Clarify, Abstain or Answer?
Strategising in Conversation with Belief-Augmented Generation

Joris Baan, **Wilker Aziz**, **Barbara Plank**, **Raquel Fernández**
University of Amsterdam, MCML Munich, LMU Munich,
European Commission, Joint Research Centre (JRC), Brussels
{j.s.baan,w.aziz,raquel.fernandez}@uva.nl, b.plank@lmu.de

Abstract

Large language models (LLMs) define a distribution over text, which can be viewed as a probabilistic representation of uncertainty: sampling K responses yields a *belief state*—responses a model deems plausible. Existing work exploits this representation for narrow tasks like either decoding or selective prediction, and often requires manual interventions, not controlling generation directly. We propose Belief-Augmented Generation (BAG): grounding LLMs in their own belief state via the prompt and letting them reason over these K samples to decide on and execute a conversational strategy: clarify, abstain, or answer. In a multi-turn ambiguous question answering (QA) setting, we find that LLMs by default rarely clarify or abstain, ignoring uncertainty about the input (aleatoric) or facts (epistemic). BAG improves QA accuracy across six models and yields strategy decisions more faithful to their belief state than prompt-only baselines. Disentangling when to clarify from when to abstain, however, remains challenging.[1]

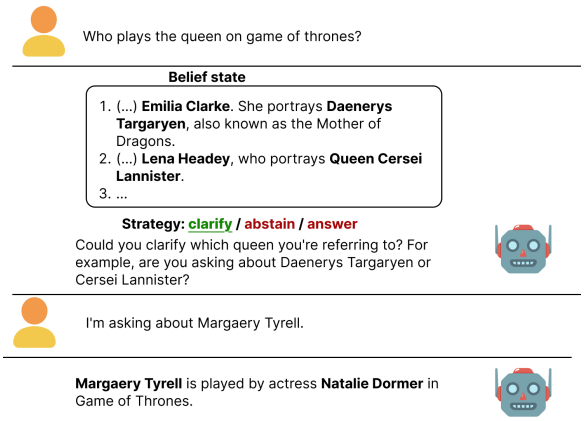


Figure 1: **Turn 1**: User asks a potentially ambiguous question. **Turn 2**: BAG samples K responses, analyses them, and formulates a strategy and response. **Turn 3**: BAG asked a clarification question so we simulate a user answer. **Turn 4**: Final answer, optionally with another round of BAG. Real Qwen3-14B example.

1 Introduction

Large language models (LLMs) are statistical models that autoregressively define a distribution over text. This can be viewed as a probabilistic representation of uncertainty or belief (Baan et al., 2023). By drawing multiple samples we obtain a textual representation, *i.e., belief state*: responses a model deems plausible under a given prompt.

Such representations are routinely and effectively used in various research subfields: in decoding, by selecting the highest expected utility or majority generation (Eikema and Aziz, 2020; Wang et al., 2023); in selective prediction, abstaining when entropy across semantic clusters of generations is high (Kuhn et al., 2023b); in clarification finetuning, to synthesise clarification questions (Zhang et al., 2025); and in verbalised uncertainty, to communicate uncertainty reflective of the relative frequency of a claim in the belief state (Yona et al., 2024; Eikema et al., 2025).

Our main contribution is to unify these disparate research strands, all using the same set of samples for a single narrow decision-making task, and do so without manual heuristics and interventions (*e.g.,* decoders, uncertainty quantifiers, selective prediction heuristics, or finetuning).

We capitalise on the intuition that inspecting multiple generations provides information about reliability. For example, sample diversity can be characteristic of different sources of uncertainty and, as such, informative of when and how to clarify, abstain, or answer. Specifically, we let LLMs reason over an explicit textual representation of their own uncertainty, obtained via sampling, guiding them to clarify, abstain or answer directly. This harnesses LLMs' strong abilities to reason about avail-

[1]Code available at github.com/jsbaan/belief-augmented-generation, and an explorer to browse questions, generations and evaluations at jorisbaan.nl/belief-augmented-generation.

Proceedings of the 19th International Natural Language Generation Conference: Long Papers, pages 842–864
October 17-21, 2026 ©2026 Association for Computational Linguistics

able textual evidence, as demonstrated in retrieval-augmented generation or RAG (Lewis et al., 2020).

We call our method *Belief-Augmented Generation* or BAG: conditioning generation directly on the belief state (rather than external documents in RAG) via the prompt. BAG lets an LLM decide when to answer, what strategy to follow, and how to formulate the answer all at once. BAG can be viewed as test-time compute scaling (Snell et al., 2025), requiring no training, no external data, and being applicable to any model, including closed models. Figure 1 shows an example.

The six LLMs in our experiments by default rarely clarify or abstain and ignore uncertainty in the input (sometimes called aleatoric or data uncertainty, for example due to ambiguity) and about facts (sometimes called epistemic or model uncertainty). In contrast, belief-augmented generation improves QA performance and leads to strategies more faithful to probabilistic uncertainty.

To complement automatic QA evaluation of increasingly complex and verbose LLM generations—which is non-trivial (Kamalloo et al., 2023; Adlakha et al., 2024)—we also release BAG Explorer 🖵, an online tool to manually browse the questions, references, generations and evaluations used in this paper.

2 Related Work

Decoding. One of the earlier works that exploit a textual representation of probabilistic belief (*i.e.,* K sampled generations) is Minimum Bayes Risk decoding (MBR), which selects the generation that best represents the entire set, demonstrating improved response quality (Eikema and Aziz, 2020; Suzgun et al., 2023; Wu et al., 2025). Similarly, self-consistency samples multiple reasoning chains and pick the majority answer (Wang et al., 2023; Bertsch et al., 2023). Both can be viewed as test-time compute scaling (Snell et al., 2025).

Selective prediction. Kuhn et al. (2023b); Farquhar et al. (2024) instead capitalise on the idea that semantically distinct claims in the belief state may be indicative of error (*i.e.,* model or epistemic uncertainty)—an intuition widely used to detect hallucinations (Geng et al., 2024; Vashurin et al., 2025). Others view variation in belief states as uncertainty about the input (*i.e.,* data or aleatoric uncertainty; Giulianelli et al., 2023; Baan et al., 2024), and there have been efforts to disentangle these sources (Cole et al., 2023; Hou et al., 2024). However, both decoding and selective prediction methods rely on hand-crafted features and their end goal is to predict error or select a generation rather than directly controlling generation.

Faithful uncertainty. Another line of research guides models to communicate their uncertainty faithfully, *i.e.,* hedge or strengthen a claim such that it reflects the relative frequency of that claim in the belief state (Yona et al., 2024; Xu et al., 2024; Eikema et al., 2025). Most similar to our work is Kirchhof et al. (2025), who propose to communicate the full internal distribution over possible answers rather than hedging a single answer. They find that directly conditioning on K samples in the prompt is most effective. We go beyond a simple summary and instead instruct models to analyse their own belief state to power dynamic conversational decisions. Also similar is Nan et al. (2025) who condition a bigger, auxiliary model on another model's belief state to analyse the source of its uncertainty. We instead let the same model reason about its own uncertainty to directly inform its generation process.

Conversational strategies. Predicting conversational strategies has a long history in (spoken) dialogue systems, from decision-theoretic action selection (Paek and Horvitz, 2000; Horvitz and Paek, 2000) to belief representation and estimation (Bohus and Rudnicky, 2005a,b; Williams et al., 2013).

However, LLMs use a limited range of conversational strategies and often answer directly, failing to recognize ambiguity or ask clarification questions, instead implicitly assuming an interpretation (Deng et al., 2023; Shaikh et al., 2024; Zhang et al., 2024; Testoni et al., 2025). While there have been attempts to remedy this via prompting (Kuhn et al., 2023a; Zhang et al., 2024) and fine-tuning (Andukuri et al., 2024; Chen et al., 2025), the belief state has been under-explored.

Testoni and Fernández (2024) suggest that human clarification questions as supervision for deciding when and how to clarify may not be ideal to resolve model uncertainty, which is echoed by Kim et al. (2024) who finetune on questions 'perceived' by a model as ambiguous, and Zhang et al. (2025) who finetune on synthetic clarifications generated by an external model instructed to distinguish between K sampled generations. Zhang and Choi (2025) predict ambiguity via entropy over multiple simulated user responses without exploiting them for how to clarify, and do not allow abstaining.

Crucially, unlike classical selective prediction, BAG guides models to abstain with a direct response to the user based on the belief state (see the example in Figure 2). In fact, the abstain strategy can easily be replaced by or enhanced with faithfully communicating uncertainty, or fetching external information, *i.e.,* deciding when to RAG. Similarly, clarification is but one strategy to address ambiguity. There are others, like long form answers (Berant et al., 2026; Saparina and Lapata, 2026), which are fully compatible with BAG. We are not aware of recent work combining direct answer, clarification and abstention strategies.

3 Belief-Augmented Generation (BAG)

BAG feeds a textual representation of probabilistic belief directly into the prompt, guiding models to improve their responses by making belief-informed conversational decisions. We capitalise on the intuition that inspecting multiple generations gives insight into the response reliability, *e.g.,* that sample diversity can be characteristic of different sources of uncertainty and, as such, informative of when and how to clarify, abstain, or answer.

3.1 Constructing the Belief State

An LLM with parameters θ defines a distribution $P_\theta(Y|X = x)$ over responses $Y \in \mathcal{X}$ given input $X = x \in \mathcal{X}$, where $\mathcal{X}$ is the space of strings. We take the stance that, for any input $X = x$, the probability measure P_θ under a given sampling algorithm is a representation of the LLM's uncertainty about the response Y, specifying the space of responses plausible under the model (Baan et al., 2023).

To summarise the predictive distribution for a given input $X = x$ one can draw multiple samples using Monte-Carlo (MC) simulation. Practitioners often reduce this set of strings to a single number to power a decision making tool, like what generation to pick (Eikema and Aziz, 2020; Wu et al., 2025) or when to abstain (Farquhar et al., 2024; Vashurin et al., 2025).

With BAG we retain the full set of K samples as the *belief state* $B_\theta(x) := (Y^{(1)}, \ldots, Y^{(K)})$, where $Y^{(k)} \sim P_\theta$. Interestingly, this representation not only captures diversity across generations that a model deems plausible, but also linguistic cues within generations, like hedging, refusals, clarifications, knowledge cut-offs, or nuanced acknowledgments of ambiguity.

3.2 Prompting for Conversational Strategies

BAG is a two-stage approach. First a model is shown a question to obtain multiple generations. Then these generations are included in a new prompt to help decide on and execute a conversational strategy. This prompt includes the user question x and belief state B and defines the allowed strategies, output format and how to analyse the belief state to arrive at a strategy. Appendix Figure 6d shows an example prompt.

A powerful property of BAG is the ability to capture multiple intuitions behind popular uncertainty quantification methods in plain text, directly in the prompt. We explore four intuitions: (1) inconsistent facts among generations point to hallucinations and indicate a form of epistemic uncertainty that should lead to abstaining (akin to semantic entropy; Kuhn et al., 2023b), (2) generations that address different interpretations of the question should lead to a clarification question, (3) generations that are largely consistent should lead to a direct answer, and (4), direct answers should resemble the entire belief state (akin to MBR; Eikema and Aziz, 2020). We discuss the exact prompts in Section 4.2.

