI task or for most sequential tasks. We remove this requirement by replacing the unlabeled samples with intents generated by the model, and we use the response distribution of the model for the given query as a proxy for the test prior. With this, the method becomes zero-shot, i.e., it does not require labeled or unlabeled data, or additional training. Moreover, it does not add any content to the context of the model. While these advantages come at the cost of an inference run for each alternative intent, this can be done in parallel limiting the impact in terms of latency.

Since the RSA has not been originally designed for token-by-token generation, we also propose a token-level version of the adaptation, which is applied at each decoding step. This makes the method usable for sequences such as tool calls, where the set of possible responses is not known in advance. Moreover, since the alternative intents are explicit text, the method allows to inspect the weight of each intent in the model's response, adding interpretability to the predicted output.

The proposed method is general and can be applied to most language tasks. However, TI provides a controlled evaluation setting that allows rigorous evaluation and it is central to agentic workflows. Our method requires customized logits postprocessing, and we found that current leaderboard-oriented benchmarks such as the BFCL (Patil et al., 2025) provide a reliable way for comparing models from different providers but make the comparison between new and non-standard methods, common in academic research, rather difficult. For this reason, we build the Static Tool Invocation (StaTI) dataset. StaTI unifies some of the existing open-source tool invocation datasets into an OpenAI-compatible JSON Schema format. Each sample contains the conversation, the tool documentation, and the ground-truth tool calls. Unlike agentic or dynamic evaluations, this evaluation is static and therefore assesses only the agent's ability to perform the TI task, without evaluating its broader capabilities in dynamic or interactive settings. This way, the only aspect of the agent that is evaluated is the TI task resolution capability. In summary, our contributions are the following:

- We introduce *Sampled Pragmatic Adaptation* (SPA), a zero-shot adaptation that generate alternative intents for a given prompt with the same LLM, and use them to adapt its output distribution. We provide the classic utterance-level variant that inherits from the RSA and add a new token-level version by modeling the meanings prior.
- We formulate SPA as a minimum-intervention KL projection under query-local marginal constraints, connecting RSA, IPFP, and prior-shift calibration through a shared response-dependent potential transferred to the target query.
- We establish consistency and asymptotic normality of the sampled pragmatic adaptation, and derive a covariance decomposition characterizing the effects of local curvature and alternative-intent variability.
- We introduce StaTI[1], an open-source static benchmark that unifies existing tool-invocation datasets and enables evaluation independently of downstream tool execution.

[1] https://anonymous-hf.com/a/8szalkc5isxs/

2 Related Work

Pragmatic reasoning and RSA. The RSA model describes communication as a recursive process in which a speaker and a listener reason about each other (Frank & Goodman, 2012; Degen, 2023). In NLP, it has been combined with neural models for machine translation (Cohn-Gordon & Goodman, 2019), image captioning (Cohn-Gordon et al., 2018) and controllable text generation (Shen et al., 2019; Wang & Demberg, 2024), among others. These studies derive alternative meanings from the task or the data. In contrast, we generate them with the same LLM at inference time. Cohn-Gordon et al. (2018) apply RSA at the character level, which is close to our token-level variant. Zaslavsky et al. (2021) interpret the repeated RSA update as alternating maximization of a rate-distortion objective. We give a different view: when the pragmatic scaling parameter is set to one, the repeated update can be seen as an iterative I-projection (IPFP algorithm) or as a prior-shift adaptation calibration process.

Regularized optimal transport and prior adaptation. Adapting a classifier to new class priors is a classic problem, usually solved with unlabeled test data (Saerens et al., 2002; Lipton et al., 2018; Alexandari et al., 2020). For LLMs, several methods correct the bias of the model toward some labels. They use content-free inputs (Zhao et al., 2021), in-domain text (Fei et al., 2023), class prototypes (Han et al., 2022) or a batch of test samples (Zhou et al., 2023). UCPA and SUCPA (Estienne, 2023), in their naive and iterative versions, give a grounded generalization of these methods. The IPFP was proposed to adjust contingency tables to known marginals (Deming & Stephan, 1940), and its convergence and optimality were proven later (Ireland & Kullback, 1968; Fienberg, 1970; Csiszar, 1975). IPFP is also known as Sinkhorn–Knopp matrix scaling (Knopp & Sinkhorn, 1967), and is used to solve entropy-regularized optimal transport (Cuturi, 2013). All these methods work with a fixed set of classes, and they either need task data or use the same inputs for every query. In our case, the alternative intents are specific to each query, and the method works with sequences, where the classes are the possible utterances or tokens.

Methods and benchmarks for TI. LLMs can use external tools by generating structured tool calls as part of their outputs (Schick et al., 2023; Yao et al., 2022). Existing approaches typically improve this capability by finetuning models on datasets containing real or synthetically generated tool-use demonstrations (Patil et al., 2024; Qin et al., 2023; Liu et al., 2024b;a; Lin et al., 2024). At inference time, the model usually reviews and corrects its own output in extra turns (Madaan et al., 2023; Shinn et al., 2023), decodes under syntax constraints (Zhang et al., 2024), or asks the user for clarification whenever user intent is unclear (Wang et al., 2025; Zhang et al., 2025). In line with those studies, we see the arguments of a call as a reflection of the user intent. However, we reason about alternative intents without adding turns to the conversation. To evaluate tool invocation, agentic benchmarks score full trajectories in an environment (Yao et al., 2024; Trivedi et al., 2024; Lu et al., 2025), where tool invocation is mixed with planning and memory. Static benchmarks, instead, score one call at a time given the conversation. Together, BFCL (Patil et al., 2025), APIBench (Patil et al., 2024), API-Bank (Li et al., 2023) and NoisyToolBench (Wang et al., 2025) cover call construction, tool selection, the decision to call a tool or not, API hallucination, and the need for clarification. Other static benchmarks use less informative scores (Wu et al., 2024), depend on real APIs (Qin et al., 2023; Guo et al., 2024), or, in our experience, cannot be run with recent models because the data or code is unavailable (Kokane et al., 2025). StaTI unifies these static benchmarks into a common format, decoupling evaluation from generation for fair comparison of decoding methods.

3 Sampling meanings for adaptation

Let $\Omega^\star$ be the (countable) set of possible tokens sequences and $\mathbf{m}^\star \in \Omega^\star$ denote the sequence of tokens expressing the meaning of the observed query, and let $\mathbf{c} \in \Omega^\star$ collect the remaining conversational context. For any candidate meaning $\mathbf{m} \in \Omega^\star$, a literal language model P_θ defines the response distribution

$$S_0(u \mid \mathbf{m}) := P_\theta(u \mid \mathbf{c} \| \mathbf{m}), \qquad u \in \mathcal{U}. \tag{1}$$

where the $\|$ symbol means concatenation of two sequences and $\mathcal{U}$ is the set of possible responses, which could be a pre-defined set of utterances or the entire tokens vocabulary (see Section 3.3). In particular, without adaptation, the response to the observed query $\mathbf{m}^\star$ is generated from $S_0(\cdot \mid \mathbf{m}^\star)$, which we refer to as the *literal speaker*, in our case, implemented by the language model. This distribution measures how plausible each response is for the target query.

Our starting point to improve the literal speaker is the Rational Speech Act (RSA) view of communication. In this framework, a literal speaker selects a response according to its compatibility with a given meaning, whereas a pragmatic speaker also reasons about how that response distinguishes the intended meaning from other plausible meanings. Thus, a response u should not be preferred merely because it is likely under $\mathbf{m}^\star$: it should also be informative about $\mathbf{m}^\star$ relative to plausible alternatives. For example, a generic response may have high literal probability for the observed query while being equally plausible for many nearby interpretations; such a response carries little information about which interpretation was actually intended. This motivates the query-local question: *What else could the user plausibly have meant in essentially the same conversational context?*

In our case, rather than assuming that these alternatives are supplied by the task definition, we generate them at inference time, using the same LLM. Let $\mathbf{i}$ denote a meta-prompt asking the model to generate plausible alternative meanings for the observed query. We define

$$\mu(\mathbf{m}; \mathbf{m}^\star) := P_\theta(\mathbf{m} \mid \mathbf{i} \| \mathbf{c} \| \mathbf{m}^\star). \tag{2}$$

Note that the distribution $\mu(\cdot; \mathbf{m}^\star)$ should not be interpreted as a global prior over user intents. It is a *query-local pragmatic neighborhood*: it describes meanings that are plausible competitors to the observed meaning $\mathbf{m}^\star$ in the current context. For the finite-dimensional projection below, let $\mathcal{M}_{\mathbf{m}^\star} := \operatorname{supp} \mu(\cdot; \mathbf{m}^\star)$ denote its (finite, but very extensive) support.

Each alternative meaning $\mathbf{m}$ induces its own literal response distribution $S_0(\cdot \mid \mathbf{m})$. Together with the neighborhood distribution, these conditionals define the literal joint model

$$\pi_0(\mathbf{m}, u; \mathbf{m}^\star) := \mu(\mathbf{m}; \mathbf{m}^\star) S_0(u \mid \mathbf{m}). \tag{3}$$

Intuitively, π_0 describes how the unadapted LLM behaves across the local collection of meanings that could plausibly compete with the target query. In addition, we fix the desired response marginal as its literal response distribution

$$\nu(u; \mathbf{m}^\star) := S_0(u \mid \mathbf{m}^\star), \qquad u \in \mathcal{U}, \tag{4}$$

which can be understood as a proxy for the true (not known in practice) marginal.

RSA motivates using these alternative meanings to modify the response behavior, but it does not require discarding the information already encoded by the literal LLM. We therefore adopt the ***minimum-intervention principle*** **in the following way: pragmatic reasoning should alter the literal model only by incorporating the information about the (locally-defined) marginals**.