3.3 BAG+

If there are multiple conversational turns, like when the assistant model asks a clarification question, one could apply BAG to every assistant turn, repeatedly augmenting the generation process—sampling K new generations given the conversation history with another range of available strategies. We call this version BAG+. BAG+ is particularly useful when, even after a clarification interaction with the user, the assistant model is (too) uncertain to provide an answer. Appendix Figure 12 shows an overview of the entire pipeline.

4 Experimental Setup

4.1 Dataset

We evaluate on real-world Google search questions from AmbigQA (Min et al., 2020), for four reasons. First, it is the standard in ambiguity and clarification research (e.g., Gao et al., 2021; Min et al., 2021; Cole et al., 2023; Kim et al., 2024; Zhang et al., 2025; Zhang and Choi, 2025; Berant et al., 2026). Second, many popular alternatives (*e.g.,* CLAMBER, ASQA, CAmbigNQ, CondAmbigQA; Zhang et al., 2024; Stelmakh et al., 2022; Lee et al., 2023; Gao et al., 2023; Li et al., 2025) are derived from AmbigQA or its source Natural Questions

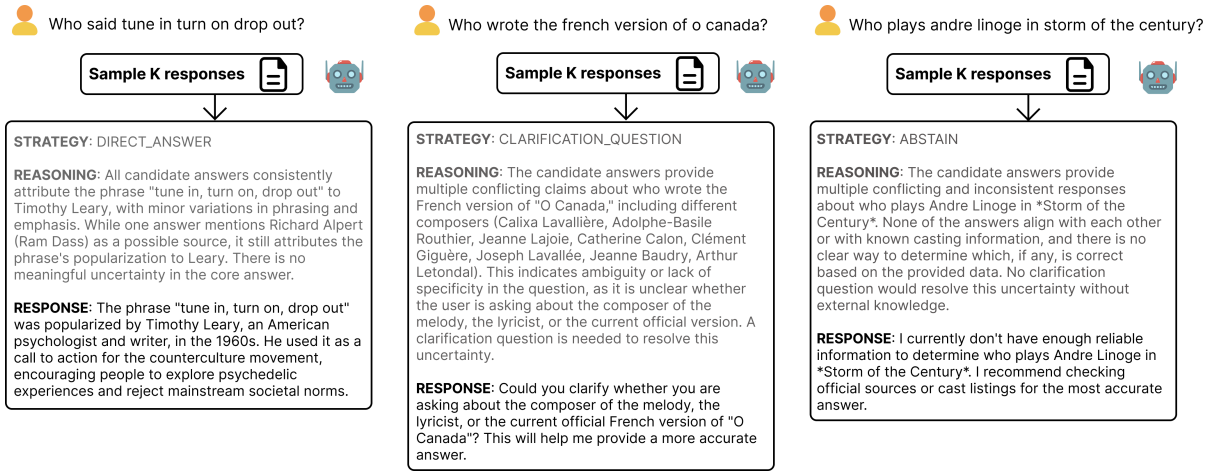


Figure 2: Three real examples of Qwen3-14B output when strategising about the best conversational strategy based on the user's question and its own belief state (which we omit for brevity) using BAG2. User only sees the **response**.

(Kwiatkowski et al., 2019), for instance, SituatedQA (Zhang and Choi, 2021), inheriting the same problems while adding little independent evidence. Third, AmbigQA contains uncertainty arising from the input due to ambiguity and underspecification, but also the model, due to the closed-book QA task that forces models to retrieve facts from their parameters. A dataset exhibiting only one does not allow evaluation of both clarify and abstain. Fourth, AmbigQA supplies unique annotations that we exploit as **user intents** to evaluate BAG rigorously and interactively (Section 5).

User intents. AmbigQA annotates each question with all possible answers and provides multiple disambiguated questions for them. For instance, *"Who is the female singer on Gimme Shelter?"* yields two disambiguated pairs: *Who was the female singer on the recorded version of Gimme Shelter?* → Merry Clayton and *Who was the female singer on Gimme Shelter on tour?* → Lisa Fischer. Following recent work, we treat each disambiguation-answer pair as a separate *user intent* (Zhang et al., 2025; Zhang and Choi, 2025; Berant et al., 2026). We evaluate on the full development set (the test set is not public), filtering out 170 questions with conflicting annotations, resulting in 1,832 questions. Roughly half (58.5%) of the questions are ambiguous, and those have three intents on average. See Appendix A for more dataset statistics.

4.2 Instantiating BAG

Conversational Strategies. We experiment with three strategies for BAG: **clarify** to address uncertainty in the data (*e.g.,* ambiguity in the input), **abstain** to address uncertainty in the model (*e.g.,* about facts), and **direct answer** for little to no uncertainty. We leave other strategies, like retrieving external information (using BAG to decide when to RAG) or long-form answers, to future work. When applying BAG+ to the final assistant turn, we allow only the **direct answer** and **abstain** strategies, limiting the total number of conversational turns to four.

BAG prompts. We test three prompts with slightly different instructions for how to analyse the belief state (BAG1-3). BAG1 in Appendix Figure 6d shows the most basic variant, focusing on sample consistency. BAG2 in Appendix Figure 7 includes a Minimum Bayes Risk-like intuition, and further delineates when to abstain versus clarify. BAG3 in Appendix Figure 8 extends it with a two-step instruction: first cluster generations, then conjecture if multiple clusters represent different interpretations of the question or just factual inconsistencies (*i.e.,* hallucinations). We use a single prompt for BAG+, shown in Appendix Figure 9, and provide an overview of all prompts in Appendix Figure 12.

Models. We test models of various size and focus: the 7B and 13B OLMo2-instruct, 7B OLMo3-instruct, 8B and 14B Qwen3, and Gemini-2.5-Flash.[2] We use $K = 10$ unbiased samples to construct the belief state and use each model's recommended decoding setting (temperature, top-k, top-p and min-p) to generate user-facing responses.

[2]We disable thinking mode because of the extremely large token cost, especially in Olmo3 and Qwen3.

	Direct Generation				Augmented Generation			
	One intent		Any intent		One intent			
	Standard	Disambig	Standard	Disambig	SAG+	BAG1+	BAG2+	BAG3+
OLMo2-7B	33.9	36.3	44.8	41.2	41.5	44.7	42.8	41.7
OLMo2-13B	45.3	49.6	59.2	55.9	46.8	54.9	55.6	55.1
OLMo3-7B	24.7	25.2	32.4	29.9	29.5	39.8	41.8	52.9
Qwen3-8B	33.1	33.7	43.2	40.3	34.6	60.2	45.0	40.8
Qwen3-14B	40.1	42.8	52.4	49.3	42.8	63.8	58.8	60.4
Gemini-2.5-Flash	59.1	68.7	75.6	74.2	67.4	68.8	69.8	68.4

Table 1: Accuracy on AmbigQA for direct generation (left) and augmented generation (right) in both the first and final assistant turns (SAG+/BAG+). All BAG+ prompt/model configurations outperform standard generation and the prompt-only baseline SAG+. For 17/18 configurations, BAG+ also outperforms the disambiguation oracle. Colours are row-normalised.

5 Evaluation

We introduce four generation settings. For an overview of the prompts, see Appendix Figure 6.

Direct generation. The **standard baseline** generates a response given the original question. The **disambiguation oracle** uses an (annotated) disambiguated question from AmbigQA, if one exists, or the original question otherwise (collapsing to the standard baseline). We consider this an oracle for when and how to clarify, removing ambiguity and evaluating if models have the relevant factual knowledge to answer a particular intent.

Augmentation methods. The main augmentation method conditions on the belief state: **Belief-Augmented Generation (BAG)**, with an optional second round for turn 4: **BAG+**. We also introduce **Strategy-Augmented Generation (SAG)**, a prompt-only baseline that receives the question and strategy instructions, but no belief state. **SAG+** is the application of this prompt-only baseline to the fourth turn, allowing for abstention after clarifications just like BAG+.

Simulating Interaction. When the assistant model asks a clarification question, we use Gemini-2.5-Flash to role-play a user. We prompt it with a user intent (hidden from the assistant model): a disambiguated version of the question if the annotation exists, and the reference answer if the question is not ambiguous. We design a conservative prompt (Figure 10) that elicits "I don't know" if the clarification question is unanswerable or off topic to avoid rewarding bad clarification questions. Appendix B provides an error analysis—we generally find high-quality user answers.

We limit each conversation to four turns: user question, potential assistant clarification question, user response, assistant response. See Figure 1 for an example. We evaluate direct and clarify strategies against one specific user intent. Following standard selective prediction practices (*e.g.,* Kuhn et al., 2023b), we exclude questions that a model abstains on and compute accuracy on the remaining subset. An increase in accuracy means that models abstain on questions that they would have gotten wrong otherwise—which is exactly the goal.

5.1 Metrics

We measure QA accuracy with a **human reference-based** LLM judge because 1) we found many false positives and false negatives with lexical metrics, 2) to mitigate hallucinations by the LLM judge. We employ Gemini-2.5-Flash. See Appendix Figure 11 for the prompt. A small manual analysis detailed in Appendix C reveals a 97% correctness rate of our reference-based LLM judge. In addition, we encourage the reader to manually explore questions, generations, user simulations, and judge verdicts with the BAG explorer.

For the direct generation settings, we evaluate ambiguous questions with multiple intents against **one** (randomly chosen) intent and **any** of the possible intents.

One intent. This is the core metric evaluating whether models give the correct answer to a specific user intent. This metric is strict and may be unfair without the potential for interaction, *i.e.,* it may reject generations that answer another intent, even though a user might judge the generation's relevance themselves, potentially following up by making their intent more explicit.

Any intent. Do models give a correct answer w.r.t. any of the possible intents? We consider this an upper bound that does not care about ambiguity.

6 Results

We first analyse direct generation results, with no possibility to clarify or abstain. See Table 1, left. We care about the relative differences between generation settings, not models. Therefore, colours are row-normalised.

Disambiguations cannot fix lack of knowledge. The disambiguation oracle yields relatively small improvements over the standard single intent baseline for some models (*e.g.,* Qwen3-8B), and much larger improvements for others (*e.g.,* Gemini-2.5-Flash). This demonstrates that some models are unable to exploit disambiguations due to a lack of factual knowledge, rather than an inability to handle ambiguity. This indicates varying and sometimes limited potential for clarification, further motivating why BAG supports both clarification and abstention.

Specific intents are difficult. The gap between the standard one vs. any intent metric reflects the inherent difficulty of recovering a specific user intent without interaction or disambiguation—a model may give a perfectly reasonable answer while missing the target interpretation. Still, the one intent metric is valuable when considering that users may ask an ambiguous question while having a specific intent in mind, especially if they cannot foresee which interpretation is salient to the model.