Formally, we consider the set of joint distributions having meaning marginal $\mu := \mu(\cdot; \mathbf{m}^\star)$ and response marginal $\nu := \nu(\cdot; \mathbf{m}^\star)$,

$$\Pi(\nu, \mu) := \Big\{ \pi \geq 0 : \sum_{u \in \mathcal{U}} \pi(\mathbf{m}, u) = \mu(\mathbf{m}; \mathbf{m}^\star), \quad \sum_{\mathbf{m} \in \mathcal{M}_{\mathbf{m}^\star}} \pi(\mathbf{m}, u) = \nu(u; \mathbf{m}^\star) \Big\}, \tag{5}$$

and select among all joint distributions satisfying these constraints, the one that changes the literal joint model as little as possible in terms of relative entropy:

$$\pi^\star := \operatorname*{arg\,min}_{\pi \in \Pi(\nu, \mu)} D_{\mathrm{KL}}(\pi \| \pi_0)\,. \tag{6}$$

Hence, "minimum intervention" has a precise meaning: the original LLM remains the reference model, and pragmatic reasoning introduces the least distributional change, in the KL sense, compatible with the prescribed marginals. As a consequence, the projection preserves as much as possible of the probabilistic structure of the literal model while enforcing the query-local constraints.

3.1 Zero-shot pragmatic sampled adaptation

Equation 6 is an information-projection problem and can be solved by iterative proportional fitting (IPFP) Csiszar (1975).

Proposition 1 (Optimal potential)**.** *Assume that $S_0(u \mid \mathbf{m}) > 0$, $\mu(\mathbf{m}; \mathbf{m}^\star) > 0$, and $\nu(u; \mathbf{m}^\star) > 0$ on their respective supports. Then the minimum-intervention problem equation 6 admits a unique joint solution of the form*

$$\pi^\star(\mathbf{m}, u) = \mu(\mathbf{m}; \mathbf{m}^\star) \frac{S_0(u \mid \mathbf{m}) \exp(\gamma^\star(u))}{\sum_{u' \in \mathcal{U}} S_0(u' \mid \mathbf{m}) \exp(\gamma^\star(u'))}, \tag{7}$$

where $\gamma^\star$ is unique up to an additive constant and is obtained as the limit of the recursion

$$\gamma^{t+1}(u) := \log \nu(u; \mathbf{m}^\star) - \log \left[\mathbb{E}_{\mathbf{m} \sim \mu(\cdot; \mathbf{m}^\star)} \frac{S_0(u \mid \mathbf{m})}{\sum_{u' \in \mathcal{U}} S_0(u' \mid \mathbf{m}) \exp(\gamma^t(u'))} \right]. \tag{8}$$

The value $\gamma^\star$, the *potential*, is estimated from the local neighborhood and transferred as a query-local pragmatic correction to the literal target speaker, distinct from the joint KL projection itself.

Definition 1 (Pragmatic adaptation)**.** *Given the pragmatic potential $\gamma^\star$ induced by the local projection, we define the adapted response distribution for the observed meaning $\mathbf{m}^\star$ as*

$$S^\star(u \mid \mathbf{m}^\star) := \frac{S_0(u \mid \mathbf{m}^\star) \exp(\gamma^\star(u))}{\sum\limits_{u' \in \mathcal{U}} S_0(u' \mid \mathbf{m}^\star) \exp(\gamma^\star(u'))}. \tag{9}$$

Thus, the literal model remains the anchor for the final prediction, while the alternative meanings determine how its response probabilities are pragmatically reweighted.

The above procedure assumes that the marginal $\mu(\cdot; \mathbf{m}^\star)$ is known, which is not the case in most applications of interest. At inference time, we therefore draw $(\mathbf{m}_1, \ldots, \mathbf{m}_N) \overset{\text{i.i.d.}}{\sim} \mu(\cdot; \mathbf{m}^\star)$ and form the empirical neighborhood $\widehat{\mu}_N(\mathbf{m}') := \frac{1}{N} \sum_{i=1}^N \mathbf{1}\{\mathbf{m}_i = \mathbf{m}'\}$. Replacing the population expectation in equation 8 by its empirical counterpart gives

$$\gamma_N^{t+1}(u) := \log \nu(u; \mathbf{m}^\star) - \log \left[\frac{1}{N} \sum_{i=1}^N \frac{S_0(u \mid \mathbf{m}_i)}{\sum_{u' \in \mathcal{U}} S_0(u' \mid \mathbf{m}_i) \exp(\gamma_N^t(u'))} \right]. \tag{10}$$

Its limit $\gamma_N^\star$ defines the *Sampled Pragmatic Adaptation* (SPA)

$$S_N^\star(u \mid \mathbf{m}^\star) := \frac{S_0(u \mid \mathbf{m}^\star) \exp(\gamma_N^\star(u))}{\sum_{u' \in \mathcal{U}} S_0(u' \mid \mathbf{m}^\star) \exp(\gamma_N^\star(u'))}. \tag{11}$$

The resulting methodology can therefore be summarized as follows: 1) use the target query to sample a set of local alternatives, 2) compute the corresponding $S_0(\cdot|\mathbf{m}_i)$ for each of them, 3) apply equation 10 to infer the smallest pragmatic correction given by the optimal empirical potential $\gamma_N^\star$, and 4) use it to transfer that correction back to the target response distribution through equation 11.

This procedure therefore provides a zero-shot adaptation method based on an optimal minimum-intervention correction, while requiring neither labeled nor unlabeled task-level data nor any update of the model parameters θ. It comes at the expense of generating the alternative meanings and evaluate its response distribution $S_0(\cdot|\mathbf{m}_i)$ under each of them. In accordance with the sampling model above, we obtain N independent completions from the same alternative-meaning prompt $i\|c\|\mathbf{m}^\star$, using independent decoding randomness. These generations can be executed as a single inference batch for computational efficiency, but each completion is conditioned only on the common prompt and not on the other sampled alternatives. Hence, conditional on the observed query and context, $\mathbf{m}_1, \ldots, \mathbf{m}_N$ are i.i.d. samples from $\mu(\cdot; \mathbf{m}^\star)$, as assumed in Appendix B. This query-local construction is what allows the same principle to be used for open-ended generation, where neither a fixed class inventory nor a reusable test batch is naturally available (see Section 3.3).

3.2 Relation to RSA and prior-shift calibration

The construction above is motivated by RSA, but it is useful to distinguish the *conceptual* role of RSA from the *optimization* used in our method. In standard RSA, a literal speaker and a pragmatic listener are recursively coupled: responses are favored when they are not only compatible with a meaning, but also informative for recovering that meaning among competing alternatives. In our setting, the alternatives are not a fixed task-level inventory. They are sampled specifically for the observed meaning $\mathbf{m}^\star$ from the query-local neighborhood $\mu(\cdot; \mathbf{m}^\star)$.

Proposition 3 in Appendix A makes the connection precise. Repeated RSA updates depend strongly on the pragmatic rationality parameter α: the literal term appears as $S_0(u \mid \mathbf{m})^{\alpha^n}$ after n iterations, which progressively sharpens the distribution for $\alpha > 1$ and flattens it for $\alpha < 1$. We therefore focus on the neutral scaling $\alpha = 1$. In this case, the repeated RSA reweighting reduces

to a common response-dependent multiplicative correction and coincides with the IPFP recursion associated with the KL projection in equation 6. Thus, RSA supplies the pragmatic interpretation—responses should be judged relative to competing meanings—whereas the KL projection supplies a well-defined minimum-intervention criterion and a convergent procedure for computing the corresponding correction.

The same recursion also admits a prior-shift interpretation. Consider the exponentially tilted family

$$S_\gamma(u \mid \mathbf{m}) := \frac{S_0(u \mid \mathbf{m}) \exp(\gamma(u))}{\sum_{u' \in \mathcal{U}} S_0(u' \mid \mathbf{m}) \exp(\gamma(u'))}. \tag{12}$$

Proposition 4 in Appendix A shows that the IPFP potential minimizes a conditional cross-entropy over this family and enforces the prescribed response marginal. This connects SPA to previous methods of the literature (Estienne, 2023), which estimate prior corrections from unlabeled samples. The difference is operationally important: those methods assume a task-level sample from a fixed input distribution, whereas we construct a new local sample $\mathbf{m}_1, \ldots, \mathbf{m}_N \sim \mu(\cdot; \mathbf{m}^\star)$ for each test query. We also take $\nu(\cdot; \mathbf{m}^\star) = S_0(\cdot \mid \mathbf{m}^\star)$ rather than imposing a uniform response marginal, since this is a better approximation in practice for the prior token distribution. Appendix B provides consistency and asymptotic normality of the SPA, and derive a covariance decomposition characterizing the effects of local curvature and alternative-intent variability.

3.3 Adaptation for sequences

The SPA applies once a finite response set $\mathcal{U}$ has been specified. We use it in two ways, depending on whether the objects being compared are complete candidate utterances or next-token choices.

Utterance level (U-SPA). When a finite set of candidate utterances can be generated in advance, we let $\mathcal{U} = \{u_1, \ldots, u_M\}$ denote the set of *distinct* candidates. Because the projection requires a probability distribution on $\mathcal{U}$, we use the literal model restricted and renormalized to this set,

$$\bar{S}_0(u_j \mid \mathbf{m}) := \frac{P_\theta(u_j \mid \mathbf{c} \| \mathbf{m})}{\sum_{k=1}^{M} P_\theta(u_k \mid \mathbf{c} \| \mathbf{m})}, \qquad j = 1, \ldots, M. \tag{13}$$

For notational simplicity, in the utterance-level algorithm we use S_0 for this normalized restriction. For the observed meaning $\mathbf{m}^\star$, we first generate the candidate set from the literal model, then sample N alternative meanings $\mathbf{m}_1, \ldots, \mathbf{m}_N \sim \mu(\cdot; \mathbf{m}^\star)$ and evaluate $S_0(u_j \mid \mathbf{m}_i)$ for every pair (i, j). Applying the empirical recursion from equation 10 yields $\gamma^\star_N$, and the candidates are ranked using the adapted target distribution $S^\star_N(\cdot \mid \mathbf{m}^\star)$ in equation 11. Intuitively, the same candidate responses are viewed through several plausible interpretations of the query; responses that are generically likely across many alternatives can then be distinguished from responses whose probability profile is more specific to the target.