Models presuppose a "salient" interpretation. The disambiguation oracle's improvement from the one intent to any intent metric indicates that models may presuppose a salient interpretation so strongly that they ignore the disambiguation, perhaps not having the relevant factual knowledge to support other intents. This aligns with Sedova et al. (2024) who find biases toward preferred readings of entities that correlate with the frequency of those readings in training data. This may also limit the effect that clarifications have on changing such salient interpretations.[3]

Performance differences across models. Finally, while not relevant to evaluating BAG, Table 1 shows that Gemini-2.5-Flash performs best with 59.1% accuracy on the one intent metric, followed by Olmo2. Olmo2 outperforms Qwen3 and Olmo3, which is consistent with results from the original authors on PopQA (Olmo et al., 2025) and may reflect the newer Qwen3's and Olmo3's relative focus on reasoning and instruction following over factual knowledge. Gemini-2.5-flash is also better at exploiting disambiguations, with an accuracy increase of 9.6 points. Since we do not have access to training data, there are few conclusions we can draw from this. Moreover, while it may seem that the Gemini-2.5-Flash judge is unfairly favouring itself, recall that we use it as a reference-based LLM judge, with explicit instructions to use the AmbigQA references as authoritative (Section 5.1).

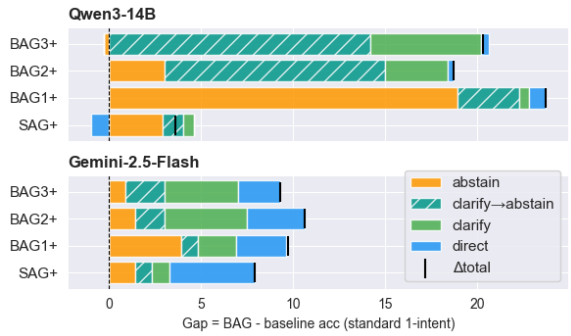


Figure 3: The contribution of each strategy to BAG+'s total accuracy increase (shown on the x-axis). Abstain is the driving force for Qwen3, while for Gemini the contributions are more balanced.

6.1 Augmented Generation

Table 1 (right) reports our main results for belief-augmentation of both the first and the final assistant turn (BAG+). To isolate the contribution of the first round, Appendix Table 3 reports BAG applied to the first assistant turn only.

A single round of BAG yields substantial gains: it outperforms the standard one intent baseline in 17 out of 18 model–prompt configurations (the exception being Qwen3-8B with BAG3); the prompt-only SAG in 15 out of 18; and even the disambiguation oracle in 11 out of 18. The best prompt variant per model improves over the standard one intent baseline by 12.4 points averaged across models (a 34% relative improvement), ranging from 5.8 (OLMo2-7B) to 25.7 (Qwen3-8B), and over SAG by 10.2 points.

Augmenting the final assistant turn as well (BAG+) improves results further: in Table 1 (right), all BAG1–3+ variants outperform both the standard baseline and SAG+; and BAG+ outperforms the disambiguation oracle in 17 out of 18 configura-

[3]To investigate to what extent interpretations salient to models align with interpretations salient to humans, we need more and more diverse annotations than AmbigQA offers.

tions. The best prompt variant per model improves over the standard baseline by 18.5 points averaged across models (a 55% relative improvement), and over the disambiguation oracle by 15.1 points. We attribute this to the assistant model often lacking the relevant factual knowledge to answer correctly even after clarification—something we also observed for the disambiguation oracle in Section 6. BAG+ is able to catch this in its updated belief state and abstains rather than incorrectly guessing.

6.1.1 Decomposing BAG Performance

To understand if and how each strategy contributes, we decompose BAG+'s accuracy improvement over the standard direct generation baseline in Figure 3. Orange shows the contribution of abstaining (by excluding those examples from evaluation); green the improvement of the final answer after clarification over the baseline on the clarify-subset; and blue of the direct answers relative to the standard baseline on the direct-subset. Every strategy contributes positively but to varying degree. For Qwen, abstaining is the driving force behind its accuracy improvement, in line with its low baseline and disambiguation oracle scores in Table 1. Appendix Figures 13,14 show other models.

6.1.2 Strategy routing

This decomposition captures both *what* strategy is chosen and the *quality* of the actual response. We next isolate *strategy routing*: how often and when are strategies chosen? Figure 4 shows the frequency of each strategy per model and prompt. While the direct answer strategy is generally most often chosen there are big differences across models. We hypothesise that this is due to structurally different belief states with varying levels of consistency (*i.e.,* uncertainty) and verbosity, which we further explore in Section 7. For example, the best model, Gemini-2.5-Flash, most often answers directly, exhibiting less uncertainty.

There are also differences across prompts. These are partly by design: *e.g.,* BAG3 instructs models to cluster generations and conjecture if and which interpretation of the user's question might lead to each cluster. Models can almost always come up with such interpretations, which explains the high rate of clarification. However, instruction following is also a factor: Gemini-2.5-Flash—the best instruction follower—shows substantially less variation across BAG prompt variants. Next, we attempt to evaluate the strategy routing: on what kind of questions was each strategy chosen?

Accuracy. Table 2 shows accuracy of the non-augmented, standard direct generation baseline on strategy subsets defined by augmented generation in the first assistant turn. This tells us if the harder questions are rightfully being routed to clarify or abstain, while the easier ones are answered directly. Indeed, baseline accuracy on the direct answer subset *acc(D)* is much higher than on the abstain subset, sometimes as much as 5-7x (Qwen3), whereas the clarified-subset accuracy lies in between. This indicates that augmented generation leads to sensible strategy decisions. Also, belief-augmented prompts make better routing decisions than the prompt-only SAG variant, demonstrated by higher accuracy on the direct and lower accuracy on the abstain subset.

Ambiguity. The clarify subsets contain the highest share of ambiguous questions across models, shown by the *amb(C)* column being dark blue. However, this share is still only 53.5-76.7%, which is consistent with related work showing that detecting ambiguity or when to clarify is a very challenging task (Zhang et al., 2024; Tanjim et al., 2025; Zhang et al., 2025). We posit that ambiguity detection in factual QA is inherently very difficult: recognizing multiple intents given a deceptively straightforward question requires both an understanding of ambiguity and highly specialised factual knowledge (*e.g.,* that a different person performed a song on tour), and we suspect that even humans would struggle to mark these questions as ambiguous. That said, since LLMs are increasingly used as a substitute for search engines (Chatterji et al., 2025), the bar may arguably be higher.

6.2 Effect of Clarification

The ideal conditions to clarify are: (1) a direct answer would have been wrong, (2) the clarification question is relevant and a user would be able to answer it, (3) without the user's answer, a model could not give the correct answer (*e.g.,* RAG would not help since a model does not know what information to fetch or fetches conflicting information), and finally (4) the model has the factual knowledge to answer the disambiguated question.

Section 6 shows that condition 1 is often met (direct generation accuracy is low) and condition 4 is not (factual knowledge, not ambiguity, is often the problem, for which abstain is a better strategy). This limits the potential for clarification, which we

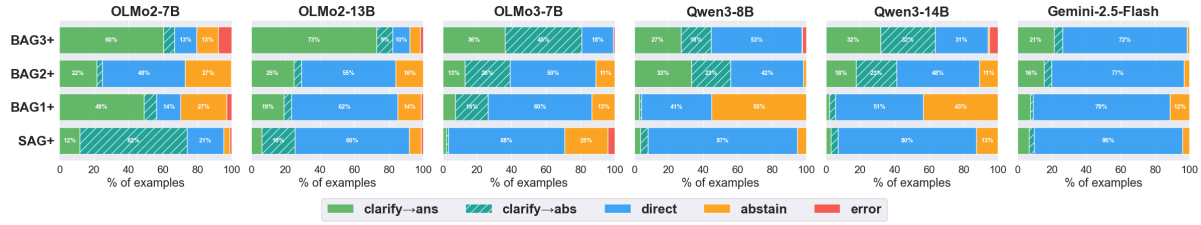


Figure 4: The strategies that models pick can vary a lot based on the model class and instructions. For example, larger models often choose to answer directly across prompts, indicating less uncertainty, whereas OLMo2-7B clarifies a lot.

		acc(C)	acc(D)	acc(A)	amb(C)	amb(D)	amb(A)
OL2 7B	S	32.6	38.6	32.4	53.5	57.1	63.2
	B	28.7	45.8	31.5	55.1	52.5	54.4
OL2 13B	S	35.5	50.0	32.5	53.8	55.4	51.6
	B	37.5	57.0	29.4	56.4	51.5	53.2
OL3 7B	S	19.4	27.0	19.2	58.1	54.2	56.8
	B	13.9	36.9	11.1	55.9	54.3	48.7
Qw3 8B	S	14.5	36.1	13.0	64.1	54.4	45.0
	B	13.2	57.2	8.6	57.0	55.2	51.0
Qw3 14B	S	18.6	45.2	19.6	69.0	54.2	50.0
	B	16.5	68.5	12.2	61.3	53.7	53.8
Gem 2.5Flash	S	37.5	63.1	27.3	76.7	52.8	44.2
	B	33.5	67.3	19.5	63.9	53.6	45.1

Table 2: How good is the choice of strategy in the first assistant turn? We group questions by the strategy a model predicts (**C**larify / **D**irect / **A**bstain) and report standard baseline **acc**uracy (left) and share of **amb**iguous questions (right) per group. For example, **acc(C)** is standard baseline accuracy on questions the model decides to clarify. Ideally **acc(D)** is high, **acc(A)** and **acc(C)** low, and **amb(C)** high. Colours are row-normalised. S: **S**AG, B: average over **B**AG1–3.

confirm in Figure 3. We will qualitatively investigate conditions 2,3 next.

Belief states display both *interpretation variation* due to multiple plausible interpretations of the question, and *hallucination variation* due to inconsistent facts, falling neatly on opposite ends of the data-model uncertainty spectrum (Baan et al., 2023). Interpretation variation should be addressed by interacting with the user, and hallucination variation by abstaining. However, we find that detecting this difference is challenging. BAG guides models to formulate a clarification question based on their belief state. When the belief state contains multiple plausible answers, each contextualised (*e.g., "in the US, ..."* vs *"in Canada, ..."*, or *"the FIFA world cup ..."* vs *"The rugby world cup..."*), the clarifications are generally useful, like the examples in Figures 1-2.

However, models sometimes come up with non-existent ambiguity to explain hallucination variation, especially if generations are not meaningfully contextualised. For example, non-existing adaptations of a movie in which different actors play the same character to explain mentions of those actors in the belief state. Put differently: clarification questions reflect what Kim et al. (2024) call *perceived ambiguity*, which does not necessarily align with how humans perceive ambiguity in the real world. While such clarification questions are truthful reflections of what models deem plausible—and arguably a better strategy than responding with one of the incorrect generations—they may still confuse a user.

6.3 Computational Cost

BAG's computational cost, like other sample-based approaches (*e.g.,* MBR, semantic entropy, self-consistency, verbalised uncertainty) is K samples, plus an additional prompt to reason over them. The optional BAG+ adds another round. However, BAG goes beyond a single decision by simultaneously determining 1) when to answer (selective prediction), 2) how to answer (self-consistency, MBR, verbalised uncertainty), and 3) when and how to clarify.

Preliminary tests reveal that Qwen3 thinking mode consumes 7x more output tokens than direct generation, for just 2 more accuracy points. This cost is not much higher than sampling 10 generations, which, unlike thinking, is parallelisable. This puts BAG's costs in perspective.

To reduce token cost, we study the effect of two brevity-inducing prompts: *"Please provide a concise answer to the following question:"*, and *"(...) short answer of at most 1 sentence (...)"*. They reduce average generation length by up to a factor 10 without severely impacting accuracy (Appendix Table 6). However, the verbal cues for how models interpret a question (*e.g., "the FIFA world cup was in ..."*) are often lost (examples in Appendix Table 4), reducing the means by which users can judge

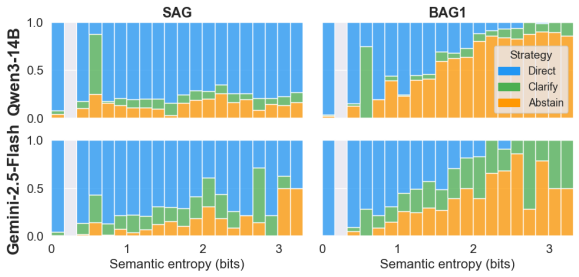


Figure 5: Routing decisions vs belief state entropy: BAG is more faithful to the predictive distribution than SAG, answering directly when entropy is low, increasingly clarifying and abstaining as entropy increases.

an answer, and BAG a belief state.

7 Faithfulness

We evaluated BAG in terms of QA performance, but equally important is: are its strategic decisions faithful to its internal distribution?

We cluster generations on the main answer they assert and compute semantic entropy (Kuhn et al., 2023b) with Gemini-2.5-Flash. We find that belief states range from just a single answer to ten totally distinct answers. For smaller models, belief states are almost uniformly distributed across this range, while Gemini-2.5-Flash has many more single-answer belief states; about 50%; see Appendix Figure 16.

BAG's strategies are more faithful. Our hypothesis is that models will choose the direct answer strategy for lower entropy, and clarify or abstain for higher entropy belief states. Indeed, Figure 5 confirms this for Qwen3-14B and Gemini-2.5-Flash with BAG. Interestingly, the prompt-only SAG is not aligned with belief state entropy, providing further evidence that prompt-based uncertainty methods do not exploit the 'internal' predictive distribution. We observe similar patterns for Olmo3 and Qwen3-8B, but not for Olmo2, which we explain by its limited ability to follow instructions.

Baselines rarely clarify or abstain. Direct generation without any augmentation strategy rarely clarifies or abstains at all: classifying generations with Gemini-2.5-Flash we find they only do so on 3% of the questions. A small manual analysis (Appendix Figure 15) confirms this, aligning with findings from previous work (Deng et al., 2023; Shaikh et al., 2024; Zhang et al., 2024; Testoni et al., 2025). This confirms that direct generation is even less faithful to the internal distribution.

Clarification reduces entropy. We sample K responses after clarification interactions (similar to BAG+), cluster them, and compute the difference between their semantic answer entropy and entropy of the turn 2 responses reported earlier. We find that clarification reduces entropy for all models. Averaged across BAG1-3, Gemini has the largest reduction of 54%, followed by Olmo2-13B at 20%, Qwen3-14B at 18%, Qwen3-8B at 16%, Olmo2-7B at 14%, and Olmo3-7B at 1.7%. This aligns with the disambiguation oracle results, which indicated that some models do not have the factual knowledge to exploit even annotated disambiguations.

8 Conclusion

Belief-augmented generation unifies disparate sampling-based uncertainty methods into a single decision making framework. It combines a probabilistic view with modern instruction and reasoning capabilities to decide *when* and *how* to answer directly, clarify, or abstain. In our interactive ambiguous QA setting, BAG improves accuracy across six models and produces strategic decisions more faithful to the belief state than the prompt-only baselines, while standard direct generation rarely clarifies or abstains at all. Disentangling (semantic) variation in belief states due to hallucinations versus different interpretations—and thus when to clarify versus abstain—remains challenging.

There are many promising avenues for future work. For example, experimenting with different strategies (*e.g.,* tool calling), deploying BAG in more autonomous, task-driven settings (*e.g.,* booking a trip with ambiguous instructions), and enriching the belief state (*e.g.,* including thinking traces, combining the power of reasoning models with a more traditional statistical perspective).

9 Acknowledgements

We thank Dennis Ulmer and Evgenia Ilia for their valuable suggestions and insightful discussions, the members of Amsterdam's Dialogue Modelling Group and Mulini lab for their feedback, as well as the anonymous reviewers and meta-reviewer. This project received funding from the ELLIS Amsterdam Unit. BP is supported by the ERC Consolidator Grant DIALECT 101043235.

10 Limitations

While AmbigQA is well suited to simulate different user intents due to its multiple disambiguation

annotations, being similarly used by several recent studies due to a lack of good alternatives, its references can be outdated and its disambiguations slightly artificial: rather than the actual intent of the user who asked a question—impossible to reconstruct after the fact—AmbigQA instead collects all possible answers via Wikipedia search and retroactively create (minimally edited) disambiguated versions of the original question. An important avenue of research is therefore to collect new datasets with real world questions and intents that correspond to actual user intents, ideally from users with diverse cultural backgrounds (*e.g.,* AmbigQA is very US-centered), which also allows comparison between models' suppositions and whether they align with dominant user intents.

Another limitation is the quality of the belief states, *i.e.,* the model's representation of uncertainty. While belief states provide richer insight into the range of responses probable under a model, and are successfully exploited for decision making tasks like decoding or selective prediction, they still merely reflect probability under a model, not reality (*e.g.,* highly consistent belief states with a single main claim can still be factually incorrect). It is important to improve the quality of belief states, which can then be more effectively exploited by methods such as BAG.

Finally, by using BAG as a test-time scaling method we incur a computational cost during inference. However, BAG can be extended for synthetic data creation, allowing models to be finetuned on uncertainty-guided strategies without additional inference costs.

References

Vaibhav Adlakha, Parishad BehnamGhader, Xing Han Lu, Nicholas Meade, and Siva Reddy. 2024. Evaluating correctness and faithfulness of instruction-following models for question answering. *Transactions of the Association for Computational Linguistics*, 12:681–699.

Chinmaya Andukuri, Jan-Philipp Fränken, Tobias Gerstenberg, and Noah Goodman. 2024. STar-GATE: Teaching language models to ask clarifying questions. In *First Conference on Language Modeling*.

Joris Baan, Nico Daheim, Evgenia Ilia, Dennis Ulmer, Haau-Sing Li, Raquel Fernández, Barbara Plank, Rico Sennrich, Chrysoula Zerva, and Wilker Aziz. 2023. Uncertainty in natural language generation: From theory to applications. *arXiv preprint arXiv:2307.15703*.

Joris Baan, Raquel Fernández, Barbara Plank, and Wilker Aziz. 2024. Interpreting predictive probabilities: Model confidence or human label variation? In *Proceedings of the 18th Conference of the European Chapter of the Association for Computational Linguistics (Volume 2: Short Papers)*, pages 268–277, St. Julian's, Malta. Association for Computational Linguistics.

Jonathan Berant, Maximillian Chen, Adam Fisch, Reza Aghajani, Fantine Eri Huot, Mirella Lapata, and Jacob Eisenstein. 2026. Learning steerable clarification policies with collaborative self-play. In *Third Conference on Language Modeling*.

Amanda Bertsch, Alex Xie, Graham Neubig, and Matthew Gormley. 2023. It's MBR all the way down: Modern generation techniques through the lens of minimum Bayes risk. In *Proceedings of the Big Picture Workshop*, pages 108–122, Singapore. Association for Computational Linguistics.

Dan Bohus and Alexander I Rudnicky. 2005a. Constructing accurate beliefs in spoken dialog systems. In *IEEE Workshop on Automatic Speech Recognition and Understanding, 2005.*, pages 272–277. IEEE.

Dan Bohus and Alexander I Rudnicky. 2005b. A principled approach for rejection threshold optimization in spoken dialog systems. In *Proc. Interspeech 2005*, pages 2781–2784.

Aaron Chatterji, Thomas Cunningham, David J Deming, Zoe Hitzig, Christopher Ong, Carl Yan Shan, and Kevin Wadman. 2025. How people use chatgpt. Working Paper 34255, National Bureau of Economic Research.

Maximillian Chen, Ruoxi Sun, Tomas Pfister, and Sercan O Arik. 2025. Learning to clarify: Multi-turn conversations with action-based contrastive self-training. In *The Thirteenth International Conference on Learning Representations*.

Jeremy Cole, Michael Zhang, Daniel Gillick, Julian Eisenschlos, Bhuwan Dhingra, and Jacob Eisenstein. 2023. Selectively answering ambiguous questions. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pages 530–543, Singapore. Association for Computational Linguistics.

Yang Deng, Lizi Liao, Liang Chen, Hongru Wang, Wenqiang Lei, and Tat-Seng Chua. 2023. Prompting and evaluating large language models for proactive dialogues: Clarification, target-guided, and non-collaboration. In *Findings of the Association for Computational Linguistics: EMNLP 2023*, pages 10602–10621, Singapore. Association for Computational Linguistics.

Bryan Eikema and Wilker Aziz. 2020. Is MAP decoding all you need? the inadequacy of the mode in neural machine translation. In *Proceedings of the 28th International Conference on Computational Linguistics*, pages 4506–4520, Barcelona, Spain (Online). International Committee on Computational Linguistics.

Bryan Eikema, Evgenia Ilia, José GC de Souza, Chrysoula Zerva, and Wilker Aziz. 2025. Teaching language models to faithfully express their uncertainty. *arXiv preprint arXiv:2510.12587*.

Sebastian Farquhar, Jannik Kossen, Lorenz Kuhn, and Yarin Gal. 2024. Detecting hallucinations in large language models using semantic entropy. *Nature*, 630(8017):625–630.

Tianyu Gao, Howard Yen, Jiatong Yu, and Danqi Chen. 2023. Enabling large language models to generate text with citations. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pages 6465–6488, Singapore. Association for Computational Linguistics.

Yifan Gao, Henghui Zhu, Patrick Ng, Cicero Nogueira dos Santos, Zhiguo Wang, Feng Nan, Dejiao Zhang, Ramesh Nallapati, Andrew O. Arnold, and Bing Xiang. 2021. Answering ambiguous questions through generative evidence fusion and round-trip prediction. In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, pages 3263–3276, Online. Association for Computational Linguistics.

Jiahui Geng, Fengyu Cai, Yuxia Wang, Heinz Koeppl, Preslav Nakov, and Iryna Gurevych. 2024. A survey of confidence estimation and calibration in large language models. In *Proceedings of the 2024 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (Volume 1: Long Papers)*, pages 6577–6595, Mexico City, Mexico. Association for Computational Linguistics.

Mario Giulianelli, Joris Baan, Wilker Aziz, Raquel Fernández, and Barbara Plank. 2023. What comes next? evaluating uncertainty in neural text generators against human production variability. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pages 14349–14371, Singapore. Association for Computational Linguistics.