Token level (T-SPA). For open-ended generation, the complete response space cannot be enumerated. We therefore apply the same idea at each decoding step, with $\mathcal{U}$ equal to the token vocabulary. Let $\mathbf{w}_{<t} := (w_1, \ldots, w_{t-1})$ denote the generated prefix and define

$$S_{0,t}(w \mid \mathbf{m}) := P_\theta(w \mid \mathbf{c} \| \mathbf{m} \| \mathbf{w}_{<t}), \qquad w \in \mathcal{U}. \tag{14}$$

The prefix itself provides evidence about which sampled alternatives remain relevant. We make this statement precise through an auxiliary latent-meaning model. Conditional on the sampled neighborhood $\mathbf{m}_{1:N}$, introduce a latent index $I \in [1 : N]$ with prior $\Pr(I = i) = 1/N$. The index is fixed along the generated sequence and identifies which sampled alternative is used to explain the prefix. After computation of the potential $\gamma^\star_{N,j}$ at step j, define the alternative-conditioned emission kernel

$$\widetilde{S}^\star_{N,j}(w \mid \mathbf{m}_i) := \frac{S_{0,j}(w \mid \mathbf{m}_i) \exp(\gamma^\star_{N,j}(w))}{\sum_{w' \in \mathcal{U}} S_{0,j}(w' \mid \mathbf{m}_i) \exp(\gamma^\star_{N,j}(w'))}. \tag{15}$$

The sequential model assumes that, conditional on $I = i$ and the already generated prefix, token W_j is drawn from this kernel. Consequently, the relevance weights are the exact posterior in this auxiliary model,

$$\omega_{i,t-1} := \Pr(I = i \mid \mathbf{W}_{<t} = \mathbf{w}_{<t}, \mathbf{m}_{1:N}) = \frac{\prod_{j=1}^{t-1} \widetilde{S}^\star_{N,j}(w_j \mid \mathbf{m}_i)}{\sum_{k=1}^{N} \prod_{j=1}^{t-1} \widetilde{S}^\star_{N,j}(w_j \mid \mathbf{m}_k)}, \qquad \omega_{i,0} = \frac{1}{N}. \tag{16}$$

		API-Bank		BFCL									NoisyToolBench		
		GD	TS	SI	L-SI	MU	L-MU	PA	L-PA	PM	L-PM	IR	IMKI	IMR	IWE
Qwen3-0.6B-FC	Literal	**0.24**	**0.09**	**0.82**	**0.66**	**0.83**	**0.55**	0.67	**0.50**	0.65	**0.58**	0.82	0.42	0.44	0.28
	T-SPA	0.20	0.02	0.80	0.64	0.82	**0.55**	**0.69**	0.44	**0.68**	0.54	0.84	**0.44**	0.42	**0.36**
	U-SPA	0.19	0.02	0.81	**0.66**	0.81	0.53	0.66	**0.50**	0.64	0.50	**0.85**	0.36	**0.46**	0.34
Qwen3-0.6B-Prompt	Literal	**0.14**	**0.09**	0.34	0.27	**0.34**	0.20	**0.05**	**0.00**	**0.04**	0.04	**0.87**	**0.16**	0.15	0.08
	T-SPA	0.12	0.00	0.35	**0.28**	0.28	0.19	0.03	**0.00**	**0.04**	**0.08**	0.82	0.10	0.15	**0.12**
	U-SPA	0.13	0.00	**0.37**	0.27	**0.34**	**0.22**	0.03	**0.00**	**0.04**	**0.08**	0.86	0.14	**0.21**	0.10

Table 1: Exact match per subtask. Within each model, the best value is in **bold**.

Equivalently, these weights admit the recursive Bayes update

$$\omega_{i,t} = \frac{\omega_{i,t-1}\widetilde{S}^{\star}_{N,t}(w_t \mid \mathbf{m}_i)}{\sum_{k=1}^{N} \omega_{k,t-1}\widetilde{S}^{\star}_{N,t}(w_t \mid \mathbf{m}_k)}. \tag{17}$$

Thus an alternative remains influential precisely when the observed prefix is probable under its adapted token-level speaker. This gives the prefix reweighting a probabilistic interpretation rather than treating it as an ad hoc importance score. The latent-index construction is, however, a modeling choice for extending the fixed-query SPA projection to sequential generation; it is not implied uniquely by the utterance-level KL projection.

At step t, conditional on the realized prefix, the vector $\omega_{t-1} = (\omega_{1,t-1}, \ldots, \omega_{N,t-1})$ is a distribution on the sampled alternatives. We set the target response marginal to the literal next-token distribution

$$\nu_t(w; \mathbf{m}^{\star}) := S_{0,t}(w \mid \mathbf{m}^{\star}) \tag{18}$$

and define the step-t seed joint on $\{1, \ldots, N\} \times \mathcal{U}$ by

$$\pi_{0,t}(i, w) := \omega_{i,t-1} S_{0,t}(w \mid \mathbf{m}_i). \tag{19}$$

The token-level correction is the KL projection of $\pi_{0,t}$ onto joints having meaning-index marginal ω_{t-1} and token marginal ν_t. By Proposition 1, its response potential is obtained from the weighted empirical recursion

$$\gamma^{n+1}_{N,t}(w) := \log \nu_t(w; \mathbf{m}^{\star}) - \log \left[\sum_{i=1}^{N} \omega_{i,t-1} \frac{S_{0,t}(w \mid \mathbf{m}_i)}{\sum_{u' \in \mathcal{U}} S_{0,t}(u' \mid \mathbf{m}_i) \exp(\gamma^{n}_{N,t}(u'))} \right]. \tag{20}$$

After convergence, the adapted next-token distribution is

$$S^{\star}_{N,t}(w \mid \mathbf{m}^{\star}) := \frac{S_{0,t}(w \mid \mathbf{m}^{\star}) \exp(\gamma^{\star}_{N,t}(w))}{\sum_{w' \in \mathcal{U}} S_{0,t}(w' \mid \mathbf{m}^{\star}) \exp(\gamma^{\star}_{N,t}(w'))}. \tag{21}$$

The selected token is appended to the prefix, the weights are updated by equation 17, and the procedure is repeated. Conceptually, the utterance-level method compares complete candidate responses once, whereas the token-level method repeatedly asks which alternatives remain compatible with the partial response generated so far.

The fixed-query finite-sample results in Appendix B apply directly to the i.i.d. neighborhood used in the utterance-level projection. At each token step, conditional on the realized prefix and weights, equation 20 is again a KL projection with meaning marginal $(\omega_{1,t-1}, \ldots, \omega_{N,t-1})$ and response marginal ν_t. Across decoding steps, however, these marginals are random and history dependent. The token-level construction should therefore be understood as a recursively defined Bayesian-filtering extension of SPA, not as a consequence of the fixed-query i.i.d. theorem; we do not claim that the same concentration bounds hold unchanged along an entire generated trajectory.

4 Experiments

We evaluate the proposed methods on **StaTI** (Static Tool-Invocation), an open source static function-calling dataset collected from selected samples of three different benchmarks: BFCL (Patil et al., 2025), API-Bank (Li et al., 2023) and NoisyToolBench (Wang et al., 2025). StaTI collects 7,648

samples in 20 subtasks from these three sources. Table 3 in Appendix C shows the complete statistics for these dataset. Each of the 20 tasks contained conversations between an aassitant and a user who eventually produce a query that should trigger (at least once) either a tool invocation or a message that reports a "no tool invocation is possible" error. We create each sample of the StaTI dataset by collecting three items: 1) the conversation up to the query that triggers the function call or the error message, 2) the tool documentation available to the assistant up to that point of the conversation, and 3) the expected list of tool invocations that the assistant should predict. If there is more than one function call in the conversation, as it the case of most multi-turn samples, the conversation is split to produce multiple entries with one function call per entry.

The StaTI ground truth contains either an empty list when no tool should be invoked or one or more acceptable calls, accounting for optional arguments. Predictions are evaluated against these calls without executing them, using tool specifications including argument types, optional fields, and default values. We normalize the heterogeneous specifications of the three source benchmarks to an OpenAI-compatible JSON-Schema format, automatically correcting them when possible and manually resolving remaining ambiguities and inconsistencies.

We compare literal decoding with the two pragmatic methods of Section 3.3, T-SPA and U-SPA. For each (model, sample, seed), we independently sample $N = 8$ alternative queries from the same meta-prompt using the evaluated model and independent decoding randomness. The completions are generated in parallel but conditioned only on the common meta-prompt, and are therefore independent. They are cached and shared by T-SPA and U-SPA, with two seeds per pragmatic configuration. The alternative-meaning prompt is given in Appendix C.

Experiments use `Qwen3-0.6B` under two tool-calling modes (Table 2), sharing the same checkpoint, turn structure, and reasoning handling but differing in how tools are presented and invoked, thereby isolating the tool-call interface. [2]

Table 1 reports exact-match (AST) accuracy for the literal model and the two SPA variants. The reported pragmatic results are averaged over two independent sampling seeds. To reduce the cost of the token-level projection, T-SPA applies top-p masking with $p = 0.9$. All methods use greedy decoding. For U-SPA, we generate eight candidate utterances per sample. Outputs that cannot be parsed as function calls receive an exact-match score of zero, whereas parsable outputs containing no function call are interpreted as an empty list and compared with the ground truth.

The effect of pragmatic adaptation is heterogeneous across tasks and interfaces. In the FC configuration, SPA degrades performance on the two API-Bank subtasks, while its effect on BFCL depends on the subtask. T-SPA improves the non-live Parallel task (PA: $0.67 \rightarrow 0.69$) and Parallel-Multiple task (PM: $0.65 \rightarrow 0.68$), and both variants improve Irrelevance (IR: $0.82 \rightarrow 0.84$ for T-SPA and 0.85 for U-SPA). Conversely, some live BFCL variants deteriorate, most notably Live Parallel-Multiple ($0.58 \rightarrow 0.54/0.50$). On NoisyToolBench, the effect is again mixed: both variants improve IWE ($0.28 \rightarrow 0.36/0.34$), whereas the relative behavior on IMKI and IMR depends on the SPA variant.

The Prompt configuration leads to substantially lower exact-match accuracy than the FC configuration on most BFCL tasks, confirming that the tool-call interface itself has a large effect on performance. Within this more difficult configuration, SPA again produces mixed changes: U-SPA improves, for example, BFCL Simple ($0.34 \rightarrow 0.37$) and NoisyToolBench IMR ($0.15 \rightarrow 0.21$), while other tasks remain essentially unchanged or deteriorate. Overall, these results do not indicate a uniform accuracy gain from pragmatic adaptation; rather, they show that the query-local correction can improve particular tool-invocation regimes while degrading others.

To better understand these differences, Figure 1 reports unparsable outputs, while Figure 2 analyzes structural errors among parsable predictions using the minimum graph edit distance (GED) to a valid expected call. Both figures report median across two seeds and 1st and 3rd quartiles as confidence intervals. Invalid-output failures explain part of the accuracy differences, particularly between FC and Prompt, but SPA also affects the structure of parsable calls. Under FC, T-SPA reduces mean GED on BFCL Multiple ($0.33 \rightarrow 0.29$), Parallel ($1.66 \rightarrow 1.44$), Parallel-Multiple ($1.01 \rightarrow 0.93$), and Irrelevance ($0.54 \rightarrow 0.50$), while increasing it on Simple ($0.27 \rightarrow 0.35$). U-SPA shows a different

[2] Code for reproducing the results is available at `https://anonymous.4open.science/r/StaTI_public-E07E/README.md`.

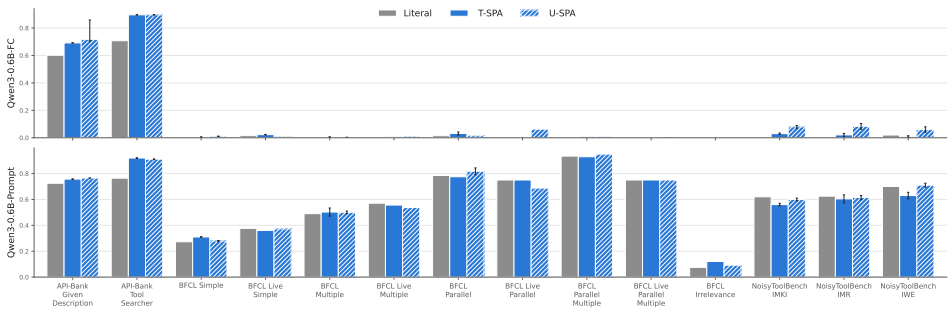

Figure 1: Failures produced for each task.

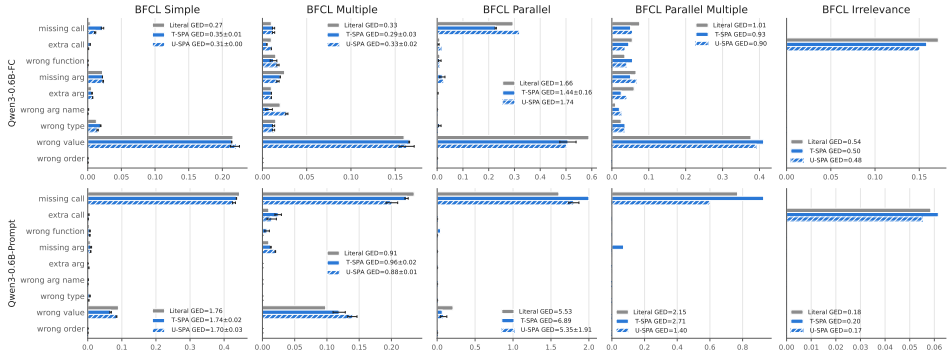

Figure 2: Error types for each model and task.

profile, reducing GED on Parallel-Multiple ($1.01 \rightarrow 0.90$) and Irrelevance ($0.54 \rightarrow 0.48$), but not consistently across tasks.

These results reinforce the picture from exact match: SPA does not produce a uniform improvement, and T-SPA and U-SPA can behave differently on the same task. The gains observed on several multi-call and parallel-call settings suggest that pragmatic contrast with alternative intents can be useful when the output structure is more complex, but the current experiments are not sufficient to establish this as a general property.

5 Conclusion

We introduced Sampled Pragmatic Adaptation (SPA), a zero-shot inference-time method that uses query-local alternative intents to pragmatically reweight an LLM's output distribution. SPA formulates pragmatic adaptation as a minimum-intervention KL projection, connecting RSA-style reasoning, iterative proportional fitting, and prior-shift calibration through a common response-dependent potential. It requires neither task-level calibration data nor parameter updates, admits utterance- and token-level variants, and provides consistency and asymptotic normality guarantees for the fixed-query sampled adaptation.

Experiments on StaTI show that SPA can improve exact match and structural error on some tool-invocation tasks, but its effect is heterogeneous across tasks and variants. This motivates a central open question: when does the query-local pragmatic correction improve downstream task risk? Addressing this question will require better understanding the quality and size of the sampled intent neighborhood, finite-N behavior, and when adaptation should be applied or suppressed. Extending the fixed-query theory to the history-dependent token-level construction is another important direction.

AI USE STATEMENT

Generative AI tools were used for coding, writing, and editorial assistance, but not to originate the research questions, methods, or experimental design. AI-assisted coding tools were used to help plan and implement parts of the experimental code; all generated code was manually reviewed, tested, and validated by the authors.

AI assistance was also used in preparing the manuscript. For the first two sections, the authors provided the intended narrative and a curated bibliography to generate an initial draft, which was subsequently substantially revised by the authors. The methods and experimental sections were written by the authors, with AI used for editorial review and revision. References and BibTeX entries were selected and added manually by the authors.

The mathematical arguments and proofs were developed by the authors. AI assistance was used to help formalize and present some of these arguments as propositions, theorems, corollaries, and proofs. All AI-assisted mathematical content was checked and validated by the authors. The authors take responsibility for the final content of the paper, including all AI-assisted text, code, and mathematical statements.

REPRODUCIBILITY STATEMENT

Code for reproducing the experiments, plots, and tables is available in an anonymous GitHub repository. The corresponding LLM generations are provided as an `outputs.zip` file in the supplementary material, allowing the reported results to be regenerated without rerunning model inference. The code and data used to construct StaTI are additionally available through an anonymous Hugging Face repository. Instructions and links to these resources are provided in the paper and supplementary material.

REFERENCES

Amr Alexandari, Anshul Kundaje, and Avanti Shrikumar. Maximum Likelihood with Bias-Corrected Calibration is Hard-To-Beat at Label Shift Adaptation. In *Proceedings of the 37th International Conference on Machine Learning*, pp. 222–232. PMLR, November 2020. URL https://proceedings.mlr.press/v119/alexandari20a.html.

Reuben Cohn-Gordon and Noah Goodman. Lost in Machine Translation: A Method to Reduce Meaning Loss. In Jill Burstein, Christy Doran, and Thamar Solorio (eds.), *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pp. 437–441, Minneapolis, Minnesota, June 2019. Association for Computational Linguistics. doi: 10.18653/v1/N19-1042. URL https://aclanthology.org/N19-1042/.

Reuben Cohn-Gordon, Noah Goodman, and Christopher Potts. Pragmatically Informative Image Captioning with Character-Level Inference. In Marilyn Walker, Heng Ji, and Amanda Stent (eds.), *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 2 (Short Papers)*, pp. 439–443, New Orleans, Louisiana, June 2018. Association for Computational Linguistics. doi: 10.18653/v1/N18-2070. URL https://aclanthology.org/N18-2070/.

I. Csiszar. I-Divergence Geometry of Probability Distributions and Minimization Problems. *The Annals of Probability*, 3(1):146–158, February 1975. ISSN 0091-1798, 2168-894X. doi: 10.1214/aop/1176996454. URL https://projecteuclid.org/journals/annals-of-probability/volume-3/issue-1/I-Divergence-Geometry-of-Probability-Distributions-and-Minimization-Problems/10.1214/aop/1176996454.full.

Marco Cuturi. Sinkhorn Distances: Lightspeed Computation of Optimal Transport. In *Advances in Neural Information Processing Systems*, volume 26. Curran Associates, Inc., 2013. URL https://proceedings.neurips.cc/paper_files/paper/2013/hash/af21d0c97db2e27e13572cbf59eb343d-Abstract.html.

Judith Degen. The Rational Speech Act Framework. *Annual Review of Linguistics*, 9 (Volume 9, 2023):519–540, January 2023. ISSN 2333-9683, 2333-9691. doi: 10.1146/annurev-linguistics-031220-010811. URL https://www.annualreviews.org/content/journals/10.1146/annurev-linguistics-031220-010811. Publisher: Annual Reviews.

W. Edwards Deming and Frederick F. Stephan. On a Least Squares Adjustment of a Sampled Frequency Table When the Expected Marginal Totals are Known. *The Annals of Mathematical Statistics*, 11(4):427–444, 1940. ISSN 0003-4851. URL https://www.jstor.org/stable/2235722.

Jan Drchal. Object Aligner: A Configurable JSON Schema Similarity Score for Graphs, Applied to LLM Prompt Optimization, July 2026. URL http://arxiv.org/abs/2607.01972. arXiv:2607.01972 [cs.CL].

Lautaro Estienne. Unsupervised Calibration through Prior Adaptation for Text Classification using Large Language Models. In Momchil Hardalov, Zara Kancheva, Boris Velichkov, Ivelina Nikolova-Koleva, and Milena Slavcheva (eds.), *Proceedings of the 8th Student Research Workshop associated with the International Conference Recent Advances in Natural Language Processing*, pp. 13–22, Varna, Bulgaria, September 2023. INCOMA Ltd., Shoumen, Bulgaria. URL https://aclanthology.org/2023.ranlp-stud.2/.

Yu Fei, Yifan Hou, Zeming Chen, and Antoine Bosselut. Mitigating Label Biases for In-context Learning. In Anna Rogers, Jordan Boyd-Graber, and Naoaki Okazaki (eds.), *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 14014–14031, Toronto, Canada, July 2023. Association for Computational Linguistics. doi: 10.18653/v1/2023.acl-long.783. URL https://aclanthology.org/2023.acl-long.783/.

Stephen E. Fienberg. An Iterative Procedure for Estimation in Contingency Tables. *The Annals of Mathematical Statistics*, 41(3):907–917, June 1970. ISSN 0003-4851, 2168-8990. doi: 10.1214/aoms/1177696968. URL https://projecteuclid.org/journals/annals-of-mathematical-statistics/volume-41/issue-3/An-Iterative-Procedure-for-Estimation-in-Contingency-Tables/10.1214/aoms/1177696968.full.

Michael C. Frank and Noah D. Goodman. Predicting Pragmatic Reasoning in Language Games. *Science*, 336(6084):998–998, May 2012. doi: 10.1126/science.1218633. URL https://www.science.org/doi/10.1126/science.1218633.

Tilmann Gneiting and Adrian E Raftery. Strictly Proper Scoring Rules, Prediction, and Estimation. *Journal of the American Statistical Association*, 102(477):359–378, March 2007. ISSN 0162-1459, 1537-274X. doi: 10.1198/016214506000001437. URL http://www.tandfonline.com/doi/abs/10.1198/016214506000001437.