Eric Horvitz and Tim Paek. 2000. DeepListener: Harnessing expected utility to guide clarification dialog in spoken language systems. In *Proceedings of the 6th International Conference on Spoken Language Processing (ICSLP 2000)*, pages 226–229.

Bairu Hou, Yujian Liu, Kaizhi Qian, Jacob Andreas, Shiyu Chang, and Yang Zhang. 2024. Decomposing uncertainty for large language models through input clarification ensembling. In *Proceedings of the 41st International Conference on Machine Learning*, ICML'24. JMLR.org.

Ehsan Kamalloo, Nouha Dziri, Charles Clarke, and Davood Rafiei. 2023. Evaluating open-domain question answering in the era of large language models. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 5591–5606, Toronto, Canada. Association for Computational Linguistics.

Hyuhng Joon Kim, Youna Kim, Cheonbok Park, Junyeob Kim, Choonghyun Park, Kang Min Yoo, Sanggoo Lee, and Taeuk Kim. 2024. Aligning language models to explicitly handle ambiguity. In *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing*, pages 1989–2007, Miami, Florida, USA. Association for Computational Linguistics.

Michael Kirchhof, Luca Füger, Adam Golinski, Eeshan Gunesh Dhekane, Arno Blaas, and Sinead Williamson. 2025. Self-reflective uncertainties: Do llms know their internal answer distribution? In *ICML 2025 Workshop on Reliable and Responsible Foundation Models*.

Lorenz Kuhn, Yarin Gal, and Sebastian Farquhar. 2023a. Clam: Selective clarification for ambiguous questions with generative language models. *Workshop on Challenges in Deployable Generative AI at International Conference on Machine Learning (ICML)*.

Lorenz Kuhn, Yarin Gal, and Sebastian Farquhar. 2023b. Semantic uncertainty: Linguistic invariances for uncertainty estimation in natural language generation. In *The Eleventh International Conference on Learning Representations*.

Tom Kwiatkowski, Jennimaria Palomaki, Olivia Redfield, Michael Collins, Ankur Parikh, Chris Alberti, Danielle Epstein, Illia Polosukhin, Jacob Devlin, Kenton Lee, Kristina Toutanova, Llion Jones, Matthew Kelcey, Ming-Wei Chang, Andrew M. Dai, Jakob Uszkoreit, Quoc Le, and Slav Petrov. 2019. Natural questions: A benchmark for question answering research. *Transactions of the Association for Computational Linguistics*, 7:452–466.

Dongryeol Lee, Segwang Kim, Minwoo Lee, Hwanhee Lee, Joonsuk Park, Sang-Woo Lee, and Kyomin Jung. 2023. Asking clarification questions to handle ambiguity in open-domain QA. In *Findings of the Association for Computational Linguistics: EMNLP 2023*, pages 11526–11544, Singapore. Association for Computational Linguistics.

Patrick Lewis, Ethan Perez, Aleksandra Piktus, Fabio Petroni, Vladimir Karpukhin, Naman Goyal, Heinrich Küttler, Mike Lewis, Wen-tau Yih, Tim Rocktäschel, and 1 others. 2020. Retrieval-augmented generation for knowledge-intensive nlp tasks. *Advances in neural information processing systems*, 33:9459–9474.

Zongxi Li, Yang Li, Haoran Xie, and S. Joe Qin. 2025. CondAmbigQA: A benchmark and dataset for conditional ambiguous question answering. In *Proceedings of the 2025 Conference on Empirical Methods in Natural Language Processing*, pages 2269–2288, Suzhou, China. Association for Computational Linguistics.

Sewon Min, Kenton Lee, Ming-Wei Chang, Kristina Toutanova, and Hannaneh Hajishirzi. 2021. Joint passage ranking for diverse multi-answer retrieval. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pages 6997–7008, Online and Punta Cana, Dominican Republic. Association for Computational Linguistics.

Sewon Min, Julian Michael, Hannaneh Hajishirzi, and Luke Zettlemoyer. 2020. AmbigQA: Answering ambiguous open-domain questions. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 5783–5797, Online. Association for Computational Linguistics.

Yang Nan, Pengfei He, Ravi Tandon, and Han Xu. 2025. Can multiple responses from an LLM reveal the sources of its uncertainty? In *Findings of the Association for Computational Linguistics: EMNLP 2025*, pages 15551–15569, Suzhou, China. Association for Computational Linguistics.

Team Olmo, Allyson Ettinger, Amanda Bertsch, Bailey Kuehl, David Graham, David Heineman, Dirk Groeneveld, Faeze Brahman, Finbarr Timbers, Hamish Ivison, and 1 others. 2025. Olmo 3. *arXiv preprint arXiv:2512.13961*.

Tim Paek and Eric Horvitz. 2000. Conversation as action under uncertainty. In *Proceedings of the Sixteenth Conference on Uncertainty in Artificial Intelligence*, UAI'00, page 455–464, San Francisco, CA, USA. Morgan Kaufmann Publishers Inc.

Sergey Pletenev, Maria Marina, Nikolay Ivanov, Daria Galimzianova, Nikita Krayko, Mikhail Salnikov, Vasily Konovalov, Alexander Panchenko, and Viktor Moskvoretskii. 2025. Will it still be true tomorrow? multilingual evergreen question classification to improve trustworthy QA. In *Proceedings of the 2025 Conference on Empirical Methods in Natural Language Processing*, pages 8614–8631, Suzhou, China. Association for Computational Linguistics.

Irina Saparina and Mirella Lapata. 2026. Reasoning About Intent for Ambiguous Requests. In *Third Conference on Language Modeling*.

Anastasiia Sedova, Robert Litschko, Diego Frassinelli, Benjamin Roth, and Barbara Plank. 2024. To know or not to know? analyzing self-consistency of large language models under ambiguity. In *Findings of the Association for Computational Linguistics: EMNLP 2024*, pages 17203–17217, Miami, Florida, USA. Association for Computational Linguistics.

Omar Shaikh, Kristina Gligoric, Ashna Khetan, Matthias Gerstgrasser, Diyi Yang, and Dan Jurafsky. 2024. Grounding gaps in language model generations. In *Proceedings of the 2024 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (Volume 1: Long Papers)*, pages 6279–6296, Mexico City, Mexico. Association for Computational Linguistics.

Charlie Victor Snell, Jaehoon Lee, Kelvin Xu, and Aviral Kumar. 2025. Scaling LLM test-time compute optimally can be more effective than scaling parameters for reasoning. In *The Thirteenth International Conference on Learning Representations*.

Ivan Stelmakh, Yi Luan, Bhuwan Dhingra, and Ming-Wei Chang. 2022. ASQA: Factoid questions meet long-form answers. In *Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing*, pages 8273–8288, Abu Dhabi, United Arab Emirates. Association for Computational Linguistics.

Mirac Suzgun, Luke Melas-Kyriazi, and Dan Jurafsky. 2023. Follow the wisdom of the crowd: Effective text generation via minimum Bayes risk decoding. In *Findings of the Association for Computational Linguistics: ACL 2023*, pages 4265–4293, Toronto, Canada. Association for Computational Linguistics.

Mehrab Tanjim, Yeonjun In, Xiang Chen, Victor Bursztyn, Ryan A. Rossi, Sungchul Kim, Guang-Jie Ren, Vaishnavi Muppala, Shun Jiang, Yongsung Kim, and Chanyoung Park. 2025. Disambiguation in conversational question answering in the era of LLMs and agents: A survey. In *Proceedings of the 2025 Conference on Empirical Methods in Natural Language Processing*, pages 9548–9561, Suzhou, China. Association for Computational Linguistics.

Alberto Testoni and Raquel Fernández. 2024. Asking the right question at the right time: Human and model uncertainty guidance to ask clarification questions. In *Proceedings of the 18th Conference of the European Chapter of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 258–275, St. Julian's, Malta. Association for Computational Linguistics.

Alberto Testoni, Barbara Plank, and Raquel Fernández. 2025. RAcQUEt: Unveiling the dangers of overlooked referential ambiguity in visual LLMs. In *Proceedings of the 2025 Conference on Empirical Methods in Natural Language Processing*, pages 23638–23658, Suzhou, China. Association for Computational Linguistics.

Roman Vashurin, Ekaterina Fadeeva, Artem Vazhentsev, Lyudmila Rvanova, Daniil Vasilev, Akim Tsvigun, Sergey Petrakov, Rui Xing, Abdelrahman Sadallah, Kirill Grishchenkov, Alexander Panchenko, Timothy Baldwin, Preslav Nakov, Maxim Panov, and Artem Shelmanov. 2025. Benchmarking uncertainty quantification methods for large language models with LM-polygraph. *Transactions of the Association for Computational Linguistics*, 13:220–248.

Xuezhi Wang, Jason Wei, Dale Schuurmans, Quoc V Le, Ed H. Chi, Sharan Narang, Aakanksha Chowdhery, and Denny Zhou. 2023. Self-consistency improves chain of thought reasoning in language models. In *The Eleventh International Conference on Learning Representations*.

Jason Williams, Antoine Raux, Deepak Ramachandran, and Alan Black. 2013. The dialog state tracking challenge. In *Proceedings of the SIGDIAL 2013 Conference*, pages 404–413, Metz, France. Association for Computational Linguistics.

Ian Wu, Patrick Fernandes, Amanda Bertsch, Seungone Kim, Sina Khoshfetrat Pakazad, and Graham Neubig. 2025. Better instruction-following through minimum bayes risk. In *The Thirteenth International Conference on Learning Representations*.

Tianyang Xu, Shujin Wu, Shizhe Diao, Xiaoze Liu, Xingyao Wang, Yangyi Chen, and Jing Gao. 2024. SaySelf: Teaching LLMs to express confidence with self-reflective rationales. In *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing*, pages 5985–5998, Miami, Florida, USA. Association for Computational Linguistics.

Gal Yona, Roee Aharoni, and Mor Geva. 2024. Can large language models faithfully express their intrinsic uncertainty in words? In *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing*, pages 7752–7764, Miami, Florida, USA. Association for Computational Linguistics.

Michael Zhang and Eunsol Choi. 2021. SituatedQA: Incorporating extra-linguistic contexts into QA. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pages 7371–7387, Online and Punta Cana, Dominican Republic. Association for Computational Linguistics.

Michael JQ Zhang and Eunsol Choi. 2025. Clarify when necessary: Resolving ambiguity through interaction with LMs. In *Findings of the Association for Computational Linguistics: NAACL 2025*, pages 5526–5543, Albuquerque, New Mexico. Association for Computational Linguistics.

Michael JQ Zhang, W. Bradley Knox, and Eunsol Choi. 2025. Modeling future conversation turns to teach LLMs to ask clarifying questions. In *The Thirteenth International Conference on Learning Representations*.

Tong Zhang, Peixin Qin, Yang Deng, Chen Huang, Wenqiang Lei, Junhong Liu, Dingnan Jin, Hongru Liang, and Tat-Seng Chua. 2024. CLAMBER: A benchmark of identifying and clarifying ambiguous information needs in large language models. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 10746–10766, Bangkok, Thailand. Association for Computational Linguistics.