Zhicheng Guo, Sijie Cheng, Hao Wang, Shihao Liang, Yujia Qin, Peng Li, Zhiyuan Liu, Maosong Sun, and Yang Liu. StableToolBench: Towards Stable Large-Scale Benchmarking on Tool Learning of Large Language Models. In Lun-Wei Ku, Andre Martins, and Vivek Srikumar (eds.), *Findings of the Association for Computational Linguistics: ACL 2024*, pp. 11143–11156, Bangkok, Thailand, August 2024. Association for Computational Linguistics. doi: 10.18653/v1/2024.findings-acl.664. URL https://aclanthology.org/2024.findings-acl.664/.

H. Paul Grice. Logic and Conversation. In *Syntax and Semantics*, volume Speech Acts, pp. 43–58. Academic Press, New York, NY, USA, peter cole and jerry morgan edition, 1975.

Zhixiong Han, Yaru Hao, Li Dong, Yutao Sun, and Furu Wei. Prototypical Calibration for Few-shot Learning of Language Models. September 2022. URL https://openreview.net/forum?id=nUsP9lFADUF.

C. T. Ireland and S. Kullback. Contingency Tables with Given Marginals. *Biometrika*, 55(1):179–188, 1968. ISSN 0006-3444. doi: 10.2307/2334462. URL https://www.jstor.org/stable/2334462.

Paul Knopp and Richard Sinkhorn. Concerning nonnegative matrices and doubly stochastic matrices. *Pacific Journal of Mathematics*, 21(2):343–348, January 1967. ISSN 0030-8730. URL https://projecteuclid.org/journals/pacific-journal-of-mathematics/volume-21/issue-2/Concerning-nonnegative-matrices-and-doubly-stochastic-matrices/pjm/1102992505.full.

Shirley Kokane, Ming Zhu, Tulika Manoj Awalgaonkar, Jianguo Zhang, Akshara Prabhakar, Thai Quoc Hoang, Zuxin Liu, Rithesh R. N, Liangwei Yang, Weiran Yao, Juntao Tan, Zhiwei Liu, Huan Wang, Juan Carlos Niebles, Shelby Heinecke, Caiming Xiong, and Silvio Savarese. ToolScan: A Benchmark For Characterizing Errors In Tool-Use LLMs. March 2025. URL https://openreview.net/forum?id=09tnQgqKuZ.

Minghao Li, Yingxiu Zhao, Bowen Yu, Feifan Song, Hangyu Li, Haiyang Yu, Zhoujun Li, Fei Huang, and Yongbin Li. API-Bank: A Comprehensive Benchmark for Tool-Augmented LLMs. In Houda Bouamor, Juan Pino, and Kalika Bali (eds.), *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pp. 3102–3116, Singapore, December 2023. Association for Computational Linguistics. doi: 10.18653/v1/2023.emnlp-main.187. URL https://aclanthology.org/2023.emnlp-main.187/.

Qiqiang Lin, Muning Wen, Qiuying Peng, Guanyu Nie, Junwei Liao, Jun Wang, Xiaoyun Mo, Jiamu Zhou, Cheng Cheng, Yin Zhao, Jun Wang, and Weinan Zhang. Robust Function-Calling for On-Device Language Model via Function Masking. October 2024. URL https://openreview.net/forum?id=yVQcr4qjD6.

Zachary Lipton, Yu-Xiang Wang, and Alexander Smola. Detecting and Correcting for Label Shift with Black Box Predictors. In *Proceedings of the 35th International Conference on Machine Learning*, pp. 3122–3130. PMLR, July 2018. URL https://proceedings.mlr.press/v80/lipton18a.html.

Weiwen Liu, Xu Huang, Xingshan Zeng, Xinlong Hao, Shuai Yu, Dexun Li, Shuai Wang, Weinan Gan, Zhengying Liu, Yuanqing Yu, Zezhong Wang, Yuxian Wang, Wu Ning, Yutai Hou, Bin Wang, Chuhan Wu, Wang Xinzhi, Yong Liu, Yasheng Wang, Duyu Tang, Dandan Tu, Lifeng Shang, Xin Jiang, Ruiming Tang, Defu Lian, Qun Liu, and Enhong Chen. ToolACE: Winning the Points of LLM Function Calling. October 2024a. URL https://openreview.net/forum?id=8EB8k6DdCU.

Zuxin Liu, Thai Quoc Hoang, Jianguo Zhang, Ming Zhu, Tian Lan, Shirley Kokane, Juntao Tan, Weiran Yao, Zhiwei Liu, Yihao Feng, Rithesh R. N, Liangwei Yang, Silvio Savarese, Juan Carlos Niebles, Huan Wang, Shelby Heinecke, and Caiming Xiong. APIGen: Automated PIpeline for Generating Verifiable and Diverse Function-Calling Datasets. November 2024b. URL https://openreview.net/forum?id=Jfg3vw2bjx#discussion.

Jiarui Lu, Thomas Holleis, Yizhe Zhang, Bernhard Aumayer, Feng Nan, Haoping Bai, Shuang Ma, Shen Ma, Mengyu Li, Guoli Yin, Zirui Wang, and Ruoming Pang. ToolSandbox: A Stateful, Conversational, Interactive Evaluation Benchmark for LLM Tool Use Capabilities. In Luis Chiruzzo, Alan Ritter, and Lu Wang (eds.), *Findings of the Association for Computational Linguistics: NAACL 2025*, pp. 1160–1183, Albuquerque, New Mexico, April 2025. Association for Computational Linguistics. ISBN 979-8-89176-195-7. doi: 10.18653/v1/2025.findings-naacl.65. URL https://aclanthology.org/2025.findings-naacl.65/.

Aman Madaan, Niket Tandon, Prakhar Gupta, Skyler Hallinan, Luyu Gao, Sarah Wiegreffe, Uri Alon, Nouha Dziri, Shrimai Prabhumoye, Yiming Yang, Shashank Gupta, Bodhisattwa Prasad Majumder, Katherine Hermann, Sean Welleck, Amir Yazdanbakhsh, and Peter Clark. Self-Refine: Iterative Refinement with Self-Feedback. November 2023. URL https://openreview.net/forum?id=S37hOerQLB.

Shishir G. Patil, Tianjun Zhang, Xin Wang, and Joseph E. Gonzalez. Gorilla: Large Language Model Connected with Massive APIs. In *Advances in Neural Information Processing Systems*, volume 37, pp. 126544–126565. Curran Associates, Inc., 2024. doi: 10.52202/079017-4020. URL https://proceedings.neurips.cc/paper_files/paper/2024/hash/e4c61f578ff07830f5c37378dd3ecb0d-Abstract-Conference.html.

Shishir G. Patil, Huanzhi Mao, Fanjia Yan, Charlie Cheng-Jie Ji, Vishnu Suresh, Ion Stoica, and Joseph E. Gonzalez. The Berkeley Function Calling Leaderboard (BFCL): From Tool Use to Agentic Evaluation of Large Language Models. June 2025. URL https://openreview.net/forum?id=2GmDdhBdDk.

Yujia Qin, Shihao Liang, Yining Ye, Kunlun Zhu, Lan Yan, Yaxi Lu, Yankai Lin, Xin Cong, Xiangru Tang, Bill Qian, Sihan Zhao, Lauren Hong, Runchu Tian, Ruobing Xie, Jie Zhou, Mark Gerstein, Dahai Li, Zhiyuan Liu, and Maosong Sun. ToolLLM: Facilitating Large Language Models to Master 16000+ Real-world APIs. October 2023. URL https://openreview.net/forum?id=dHng2O0Jjr.

Changle Qu, Sunhao Dai, Xiaochi Wei, Hengyi Cai, Shuaiqiang Wang, Dawei Yin, Jun Xu, and Jirong Wen. Tool learning with large language models: a survey. *Frontiers of Computer Science*, 19 (8):198343, January 2025. ISSN 2095-2236. doi: 10.1007/s11704-024-40678-2. URL https://doi.org/10.1007/s11704-024-40678-2.

Marco Saerens, Patrice Latinne, and Christine Decaestecker. Adjusting the outputs of a classifier to new a priori probabilities: a simple procedure. *Neural Computation*, 14(1):21–41, January 2002. ISSN 0899-7667. doi: 10.1162/089976602753284446.

Timo Schick, Jane Dwivedi-Yu, Roberto Dessi, Roberta Raileanu, Maria Lomeli, Eric Hambro, Luke Zettlemoyer, Nicola Cancedda, and Thomas Scialom. Toolformer: Language Models Can Teach Themselves to Use Tools. In *Advances in Neural Information Processing Systems*, volume 36, pp. 68539–68551. Curran Associates, Inc., 2023. doi: 10.52202/075280-2997. URL https://papers.nips.cc/paper_files/paper/2023/hash/d842425e4bf79ba039352da0f658a906-Abstract-Conference.html.

Sheng Shen, Daniel Fried, Jacob Andreas, and Dan Klein. Pragmatically Informative Text Generation. In Jill Burstein, Christy Doran, and Thamar Solorio (eds.), *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pp. 4060–4067, Minneapolis, Minnesota, June 2019. Association for Computational Linguistics. doi: 10.18653/v1/N19-1410. URL https://aclanthology.org/N19-1410/.

Noah Shinn, Federico Cassano, Ashwin Gopinath, Karthik Narasimhan, and Shunyu Yao. Reflexion: language agents with verbal reinforcement learning. In *Advances in Neural Information Processing Systems*, volume 36, pp. 8634–8652. Curran Associates, Inc., 2023. doi: 10.52202/075280-0377. URL https://proceedings.neurips.cc/paper_files/paper/2023/hash/1b44b878bb782e6954cd888628510e90-Abstract-Conference.html.

Harsh Trivedi, Tushar Khot, Mareike Hartmann, Ruskin Manku, Vinty Dong, Edward Li, Shashank Gupta, Ashish Sabharwal, and Niranjan Balasubramanian. AppWorld: A Controllable World of Apps and People for Benchmarking Interactive Coding Agents. In Lun-Wei Ku, Andre Martins, and Vivek Srikumar (eds.), *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 16022–16076, Bangkok, Thailand, August 2024. Association for Computational Linguistics. doi: 10.18653/v1/2024.acl-long.850. URL https://aclanthology.org/2024.acl-long.850/.

Wenxuan Wang, Shi Juluan, Zixuan Ling, Yuk-Kit Chan, Chaozheng Wang, Cheryl Lee, Youliang Yuan, Jen-tse Huang, Wenxiang Jiao, and Michael R. Lyu. Learning to Ask: When LLM Agents Meet Unclear Instruction. In Christos Christodoulopoulos, Tanmoy Chakraborty, Carolyn Rose, and Violet Peng (eds.), *Proceedings of the 2025 Conference on Empirical Methods in Natural Language Processing*, pp. 21773–21784, Suzhou, China, November 2025. Association for Computational Linguistics. ISBN 979-8-89176-332-6. doi: 10.18653/v1/2025.emnlp-main.1104. URL https://aclanthology.org/2025.emnlp-main.1104/.

Yifan Wang and Vera Demberg. RSA-Control: A Pragmatics-Grounded Lightweight Controllable Text Generation Framework. In Yaser Al-Onaizan, Mohit Bansal, and Yun-Nung Chen (eds.), *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing*, pp. 5561–5582, Miami, Florida, USA, November 2024. Association for Computational Linguistics. doi: 10.18653/v1/2024.emnlp-main.318. URL https://aclanthology.org/2024.emnlp-main.318/.

Mengsong Wu, Tong Zhu, Han Han, Chuanyuan Tan, Xiang Zhang, and Wenliang Chen. Seal-Tools: Self-instruct Tool Learning Dataset for Agent Tuning and Detailed Benchmark. In *Natural Language Processing and Chinese Computing: 13th National CCF Conference, NLPCC 2024, Hangzhou, China, November 1–3, 2024, Proceedings, Part II*, pp. 372–384, Berlin, Heidelberg, November 2024. Springer-Verlag. ISBN 978-981-97-9433-1. doi: 10.1007/978-981-97-9434-8_29. URL https://doi.org/10.1007/978-981-97-9434-8_29.

Shunyu Yao, Jeffrey Zhao, Dian Yu, Nan Du, Izhak Shafran, Karthik R. Narasimhan, and Yuan Cao. ReAct: Synergizing Reasoning and Acting in Language Models. September 2022. URL https://openreview.net/forum?id=WE_vluYUL-X.

Shunyu Yao, Noah Shinn, Pedram Razavi, and Karthik R. Narasimhan. {τ}-bench: A Benchmark for \underline{T}ool-\underline{A}gent-\underline{U}ser Interaction in Real-World Domains. October 2024. URL https://openreview.net/forum?id=roNSXZpUDN.

Noga Zaslavsky, Jennifer Hu, and Roger P. Levy. A Rate–Distortion view of human pragmatic reasoning? In Allyson Ettinger, Ellie Pavlick, and Brandon Prickett (eds.), *Proceedings of the Society for Computation in Linguistics 2021*, pp. 347–348, Online, February 2021. Association for Computational Linguistics. URL https://aclanthology.org/2021.scil-1.32/.

Kexun Zhang, Hongqiao Chen, Lei Li, and William Wang. Don't fine-tune, decode: Syntax error-free tool use via constrained decoding, 2024. URL https://arxiv.org/abs/2310.07075.

Xuan Zhang, Yongliang Shen, Zhe Zheng, Linjuan Wu, Wenqi Zhang, Yuchen Yan, Qiuying Peng, Jun Wang, and Weiming Lu. AskToAct: Enhancing LLMs Tool Use via Self-Correcting Clarification. In Christos Christodoulopoulos, Tanmoy Chakraborty, Carolyn Rose, and Violet Peng (eds.), *Proceedings of the 2025 Conference on Empirical Methods in Natural Language Processing*, pp. 13484–13511, Suzhou, China, November 2025. Association for Computational Linguistics. ISBN 979-8-89176-332-6. doi: 10.18653/v1/2025.emnlp-main.682. URL https://aclanthology.org/2025.emnlp-main.682/.

Zihao Zhao, Eric Wallace, Shi Feng, Dan Klein, and Sameer Singh. Calibrate Before Use: Improving Few-shot Performance of Language Models. In *Proceedings of the 38th International Conference on Machine Learning*, pp. 12697–12706. PMLR, July 2021. URL https://proceedings.mlr.press/v139/zhao21c.html.

Han Zhou, Xingchen Wan, Lev Proleev, Diana Mincu, Jilin Chen, Katherine A. Heller, and Subhrajit Roy. Batch Calibration: Rethinking Calibration for In-Context Learning and Prompt Engineering. October 2023. URL https://openreview.net/forum?id=L3FHMoKZcS.

A Information projection, RSA, and calibration

We first isolate the generic finite-dimensional projection underlying SPA. This avoids mixing the query-local notation of the main text with the standard IPFP and RSA notation.

Proposition 2 (KL projection and IPFP)**.** *Assume $S_0(u \mid \mathbf{m}) > 0$, $\mu(\mathbf{m}) > 0$, and $\nu(u) > 0$, with μ and ν probability distributions. Define*

$$\pi_0(\mathbf{m}, u) = \mu(\mathbf{m}) S_0(u \mid \mathbf{m})$$

and let $\Pi(\nu, \mu)$ be the set of joint distributions with marginals μ and ν. Then

$$\pi^\star = \arg \min_{\pi \in \Pi(\nu,\mu)} D_{\mathrm{KL}}(\pi \| \pi_0)$$

is unique and has the form

$$\pi^\star(\mathbf{m}, u) = \mu(\mathbf{m}) \frac{S_0(u \mid \mathbf{m}) \exp(\gamma^\star(u))}{\sum_{u' \in \mathcal{U}} S_0(u' \mid \mathbf{m}) \exp(\gamma^\star(u'))}, \tag{22}$$

where $\gamma^\star$ is unique modulo addition of a constant. Equivalently, after fixing the gauge $\mathbf{1}^\top\gamma = 0$, $\gamma^\star$ is the unique minimizer of

$$F(\gamma) = -\langle \nu, \gamma\rangle + \mathbb{E}_{\mathbf{m}\sim\mu}\left[\log \sum_{u\in\mathcal{U}} S_0(u \mid \mathbf{m}) \exp(\gamma(u))\right]. \tag{23}$$

Moreover, standard IPFP applied to π_0 converges to $\pi^\star$; in the potential parametrization its column-scaling update can be written

$$\gamma^{n+1}(u) = \log \nu(u) - \log \mathbb{E}_{\mathbf{m}\sim\mu}\left[\frac{S_0(u \mid \mathbf{m})}{\sum_{u'\in\mathcal{U}} S_0(u' \mid \mathbf{m}) \exp(\gamma^n(u'))}\right]. \tag{24}$$

Proof. Because the prescribed marginals are strictly positive, $\Pi(\nu, \mu)$ is nonempty, and strict positivity of π_0 makes the KL objective finite on its relative interior. Strict convexity of KL in the joint distribution gives uniqueness of the I-projection. Write every feasible joint with the fixed m-marginal as $\pi(\mathbf{m}, u) = \mu(\mathbf{m})Q(u \mid \mathbf{m})$. Then

$$D_{\mathrm{KL}}(\pi\|\pi_0) = \sum_{\mathbf{m}\in\mathcal{M}} \mu(\mathbf{m}) D_{\mathrm{KL}}(Q(\cdot \mid \mathbf{m})\|S_0(\cdot \mid \mathbf{m})),$$

subject to $\sum_{\mathbf{m}\in\mathcal{M}} \mu(\mathbf{m})Q(u \mid \mathbf{m}) = \nu(u)$. Lagrange multipliers for this response-marginal constraint yield

$$Q_\gamma(u \mid \mathbf{m}) = \frac{S_0(u \mid \mathbf{m}) \exp(\gamma(u))}{\sum_{u'\in\mathcal{U}} S_0(u' \mid \mathbf{m}) \exp(\gamma(u'))},$$

which gives equation 22. Substitution into the Lagrangian gives equation 23, whose gradient is

$$\nabla F(\gamma) = \mathbb{E}_{\mathbf{m}\sim\mu}[Q_\gamma(\cdot \mid \mathbf{m})] - \nu.$$

Its Hessian is an average of categorical covariance matrices. Each is positive definite on $\mathbf{1}^\perp$ because $Q_\gamma(\cdot \mid \mathbf{m})$ has full support; hence F is strictly convex on the centered gauge and the potential is unique there. The fixed-point condition $\nabla F(\gamma) = 0$ is equivalent to

$$\exp(\gamma(u))\mathbb{E}_{\mathbf{m}\sim\mu}\left[\frac{S_0(u \mid \mathbf{m})}{\sum_{u'\in\mathcal{U}} S_0(u' \mid \mathbf{m}) \exp(\gamma(u'))}\right] = \nu(u),$$

which gives equation 24. Convergence of alternating proportional fitting to the unique I-projection is the classical IPFP result (Ireland & Kullback, 1968; Fienberg, 1970; Csiszar, 1975). □

Proof of Proposition 1. Apply Proposition 2 with $m = \mathbf{m}$, $\mu(m) = \mu(\mathbf{m}; \mathbf{m}^\star)$, and $\nu(u) = \nu(u; \mathbf{m}^\star)$. The seed becomes exactly equation 3; equations 22 and 24 then become equations 7 and 8. The projected joint is unique, while the potential is invariant to addition of a common constant because that constant cancels in every row normalization. □

A.1 RSA AT UNIT PRAGMATIC SCALING

The exact connection to RSA is clearest at $\alpha = 1$. Let $P(\mathbf{m}) > 0$ be the RSA meaning prior and let the utterance cost satisfy $\sum_{u\in\mathcal{U}} \exp(-C(u)) = 1$. Starting from a literal speaker S_0, define recursively

$$L_n(\mathbf{m} \mid u) = \frac{S_{n-1}(u \mid \mathbf{m})P(\mathbf{m})}{\sum_{\mathbf{m}'\in\mathcal{M}} S_{n-1}(u \mid \mathbf{m}')P(\mathbf{m}')}, \tag{25}$$

$$S_n(u \mid \mathbf{m}) = \frac{L_n(\mathbf{m} \mid u) \exp(-C(u))}{\sum_{u'\in\mathcal{U}} L_n(\mathbf{m} \mid u') \exp(-C(u'))}. \tag{26}$$

Proposition 3 (RSA–IPFP equivalence at $\alpha = 1$)**.** *For every $n \geq 0$ there exists a response potential γ^n such that*

$$S_n(u \mid m) = \frac{S_0(u \mid \mathbf{m}) \exp(\gamma^n(u))}{\sum_{u'\in\mathcal{U}} S_0(u' \mid \mathbf{m}) \exp(\gamma^n(u'))}.$$

With $\gamma^0 = 0$, the potential obeys

$$\gamma^{n+1}(u) = -C(u) - \log \mathbb{E}_{\mathbf{m}\sim P}\left[\frac{S_0(u \mid \mathbf{m})}{\sum_{u'\in\mathcal{U}} S_0(u' \mid \mathbf{m}) \exp(\gamma^n(u'))}\right]. \tag{27}$$

Consequently, if $\mu = P$ and $\nu(u) = \exp(-C(u))$, the RSA recursion is exactly the IPFP recursion equation 24, up to the irrelevant additive gauge of γ.