A Dataset Statistics

Each AmbigQA question is annotated with a single answer (*i.e.,* intent), marking the question as unambiguous, or multiple disambiguated question-answer pairs (*i.e.,* , intents), marking it as ambiguous. The dataset contains 10,036 train and 2,002 development questions, with 10,251 and 3,312 annotations respectively (multiple annotators may annotate the same question). The development set has a higher ambiguity rate (58.5% vs. 47.3% for train) and annotation clash rate (*i.e.,* questions with both an ambiguous and unambiguous annotation: 8.5% vs. 0.1% for train—we filter those questions out). We evaluate on the full development set (the test set is not public), filtering out 170 questions with conflicting annotations, resulting in 1,832 questions. Questions average 9.0 words and answers 2.5 words across both splits. Ambiguous questions have on average 3.17 (±2.03) intents in dev and and 2.94 (±1.33) in train.

B LLM User Answer Simulator Error Analysis

The user answer simulation prompt for Gemini-2.5-flash is shown in Figure 10. We find that simulated user answers are generally of high quality. The user simulator is explicitly instructed not to reveal the reference answer and only to answer the clarification question. Nevertheless, references occasionally leak through. On average, across all assistant models and prompts, the user answer contains the reference answer for 1.7% of clarification questions.

However, some clarification questions already contain the reference answer themselves, which is not a leak on the user simulator's part. Excluding those cases, only 0.7% of user answers constitute true leaks. To assess their impact on our results, we count how often leaks co-occur with a positive clarification flip (direct answer incorrect → final answer correct), finding this in only 1.9% of such flips. We therefore conclude that reference leaks do not meaningfully inflate the reported clarification gains.

The prompt instructs the user simulator not to answer clarification questions that are irrelevant to its intent or off-topic, to avoid rewarding bad CQs with useful additional information about the users intent. On average, about 5% of the user responses are an exact match with "I don't know". This simulation strategy is very conservative, and we suspect real users to be much more helpful.

For example, consider the AmbigQA question *"What is the dog called in midsomer murders?"* with intent *"What is the dog's breed called in midsomer murders?"* and the model's clarification question *"There isn't a single dog that consistently*

(A) DIRECT GENERATION STANDARD BASELINE

```
Who was the female singer on Gimme Shelter?
```

(B) DISAMBIGUATION UPPER BOUND

```
Who was the female singer on the recorded version of Gimme
      Shelter?
```

(C) STRATEGY-AUGMENTED GENERATION (SAG)

```
# [1. Setting]
You are an AI assistant and your goal is to answer
a user's question directly, ask a clarification
question, or abstain.

# [3. Strategies \& Instructions]
- A direct answer is effective when the question is
  clear and you know the right answer.
- A clarification question is effective if it resolves
  ambiguity or underspecification in the user question,
  and if the user can answer it without needing to know
  the answer to their own question.
- Abstaining is effective if the question is clear but
  you don't know the right answer.

# [4. Context]
User question: Who was the female singer on Gimme Shelter?

# [5. Output format]
Format your response as:
STRATEGY: [DIRECT_ANSWER/CLARIFICATION_QUESTION/ABSTAIN]
REASONING: [...]
RESPONSE: [...]
```

(D) BELIEF-AUGMENTED GENERATION (BAG)

```
# [1. Setting] (same as SAG)
# [2. Belief state]
Below, you are given {K} candidate answers to the
user's question. Consider these your state of
knowledge and decide on your next action based on
the following criteria:

# [3. Strategy \& Instructions]
- A direct answer is effective when the question is
  clear and the candidate answers are consistent.
- A clarification question is effective if it resolves
  ambiguity or underspecification in the user question,
  and if the user can answer it without needing to know
  the answer to their own question.
- Abstaining is effective if the question is clear but
  the candidate answers are too contradictory to be
  reliable.

# [4. Context]
User question: Who was the female singer on Gimme Shelter?
Candidate answers: {A_1, A_2, ..., A_K}

# [5. Output format] (same as SAG)
```

Figure 6: The four generation settings. The direct generation baseline (a) uses the original question. The disambiguation upper bound (b) uses a disambiguation annotation. The SAG baseline (c) prompts for conversational strategies without a belief state. BAG (d) extends SAG by conditioning on K generation samples. See Fig 7 and 8 for other two BAG variants.

appears throughout all of Midsomer Murders with one name, as there have been different dogs belonging to the various DCI Barnabys. To give you the most accurate answer, could you tell me which DCI Barnaby you're thinking of (Tom Barnaby, played by John Nettles, or John Barnaby, played by Neil Dudgeon)?". The user simulator answers *"I don't know"*, with the following reasoning *"Internal reasoning: The user is asking about the dog's breed, not its name. The AI's clarification question is about the DCI Barnaby to identify the dog's name. Since the user is asking about the breed, the specific DCI Barnaby is not relevant to the breed of dog. Therefore, the AI's clarification question is not helpful for the user's actual question."*. A real user might just respond with the much more helpful *"Actually, I'm looking for the breed"*.

C LLM Judge Error Analysis

Figure 11 shows the two prompts we use for the reference-based LLM judge. We generally find the judge verdicts of Gemini-2.5-Flash about whether a generation is correct with respect to a reference of high quality, and manually evaluate 30 direct generation answers from Olmo2-13B against the same single reference answer that the judge is evaluating against (the 1 intent setting, *i.e.,* leftmost prompt in Figure 11). We find just one error, *i.e.,* a success rate of 97%.

This error is actually an interesting case where the judge ignores the reference and uses its own knowledge to judge the generation as correct for the question "The cat in the hat boy and girl?". In fact, the generation does correctly answer another (much more obvious) interpretation of the question (the boy and the girl characters are unnamed), while being explicit about doing so, which makes it hard to fault the generation. This is a limitation of single-intent evaluation setting, because a real user might be able to judge that the generation is answering a different interpretation of the question than what they actually intended (*e.g.,* the annotated disambiguation "Who plays the girl in the 2003 Cat in the Hat film?"), and would potentially follow up to clarify this to the model.

C.1 Reference Quality

A more important limitation of the LLM judge is not its ability to judge whether a generation contains a reference, but the quality of the AmbigQA reference itself. In Table 5 we characterise several types of questions and their annotated disambiguation-reference pairs. One notable category is questions with answers that change over time.

```
You are a helpful AI assistant in conversation with a user
      that asked a question.

Below, you are given {len(belief_samples)} sampled
      generations from a language model given the user's
      question. Consider these your belief state - a
      representation of your uncertainty about the answer.

Your job is to analyze these generations and choose one of
      three strategies:

- DIRECT_ANSWER: The generations converge on the same answer
       - there is no meaningful uncertainty. Synthesize the
      most representative response from your belief state,
      at roughly the same length as the individual
      generations.
- CLARIFICATION_QUESTION: The generations reflect multiple
      semantically distinct but plausible answers,
      indicating the question is ambiguous or underspecified.
       Ask a clarification question that the user can answer
       by specifying their intent. Never ask the user to
      answer their own question.
- ABSTAIN: The generations are factually contradictory and
      inconsistent due to your lack of knowledge, AND no
      useful clarification question exists that would
      resolve the uncertainty. Politely decline to answer.

User question: {question}
Candidate answers:
{belief_state_text}

Format your output as:
STRATEGY: [DIRECT_ANSWER/CLARIFICATION_QUESTION/ABSTAIN]
REASONING: [A compact reasoning chain explaining your
      decision]
RESPONSE: [Your response to the user's question based on the
       (hidden) candidate answers, following your chosen
      strategy.]

Please provide your output.
```

Figure 7: Prompt variant BAG2.

Outdated references. This problem is universal across QA datasets and discussed by Pletenev et al. (2025). We apply their pretrained classifier to filter out time-sensitive questions, and find it discards about 30%. When we filter those out, QA accuracy increases with a few points across all models and configurations, but the relative differences between generation settings remain the same. We hypothesise that the effect is so small because all methods are impacted equally by the filtering. We choose not to include these results in the paper because we manually observed that many subtly non-mutable questions are also being filtered out (*e.g.,* "Who did Samuel L. Jackson play in Star Wars?", or "Who played James Bond in Licence to Kill?").

Artificial ambiguity. One issue with the dataset is that the disambiguation annotations reflect what annotators could find on Wikipedia rather than what interpretation users genuinely meant when they asked their question to Google Search.

```
You are a helpful AI assistant in conversation with a user.

Below are {K} candidate answers representing your belief
      state - your uncertainty about the answer to the user'
      s question.

Analyze them in two steps, then choose a strategy and
      respond.

Step 1 - Cluster by meaning: Group the answers by what they
      assert. Ignore surface variation (wording, punctuation
      ); group by the underlying fact or claim. Give each
      group a letter label and a representative answer.

Step 2 - Interpret each cluster: For each group, ask: "What
      interpretation of the user's question would naturally
      lead to this answer?" Write one interpretation per
      group. If you cannot identify any coherent
      interpretation for a group, mark it "uninterpretable."

Step 3 - Choose a strategy:
- DIRECT_ANSWER: there is only 1 meaningful cluster, or all
      clusters share a single interpretation. Give a
      complete, direct answer synthesized from the dominant
      cluster.
- CLARIFICATION_QUESTION: there are 2-3 clusters, each with
      a distinct coherent interpretation, and the user can
      specify their intent without already knowing the
      answer. Ask about the user's context or purpose - do
      NOT simply ask "do you mean X or Y?", instead ask
      about the situation or intent that would determine
      which interpretation applies.
- ABSTAIN: clusters are uninterpretable, too numerous to
      resolve, or no clarification question could
      realistically distinguish them. Politely decline.

User question: {question}
Candidate answers:
{belief_state_text}

Format your output as:
CLUSTERS: [one line per group, e.g. "A ({{k}}/{n}): <
      representative answer>"]
INTERPRETATIONS: [one line per group, e.g. "A -> <what
      interpretation of the question leads here>"]
STRATEGY: [DIRECT_ANSWER / CLARIFICATION_QUESTION / ABSTAIN]
REASONING: [one compact sentence explaining your routing
      decision]
RESPONSE: [your response to the user - they do not see the
      candidate answers, clusters, or reasoning]
```

Figure 8: Prompt variant BAG3.

D Conversational Strategies without Augmentation

To get a sense of how standard models employ conversational actions, we select 33 random AmbigQA questions with multiple reference answers from the train set and inspect generations from the smallest OLMO2-7B-INSTRUCT and QWEN3-8B (non-thinking mode) models.

We annotate each generation with: **clarification question**, **refusal**, **multiple answers**, **(partially) contextualised answer**, or **direct answer** without any interpretation context.