Proof. Assume the displayed exponential form holds at level n. In the listener, the row-normalizing factor of $S_n(\cdot \mid \mathbf{m})$ depends only on m and therefore cancels from the speaker normalization after the listener is substituted into S_{n+1}. Collecting the remaining factors that depend only on u gives equation 27. The base case is $\gamma^0 = 0$, so induction proves the representation. Substituting $P = \mu$ and $e^{-C} = \nu$ gives equation 24. □

Remark on $\alpha \neq 1$. For general RSA rationality α, repeated recursion also raises the literal factor to $S_0(u \mid m)^{\alpha^n}$. Hence the fixed-seed KL projection used by SPA is not, in general, the same iteration. The exact equivalence claimed in this paper is therefore specifically the unit-scaling case $\alpha = 1$.

A.2 Equivalent cross-entropy characterization

Proposition 4 (Prior-shift calibration)**.** *Under the assumptions of Proposition 2, let $S_r \in \Pi(\nu, \mu)$ be any reference joint and define*

$$S_\gamma(u \mid \mathbf{m}) = \frac{S_0(u \mid \mathbf{m}) \exp(\gamma(u))}{\sum_{u'\in\mathcal{U}} S_0(u' \mid \mathbf{m}) \exp(\gamma(u'))}.$$

Then

$$\mathcal{C}_r(\gamma) = -\sum_{\mathbf{m}\in\mathcal{M}} \sum_{\mathbf{u}\in\mathcal{U}} S_r(\mathbf{m}, u) \log S_\gamma(u \mid \mathbf{m})$$

is convex and depends on S_r only through (μ, ν), up to an additive constant. Its minimizers are exactly $\gamma^\star + \mathrm{span}\{\mathbf{1}\}$, where $\gamma^\star$ is the potential in Proposition 2.

Proof. Expanding the logarithm gives

$$\mathcal{C}_r(\gamma) = K - \langle \nu, \gamma \rangle + \sum_{\mathbf{m}\in\mathcal{M}} \mu(\mathbf{m}) \log \sum_{u\in\mathcal{U}} S_0(u \mid \mathbf{m}) \exp(\gamma(u)),$$

where $K = -\sum_{\mathbf{m},u} S_r(\mathbf{m}, u) \log S_0(u \mid \mathbf{m})$ is independent of γ. Thus, apart from K, this is exactly the dual objective equation 23. Its gradient and Hessian are

$$\nabla \mathcal{C}_r(\gamma) = \sum_{\mathbf{m}\in\mathcal{M}} \mu(\mathbf{m}) S_\gamma(\cdot \mid \mathbf{m}) - \nu, \tag{28}$$

$$\nabla^2 \mathcal{C}_r(\gamma) = \sum_{\mathbf{m}\in\mathcal{M}} \mu(\mathbf{m}) \left[\mathrm{Diag}(S_\gamma(\cdot \mid \mathbf{m})) - S_\gamma(\cdot \mid m) S_\gamma(\cdot \mid \mathbf{m})^\top\right] \succeq 0. \tag{29}$$

Proposition 2 therefore gives the stated minimizer set. □

B Population and sampled-neighborhood theory

For a fixed target $\mathbf{m}^\star$, write $\mu = \mu(\cdot; \mathbf{m}^\star)$ and $\nu = \nu(\cdot; \mathbf{m}^\star)$. On the centered gauge $\mathbf{1}^\top \gamma = 0$, define

$$F(\gamma) = -\langle \nu, \gamma \rangle + \mathbb{E}_{\mathbf{m}\sim\mu}\left[\log \sum_{u\in\mathcal{U}} S_0(u \mid \mathbf{m}) \exp(\gamma(u))\right], \tag{30}$$

$$F_N(\gamma) = -\langle \nu, \gamma \rangle + \frac{1}{N} \sum_{i=1}^{N} \log \sum_{u\in\mathcal{U}} S_0(u \mid \mathbf{m}_i) \exp(\gamma(u)), \tag{31}$$

where $\mathbf{m}_1, \ldots, \mathbf{m}_N \overset{\text{iid}}{\sim} \mu$.

Proposition 5 (Consistency of the sampled potential)**.** *Assume $\mathcal{U}$ is finite, $\nu(u) > 0$ for every u, and there exists $\varepsilon > 0$ such that $S_0(u \mid \mathbf{m}) \geq \varepsilon$ almost surely for every u. Then F has a unique centered minimizer $\gamma^\star$, and any centered minimizer $\gamma_N^\star$ of F_N satisfies*

$$\gamma_N^\star \xrightarrow{\text{a.s.}} \gamma^\star.$$

Consequently,

$$S_N^\star(\cdot \mid \mathbf{m}^\star) \xrightarrow{\text{a.s.}} S^\star(\cdot \mid \mathbf{m}^\star).$$

Proof. For fixed γ, the strong law gives $F_N(\gamma) \to F(\gamma)$ almost surely. On every compact subset of $\mathbf{1}^\perp$, the log-partition integrand is uniformly Lipschitz in γ, so a finite-net argument gives uniform convergence on compacts. The Hessian of F is

$$\mathbb{E}_{\mathbf{m}\sim\mu}\left[\mathrm{Diag}(r_{\mathbf{m},\gamma}) - r_{\mathbf{m},\gamma} r_{\mathbf{m},\gamma}^\top\right], \qquad r_{\mathbf{m},\gamma}(u) = \frac{S_0(u \mid \mathbf{m}) \exp(\gamma(u))}{\sum_{u' \in \mathcal{U}} S_0(u' \mid \mathbf{m}) \exp(\gamma(u'))}.$$

Every tilted row has full support, so this Hessian is positive definite on $\mathbf{1}^\perp$. Thus F is strictly convex on the centered gauge. Uniform positivity of S_0 and strict positivity of ν imply coercivity of both F and F_N on $\mathbf{1}^\perp$: the log-partition grows as $\max_u \gamma_u$ up to a uniform additive constant, whereas $\langle \nu, \gamma \rangle$ is a strict convex combination of the coordinates. Hence the minimizers are confined to a common compact set eventually. Uniform convergence on that set and uniqueness of the population minimizer give almost-sure convergence of the argmin. The final statement follows by continuity of exponential reweighting. □

The preceding result justifies the sampled neighborhood as a Monte Carlo approximation of the population projection. The next result describes its first-order statistical fluctuations and, importantly, clarifies what the curvature of the projection does and does not measure.

Let $B \in \mathbb{R}^{|\mathcal{U}| \times d}$, $d = |\mathcal{U}| - 1$, have orthonormal columns spanning $\mathbf{1}^\perp$. Define at the population optimum

$$r_{\mathbf{m}} := r_{\mathbf{m},\gamma^\star}, \tag{32}$$

$$H := B^\top \mathbb{E}\big[\mathrm{Diag}(r_{\mathbf{m}}) - r_{\mathbf{M}} r_{\mathbf{m}}^\top\big]\, B, \tag{33}$$

$$\Sigma := B^\top \,\mathrm{Cov}(r_{\mathbf{m}}) B. \tag{34}$$

The first-order condition gives $\mathbb{E}_{\mathbf{m}\sim\mu}[r_{\mathbf{m}}] = \nu$.

Theorem 1 (Asymptotic distribution of the sampled potential)**.** *Under the assumptions of Proposition 5,*

$$\sqrt{N}\, B^\top(\gamma_N^\star - \gamma^\star) \xrightarrow{d} \mathcal{N}\big(0, H^{-1}\Sigma H^{-1}\big). \tag{35}$$

In particular, the local spectral quantity

$$\lambda_{\mathrm{SPA}} := \lambda_{\min}(H) > 0$$

controls the sensitivity of the potential estimator, but the sampling variability is governed jointly by H and the between-meaning covariance Σ.

Proof. In centered coordinates $\eta = B^\top \gamma$, the empirical first-order condition is $\nabla F_N(\eta_N^\star) = 0$. A Taylor expansion around $\eta^\star$ gives

$$0 = \nabla F_N(\eta^\star) + \nabla^2 F_N(\widetilde{\eta}_N)(\eta_N^\star - \eta^\star)$$

for a point $\widetilde{\eta}_N$ on the segment between the two minimizers. By Proposition 5, $\widetilde{\eta}_N \to \eta^\star$ almost surely, and the law of large numbers gives $\nabla^2 F_N(\widetilde{\eta}_N) \to H$ almost surely. Moreover,

$$\sqrt{N}\,\nabla F_N(\eta^\star) = \frac{1}{\sqrt{N}} \sum_{i=1}^N B^\top (r_{\mathbf{m}_i} - \nu) \xrightarrow{d} \mathcal{N}(0, \Sigma)$$

by the multivariate central limit theorem. Since H is positive definite, Slutsky's theorem yields equation 35. □

Name	Tool-call mode	Tool presentation in the system turn	Expected output format
Qwen3-0.6B-FC	Native function calling	Qwen3's native `# Tools` block: one JSON signature per tool inside `<tools></tools>` tags	One JSON object `{"name": ..., "arguments": {...}}` per call, each inside `<tool_call></tool_call>` tags (Hermes format)
Qwen3-0.6B-Prompt	Prompting	A plain-prose instruction followed by the tool list as indented JSON	A Python-style list of calls, e.g. `[f(a=1, b="x"), g(c=2)]`

Table 2: Tool-call modes and prompt formats.