Figure 15 shows that models use a narrow range of conversational actions and rarely clarify or refuse. They mostly answer directly ($> 94\%$), which we can further break down into (partially) contextualising their answer by assuming a specific interpretation of the question ($45 - 54\%$ *e.g., "In*

	Direct Generation				Augmented Generation			
	1 intent		any intent		1 intent			
	Standard	Disambig	Standard	Disambig	SAG	BAG1	BAG2	BAG3
OLMo2-7B	33.9	36.3	44.8	41.2	33.5	37.0	39.7	35.4
OLMo2-13B	45.3	49.6	59.2	55.9	44.6	51.7	51.8	47.4
OLMo3-7B	24.7	25.2	32.4	29.9	29.6	31.5	31.0	26.7
Qwen3-8B	33.1	33.7	43.2	40.3	33.4	58.8	35.1	32.6
Qwen3-14B	40.1	42.8	52.4	49.3	41.8	60.6	45.5	42.6
Gemini-2.5-Flash	59.1	68.7	75.6	74.2	66.0	67.9	67.2	66.0

Table 3: Accuracy on AmbigQA for direct generation (left) and augmented generation applied only to the first assistant turn, i.e. a single round of BAG (right). BAG outperforms the standard baseline in 17/18 configurations, SAG in 15/18, and the disambiguation oracle in 11/18. Colours are row-normalised; see Table 1 for BAG+.

Question	**Verbose generation (excerpt)**	**Single-sentence generation**
Which is the most populated country in Europe?	"…Russia was Europe's most populous country, although it's important to note that **part of Russia lies within Asia**…"	"Russia is the most populous country in Europe."
Who has the record for the most strikeouts?	"The **all-time Major League Baseball** record for the most strikeouts **by a pitcher** is held by Nolan Ryan with 5,714 strikeouts…"	"Nolan Ryan holds the MLB record for most strikeouts with 5,714."
When do we celebrate Veterans Day this year?	"Veterans Day is observed on November 11 each year **in the United States**…"	"Veterans Day will be celebrated on November 11, 2023."
When did Harry Potter and the Deathly Hallows Part 1 come out?	"…was released on November 19, 2010, **in the United States**…"	"Harry Potter and the Deathly Hallows Part 1 was released on November 19, 2010."
When did the first episode of Dragon Ball air?	"The first episode of 'Dragon Ball' aired on February 26, 1986…"	"The first episode of 'Dragon Ball' aired **in Japan** on February 26, 1986."

Table 4: Effect of brevity-inducing prompts on ambiguity acknowledgement for *olmo2-13b-instruct*. Bold text marks phrases that explicitly resolve an interpretation. When prompted for single-sentence answers, the model drops disambiguating context in four out of five cases, committing to an interpretation without signalling it.

the US ..."), giving multiple answers ($\sim 24-30\%$), or answering directly without acknowledging ambiguity in any way ($15-20\%$).

```
Below are {n} candidate answers representing your belief
      state - independent attempts at answering the original
       question given our conversation so far.

Determine whether your belief state shows enough consensus
      to give a confident answer, or whether the candidates
      diverge too much (indicating you are guessing rather
      than knowing).

Step 1 - Assess consensus: Do the {n} candidate answers
      agree on the same core factual claim? Ignore minor
      surface differences in wording - focus on whether they
       assert the same fact. Count how many samples support
      the most common claim.

Step 2 - Route:
- DIRECT_ANSWER: The candidates largely agree (> 70% support
       the same claim). Synthesize a single confident answer
       based on the consensus.
- ABSTAIN: The candidates make multiple different factual
      claims with no clear majority. This means you are
      uncertain and should not guess. Politely decline.

Candidate answers:
{belief_state_text}

Format your output as:
CONSENSUS: [one line: "X/{n} candidates agree on: <the claim
      >" or "no consensus - candidates split across N
      different claims"]
STRATEGY: [DIRECT_ANSWER / ABSTAIN]
REASONING: [one compact sentence explaining your decision]
RESPONSE: [your answer to the user if DIRECT_ANSWER, or a
      polite decline if ABSTAIN]
```

Figure 9: Prompt for applying BAG a second time after the clarification interaction with the user, to determine whether to reply or abstain after all, because the dialogue history does not reduce uncertainty sufficiently. This prompt contains an example for how to use an explicit threshold in the prompt: if there are less than 7/10 consistent answers the instruction is to abstain.

```
Pretend you are roleplaying a user that
asked a question to an AI assistant.

The AI assistant asked a clarification
question. Your task is to formulate the
user's response to this clarification
question given secret additional context.
You will be given a disambiguated version
of the user question to help resolve any
ambiguity or underspecification.

Rules:
- Never reveal the final answer to the
  user question, only answer the AI's
  clarification question.
- If the AI's clarification question is
  impossible to answer or not helpful,
  please respond "I don't know".

User question (turn 1): {question}
Secret disambiguated user question:
  {disambig_question}
AI clarification question (turn 2):
  {clarification_question}
User answer (turn 3): [your answer]

Format your response as:
REASONING: [A compact reasoning chain
explaining your answer]
USER ANSWER: [Your simulated user answer
to the AI's clarification question]
```

```
Pretend you are roleplaying a user that
asked a question to an AI assistant.

The AI assistant asked a clarification
question. Your task is to formulate the
user's response to this clarification
question given secret additional context.
You will be given the final reference
answer to the user question to help
resolve any ambiguity or underspecification.

Rules:
- Never reveal the final answer to the
  user question, only answer the AI's
  clarification question.
- If the AI's clarification question is
  impossible to answer or not helpful,
  please respond "I don't know".

User question (turn 1): {question}
Secret final reference answer: {reference}
AI clarification question (turn 2):
  {clarification_question}
User answer (turn 3): [your answer]

Format your response as:
REASONING: [A compact reasoning chain
explaining your answer]
USER ANSWER: [Your simulated user answer
to the AI's clarification question
(DON'T MENTION {reference})]
```

Figure 10: The two prompts for the user simulator LLM. The two prompts differ in the secret context provided: for ambiguous questions (left), we give a disambiguated version of the question; for non-ambiguous questions (right) we give the reference answer itself with an explicit instruction not to reveal it.

```
Your task is to evaluate whether a model answer correctly
      responds to a factual question.

Correct answer: {ref_text}

The correct answer above is ground truth - treat it as
      authoritative regardless of your own knowledge.

Guidelines:
- CORRECT ("yes"): the model directly asserts the correct
      answer as its main claim. Semantically equivalent
      forms are accepted (abbreviations, name variants,
      different date formats, singular/plural).
- INCORRECT ("no"): the model (a) states a different answer
      as its main claim, (b) mentions the correct answer
      only incidentally (e.g. as one item in a list, in a
      historical aside, or as a counterexample without
      endorsing it), or (c) is too vague or non-committal to
       constitute a clear answer.

Question: {question}
Model answer: {candidate}

Respond in this exact format:
REASONING: [one sentence explaining your decision]
VERDICT: yes / no
```

```
  Your task is to evaluate whether a model answer correctly
      responds to a factual question.

This question has multiple valid interpretations. The model
      answer is correct if it directly asserts ANY ONE of
      the following correct answers as its main claim:

{refs_block}

The correct answers above are ground truth - treat them as
      authoritative regardless of your own knowledge.

Guidelines:
- CORRECT ("yes"): the model directly asserts any one
      correct answer as its main claim. Semantically
      equivalent forms are accepted (abbreviations, name
      variants, different date formats, singular/plural).
- INCORRECT ("no"): the model (a) states a different answer
      as its main claim for none of the interpretations, (b)
       mentions a correct answer only incidentally (e.g. as
      one item in a list, in a historical aside, or as a
      counterexample without endorsing it), or (c) is too
      vague or non-committal to constitute a clear answer.

Question: {question}
Model answer: {candidate}

Respond in this exact format:
REASONING: [one sentence explaining your decision]
VERDICT: yes / no
```

Figure 11: The two prompts for the LLM judge. On the left is the prompt to assess against a single user intent, the right assesses against any of the intents.



Who plays the queen on game of thrones?

Turn 1: User question (AmbigQA)

1. Belief state

1. (...) **Emilia Clarke**. She portrays **Daenerys Targaryen**, also known as the Mother of Dragons.
2. (...) **Lena Headey**, who portrays **Queen Cersei Lannister**.
3. ...

2. BAG output

Reasoning: The candidate answers provide conflicting information about who plays 'the queen' on Game of Thrones. Some answers (...)
Strategy: clarify
Response: Could you clarify which queen you're referring to? For example, are you asking about Daenerys Targaryen or Cersei Lannister?



Turn 2: BAG (belief-augmented generation) on first assistant turn

1. **Construct belief state**: sample K generations given the question (prompt Fig6a)

2. **Belief-augmented generation** (prompt Fig7)



I'm asking about Margaery Tyrell.

Turn 3: LLM user (prompt Fig10)

1. Updated belief state

1. (...) **Natalie Dormer** plays **Margaery Tyrell** in Game of Thrones.
2. (...) **Natalie Dormer** plays **Margaery Tyrell** in Game of Thrones.
3. ...

2. BAG+ output

Reasoning: Nine out of ten candidates consistently state that Natalie Dormer plays Margaery Tyrell, with only one candidate incorrectly naming Natalie Dreyfuss.
Strategy: answer
Response: Margaery Tyrell is played by actress Natalie Dormer in Game of Thrones.



Turn 4: BAG+ (belief-augmented generation on final assistant turn)

1. Construct updated belief state: sample K generations given turn 1-3 (prompt Fig6a plus turn 2 and 3)

2. Belief-augmented generation+ (prompt Fig9)



Verdict: yes
Reasoning: The model correctly identifies Natalie Dormer as the actress who plays a queen on Game of Thrones."

LLM judge (prompt Fig11)

Figure 12: An overview of all BAG(+) pipeline steps, referencing each prompt used. Real AmbigQA dev example with ID 3312525434538504362. Generation output from Qwen3-14B with concise brevity induced generation, BAG2 (Fig. 7) and BAG+ (Fig. 9) prompts, and Gemini-2.5-Flash as user simulator (Fig. 10), and judge (Fig. 11).

#	Type	Original question	Annotated disambiguations & ref answers
1	Genuine	*Who made the song "These Boots Are Made for Walkin' '?*	wrote → Lee Hazlewood
			sang → Nancy Sinatra
2	Genuine	*Who the female singer on Gimme Shelter?*	recorded version → Merry Clayton
			on tour → Lisa Fischer
3	Time-sensitive	*Who is the **current** chairman of the African Union Commission?*	became chairman in 2017 → Moussa Faki
			became chairman in 2012 → Nkosazana Dlamini-Zuma
			became chairman in 2008 → Jean Ping
4	Time-sensitive	*When did the **current** queen of England take the throne?*	[single answer] → 6 February 1952 *(outdated since 2022)*
5	Missing ambiguity	*When was the movie Rudolph the red-nosed reindier made*	original short → 1948
			remake → 1998 *(misses iconic 1964 Rankin/Bass TV special)*
6	Missing ambiguity	*What channel is the new show FBI on?*	[single answer] → CBS *(US-centric; varies by country)*
7	Granularity	*'In which episode of Smallville does Jonathan die?*	title → reckoning
			number → season 5 episode 12
8	Granularity	*What do the symbols on the Indian flag represent?*	the wheel → Dharma Chakra
			the saffron stripe → courage and sacrifice
			the white stripe → peace and truth
			the green stripe → faith and chivalry
			the Calcutta flag lotuses → the eight provinces
9	Marginal interp.	*How long do contestants get to answer on Jeopardy?*	regular question → 5 seconds
			Final Jeopardy → 30 seconds
			online contestant test → 15 seconds
10	Marginal interp.	*What is the population of Venice Italy 2018?*	total population → 260.897
			population density → 1,600/sq mi

Table 5: AmbigQA examples. Examples 1–2 show genuine ambiguity well suited to clarification. Examples 3–4 show time sensitivity questions that are annotated as ambiguity, and can also become outdated. Example 5–6 shows ambiguity the dataset *misses*, *e.g.,* due to US-centric annotations. Examples 7–8 show overly granular disambiguations. Example 9–10 includes a marginal interpretation alongside a very salient one.