Corollary 1 (Asymptotic distribution of the adapted target speaker)**.** *Let* $s^\star := S^\star(\cdot \mid \mathbf{m}^\star)$ *and* $s_N^\star := S_N^\star(\cdot \mid \mathbf{m}^\star)$*, viewed as vectors in* $\mathbb{R}^{|\mathcal{U}|}$*, and define*

$$J^\star := \left[\mathrm{Diag}(s^\star) - s^\star (s^\star)^\top\right] B.$$

Under the assumptions of Theorem 1,

$$\sqrt{N}\,(s_N^\star - s^\star) \xrightarrow{d} \mathcal{N}\big(0, J^\star H^{-1} \Sigma H^{-1} (J^\star)^\top\big)\,. \tag{36}$$

Thus the same $N^{-1/2}$ *Monte Carlo scaling propagates directly to the adapted response probabilities used for prediction.*

Proof. On the centered gauge write $\gamma = B\eta$. The map from η to the adapted target distribution is

$$g(\eta)_u = \frac{S_0(u \mid \mathbf{m}^\star) e^{(B\eta)_u}}{\sum_{u'} S_0(u' \mid \mathbf{m}^\star) e^{(B\eta)_{u'}}}.$$

Its Jacobian at $\eta^\star$ is $J^\star$. Applying the multivariate delta method to Theorem 1 gives equation 36. □

Proposition 6 (Within–between covariance decomposition)**.** *At the population optimum,*

$$B^\top \left[\mathrm{Diag}(\nu) - \nu\nu^\top\right] B = H + \Sigma. \tag{37}$$

Consequently,

$$0 \preceq \Sigma \preceq B^\top \left[\mathrm{Diag}(\nu) - \nu\nu^\top\right] B, \qquad H = B^\top \left[\mathrm{Diag}(\nu) - \nu\nu^\top\right] B - \Sigma.$$

Thus H *should not be interpreted as a measure of semantic diversity of the sampled meanings. It is the average* within-meaning *categorical curvature, whereas* Σ *measures how much the tilted response distribution varies* between meanings*.*

Proof. Using $\mathbb{E}_{\mathbf{m}\sim\mu}[r_{\mathbf{m}}] = \nu$,

$$\begin{aligned}\mathbb{E}[\mathrm{Diag}(r_{\mathbf{m}}) - r_{\mathbf{m}} r_{\mathbf{m}}^\top] + \mathrm{Cov}(r_{\mathbf{m}}) &= \mathrm{Diag}(\mathbb{E}[r_{\mathbf{m}}]) - \mathbb{E}[r_{\mathbf{m}} r_{\mathbf{m}}^\top] + \mathbb{E}[r_{\mathbf{m}} r_{\mathbf{m}}^\top] - \nu\nu^\top \\ &= \mathrm{Diag}(\nu) - \nu\nu^\top.\end{aligned}$$

Projecting onto $\mathbf{1}^\perp$ gives equation 37. □

Implication for sampler design. Equation 37 also clarifies the role of the sampler. Maximizing $\lambda_{\min}(H)$ alone does *not* select meanings that are maximally diverse in their response profiles: for fixed ν, increasing the between-meaning covariance Σ necessarily decreases the average within-meaning curvature H in the Loewner order. For estimation of the population potential, the relevant first-order uncertainty is instead the sandwich covariance $H^{-1}\Sigma H^{-1}$ from Theorem 1. This remains distinct from downstream pragmatic usefulness, which additionally depends on whether the sampled meanings provide contrasts that are informative for the target query. We therefore do not use a spectral-gap objective as a proxy for semantic or pragmatic diversity.

C THE STATI DATASET

Base models and prompts Table 2 shows the two configurations of the `Qwen3-0.6B` model used to run the experiments. The prompt mode uses the instructions used by vLLM's pythonic tool-calling templates[3]. In both modes, earlier assistant calls in the history are rendered in the same format the model is asked to produce, and tool results are wrapped in `<tool_response>` tags inside a user turn. No additional system prompt or few-shot examples are added beyond what the sample itself contains. Reasoning is left enabled when the final response is generated. Any `<think>...</think>` span the model produces is removed before the response is parsed. Each response is parsed with the parser that matches its mode: a Hermes `<tool_call>` JSON parser for FC, and a Python-call-list parser for Prompt. A response that cannot be parsed into a list of calls counts as a failed generation.

To generate the meanings we used a one-turn conversation consisting in a system prompt followed by a user asking to generate alternative queries for given the provided conversation. The used meta-prompt was the following:

```
<System>: You are an expert in replacing the last message of a
↪  conversation with a new, realistic and different message. You will be
↪  provided with a conversation in the form of a list of messages, where
↪  each message is a dictionary containing the role (system, user,
↪  assistant or tool) and the content of the message. The last message
↪  in the conversation contains the following content: `<last message>`.
↪  Your task is to generate an alternative content different that the
↪  one provided, that could plausibly replace the content of the last
↪  message in the conversation. The conversation may contain one or more
↪  messages, and the last message is the one that needs to be replaced.
↪  Note that if the conversation contains one turn, the full
↪  conversation is the last above message. In that case, just replace
↪  the content of the last message with a new one following the same
↪  guidelines. The output should be in the form of a dictionary of the
↪  form {'role': ..., 'content': ...}, where the role is the same as the
↪  last message in the conversation and the content is the generated
↪  alternative message.

<User>:
Conversation:
`<conversation>`.
Please generate an alternative content for the last message: `{'role':
↪  'user', 'content': '<last message>'}`. Remember that the output
↪  should be in the form of a dictionary of the form {'role': ...,
↪  'content': ...}, where the role is the same as the last message in
↪  the conversation and the content is the generated alternative
↪  message. The new message should be a direct replacement, not an
↪  addition to the conversation. It should be different from the
↪  original message, while still sounding natural and realistic.
```

Error types Figure 3 shows failure rate for all the tasks in Table 1. To count the different types of errors, we first compute the number and types of edits that have to be made on the parsed model output to obtain one of the valid expected calls. This was computed with the ObjectAligner library (Drchal, 2026) using the JSON-Schema in the documentation. We report as total number of edits for a specific sample, the minimum number across all valid calls for that sample (Graph Edit Distance, GED). Once computed this, we classify the error type using the set of collected edits on that sample. The types are *missing call*, *extra call*, *wrong function* (paired call with the wrong name), *missing arg* (missing required argument), *extra arg* (argument not in the gold call), *wrong arg name*, *wrong type*, *wrong value* and *wrong order*. In subtasks whose calls have a defined order, each paired call outside the longest in-order run costs one wrong order edit. Calls in the parallel subtasks (PA, PM, L-PA, L-PM) are compared without regard to order.

[3] https://docs.vllm.ai/en/stable/features/tool_calling/

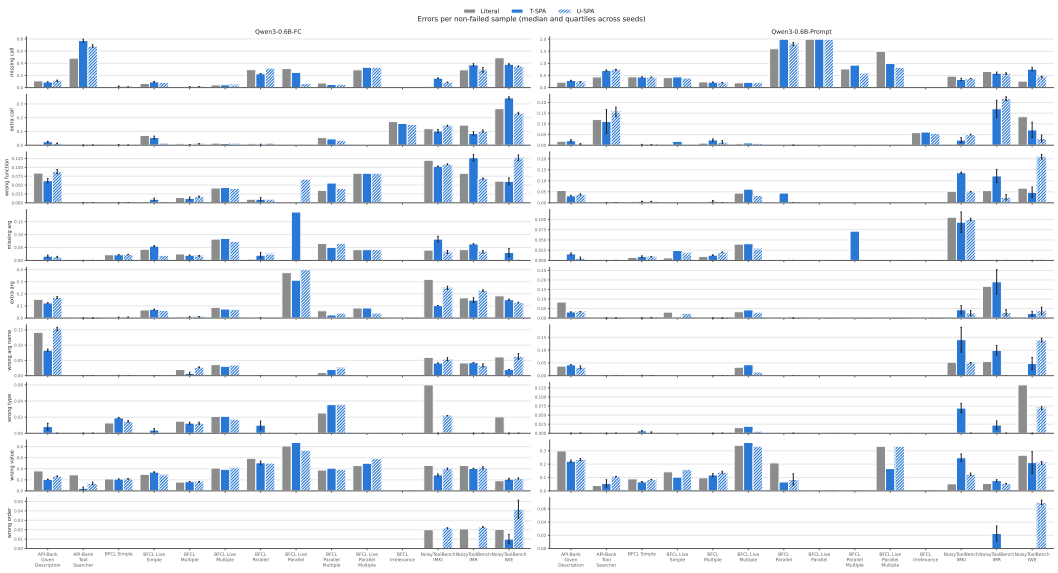

Figure 3: Error types for each model and method.

Statistics of the StaTI dataset Table 3 shows a summary of the collected tasks. The full dataset and the code used to build it are available at the code HuggingFace dataset repository.

Source	Subtask (abbr.)	# samples	What it tests
BFCL v4	Simple (SI)	550	One function, one call
	Multiple (MU)	200	One call, chosen among several functions
	Parallel (PA)	200	Several calls to the same function
	Parallel Multiple (PM)	199	Several calls to several functions
	Irrelevance (IR)	240	No function fits the request, so no call is expected
	Live Simple / Multiple / Parallel / Parallel Multiple (L-SI, L-MU, L-PA, L-PM)	258 / 1053 / 16 / 24	The same four settings, with user-contributed queries and functions
	Live Irrelevance (L-IR)	884	Irrelevance, with user-contributed queries and functions
	Multi-Turn Base (M-BA)	733	Next turn's calls in a stateful multi-turn conversation
	Multi-Turn Long Context (M-LC)	733	The same, with long tool outputs in the history
	Multi-Turn Missing Function (M-MF)	932	The same, with a needed function missing until a later turn
	Multi-Turn Missing Parameter (M-MP)	933	The same, with a required value missing until a later turn
API-Bank	Given Description (GD)	389	Next API call in a dialogue, with all APIs documented upfront
	Tool Searcher (TS)	106	The same, with only a tool-retrieval API given upfront
NoisyToolBench	IWE	50	Calls after the user corrects an error in the instruction
	IMKI	50	Calls after the user supplies missing key information
	IMR	48	Calls after the user resolves an ambiguous referent
	IBTC	50	The request is beyond the tools' capability, so no call is expected

Table 3: Benchmark subtasks used in the evaluation.