		Direct Generation					Augmented Generation			
		1 intent			any intent		1 intent			
		Standard	Disambig	Standard	Disambig	words	SAG+	BAG1+	BAG2+	BAG3+
OLMo2-7B	free	33.9	36.3	44.8	41.2	149.9	41.5	44.7	42.8	41.7
	concise	31.9	34.7	42.5	40.4	42.9	41.5	45.1	44.5	43.7
	sentence	31.8	34.1	41.6	39.2	13.5	41.5	42.7	47.4	42.8
OLMo2-13B	free	45.3	49.6	59.2	55.9	146.3	46.8	54.9	55.6	55.1
	concise	43.4	45.9	56.7	53.5	53.0	46.8	53.7	55.8	55.0
	sentence	42.3	46.3	55.4	53.3	14.6	46.8	54.0	53.4	55.8
OLMo3-7B	free	24.7	25.2	32.4	29.9	125.5	29.5	39.8	41.8	52.9
	concise	26.5	27.0	34.9	31.7	35.3	29.5	46.8	51.6	52.9
	sentence	24.5	26.0	33.0	31.3	14.0	29.5	43.8	51.1	49.9
Qwen3-8B	free	33.1	33.7	43.2	40.3	111.0	34.6	60.2	45.0	40.8
	concise	32.0	34.3	41.6	40.5	18.3	34.6	49.7	43.2	39.2
	sentence	30.6	33.6	40.5	38.5	13.3	34.6	46.1	40.4	38.8
Qwen3-14B	free	40.1	42.8	52.4	49.3	99.3	42.8	63.8	58.8	60.4
	concise	39.6	42.4	52.0	49.6	23.6	42.8	60.4	59.2	57.2
	sentence	37.5	41.6	49.8	48.4	14.7	42.8	52.6	54.9	54.8
Gemini-2.5-Flash	free	59.1	68.7	75.6	74.2	46.5	67.4	68.8	69.8	68.4
	concise	52.4	61.6	67.9	66.8	5.9	67.4	63.2	63.9	63.0
	sentence	55.6	66.2	71.4	70.4	12.7	67.4	67.0	65.5	65.7

Table 6: QA performance on 1832 AmbigQA-dev questions for all brevity-inducing settings. Colours per-row normalized. We can see that direct generation accuracy is not greatly impacted, but that the average direct generation length is, and therefore the belief state, with about a factor 10 for the open models from no (free) to sentence brevity. However, brevity-inducing prompts do affect BAG+ accuracy. There may be a trade-off between the size and complexity of belief states, and corresponding difficulties of analysing such large context windows, especially for smaller older models like Olmo2-7B, whereas, bigger models may more easily leverage the extra context.

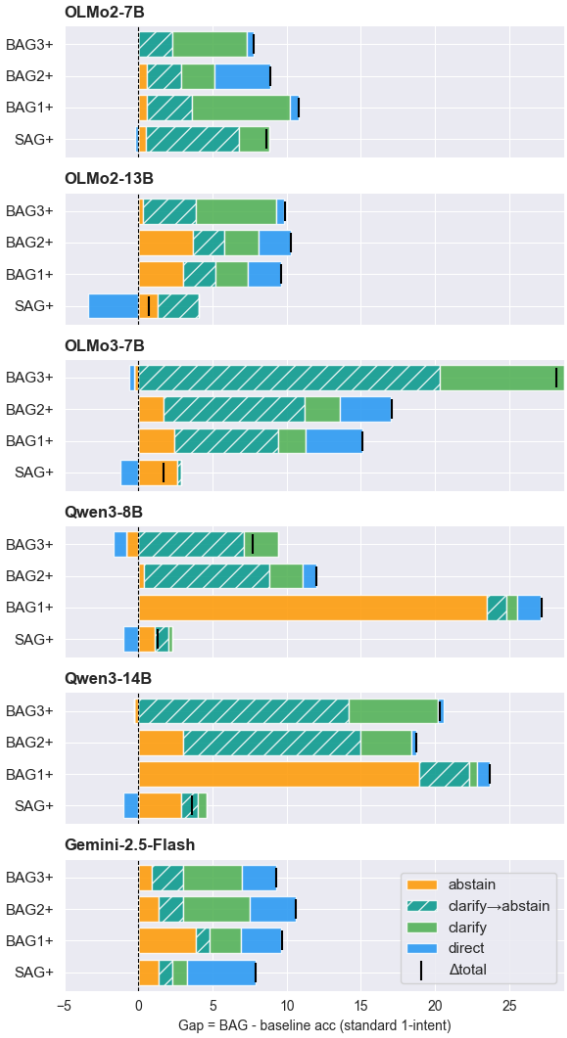


Figure 13: The contribution of each strategy to BAG+'s total accuracy increase (shown on the x-axis) for all models.

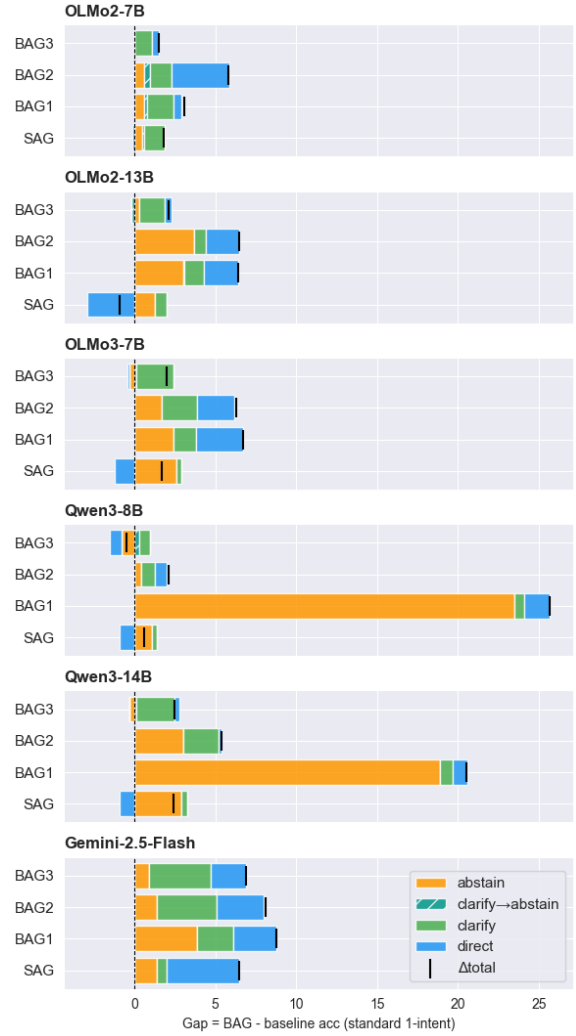


Figure 14: The contribution of each strategy to BAG's (so no second round of BAG for the final turn) total accuracy increase (shown on the x-axis) for all models.

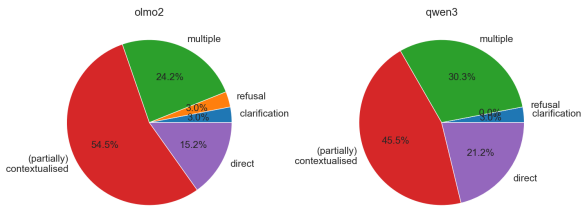


Figure 15: Strategy distribution for *olmo2-7b-instruct* and *qwen3-8b (non-thinking)* generations on 33 ambiguous AmbigQA questions with multiple intents. We manually annotated the generations with a single strategy each.

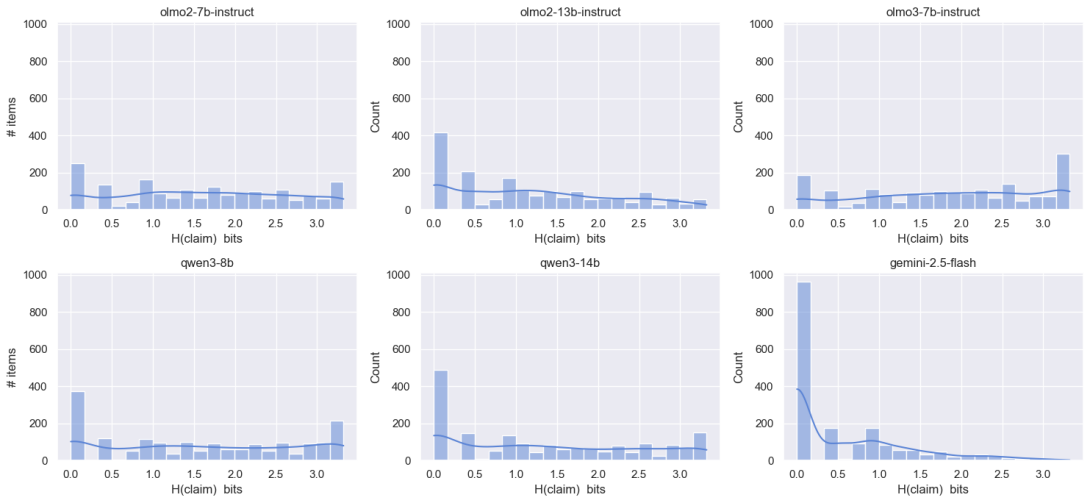


Figure 16: The distribution of belief state entropies across models.

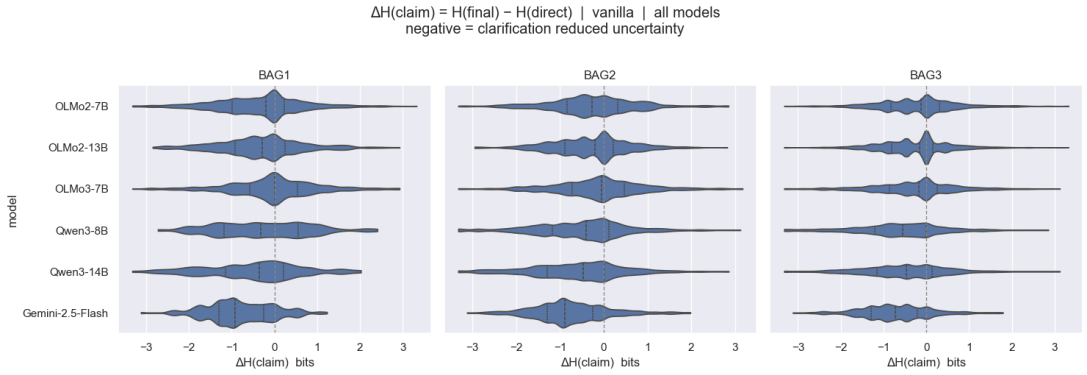


Figure 17: Uncertainty reduction after a clarification interaction for every model/prompt: largest for Gemini and smallest for Olmo3.