From Passive Metric to Active Signal: The Evolving Role of Uncertainty Quantification in Large Language Models

Jiaxin Zhang[1*] **Wendi Cui**[2] **Zhuohang Li**[3],
Lifu Huang[4] **Bradley Malin**[3,5] **Caiming Xiong**[1] **Chien-Sheng Wu**[1]
[1]Salesforce AI Research [2]Intuit [3]Vanderbilt University
[4] University of California, Davis [5]Vanderbilt University Medical Center

Abstract

While Large Language Models (LLMs) show remarkable capabilities, their unreliability remains a critical barrier to deployment in high-stakes domains. This survey charts a functional evolution in addressing this challenge: the evolution of uncertainty from a passive diagnostic metric to an active control signal guiding real-time model behavior. We demonstrate how uncertainty is leveraged as an active control signal across three frontiers: in **advanced reasoning** to optimize computation and trigger self-correction; in **autonomous agents** to govern metacognitive decisions about tool use and information seeking; and in **reinforcement learning** to mitigate reward hacking and enable self-improvement via intrinsic rewards. By grounding these advancements in emerging theoretical frameworks like Bayesian methods and Conformal Prediction, we provide a unified perspective on this transformative trend. This survey provides a comprehensive overview, critical analysis, and practical design patterns, arguing that mastering the new trend of uncertainty is essential for building the next generation of scalable, reliable, and trustworthy AI.

1 Introduction

LLMs have demonstrated unprecedented capabilities across a wide range of natural language tasks, marking a milestone in AI. Yet their inherent unreliability, which manifests through factual errors, biases, and hallucinations, remains a critical barrier to deployment in high-stakes domains such as medicine, law, and finance (Bommasani et al., 2022; Farquhar et al., 2024). To address this issue, Uncertainty Quantification (UQ) has emerged as a key technology for enhancing trustworthiness. Traditionally, UQ has focused on the **post-hoc evaluation** and calibration of outputs (Zhang, 2021; Xiong et al., 2024). Methods based on Bayesian inference, ensembles, or information-theoretic metrics aim to provide confidence scores for single-turn generations, effectively measuring "how much the model knows" about its own response (Gawlikowski et al., 2023). While foundational, this function treats uncertainty as a **passive, diagnostic metric** attached to completed outputs. Yet such an approach is insufficient for the next generation of LLM systems, which involve multi-step reasoning, interactive environments, and alignment with complex human values (Kirchhof et al., 2025; Zhang et al., 2026a,b).

The importance of this field has spurred a series of excellent surveys. Some organize the landscape around **uncertainty estimation**, including token-level analysis, consistency checks, semantic clustering, and entropy (Xia et al., 2025b; Cui et al., 2024; Shorinwa et al., 2025; Kuhn et al., 2023; Gao et al., 2024; Zhang et al., 2023). Others adopt **theory-grounded perspectives**, linking heuristics to Bayesian and information-theoretic principles (Huang et al., 2024). Broader work has examined **confidence calibration** (Geng et al., 2024), while recent efforts have begun to **rethink the definition and sources** of uncertainty in the LLM lifecycle, categorizing them along new dimensions such as computational cost or reasoning uncertainty (Beigi et al., 2024; Liu et al., 2025c; Li et al., 2025b).

While aforementioned resources provide valuable overviews of how uncertainty can be *measured*, this paper complements that body of work by surveying an emerging technological trend: the evolution of uncertainty from a passive metric to an **active, real-time control signal**. This enables systems that can *"know what they don't know"* (Kadavath et al., 2022; Yin et al., 2023) and take action based on this self-awareness (Betley et al., 2025).

Our key contribution is to categorize and analyze research where uncertainty functions as a control mechanism. While prior work focused on **how to measure** uncertainty, we focus on **how to use** it,

*Correspondence to jiaxin.zhang@salesforce.com

organizing the discussion around three domains where this functional evolution is most evident:

- **Advanced Reasoning:** How uncertainty guides dynamic reasoning strategies, optimizes computational effort, and triggers self-correction.
- **Autonomous Agents:** How uncertainty drives decisions on tool use, information seeking, and risk management in interactive settings.
- **RL and Reward Models:** How modeling uncertainty in human preferences and rewards enables more robust alignment and mitigates failure modes like reward hacking.

By tracing uncertainty's evolving role from passive evaluation to active control, we provide a comprehensive overview of this emerging frontier and outline the fundamental challenges and future research directions.

2 The Limits of Traditional UQ

The classical paradigm of UQ provides a foundational, but ultimately limited, framework for assessing the reliability of LLMs. Traditionally, UQ distinguishes between *aleatoric* uncertainty, arising from inherent data noise, and *epistemic* uncertainty, stemming from the model's lack of knowledge and reducible with more data (Kendall and Gal, 2017). The principal objective has been **post-hoc evaluation**, where a confidence score is assigned after a model generates an output (Xia et al., 2025b; Shorinwa et al., 2025; Tian et al., 2023).

While useful for simple generation tasks, this "generate-then-evaluate" function treats uncertainty as a **passive, diagnostic metric**. Its inability to provide real-time, actionable feedback becomes especially limiting in the complex, dynamic, and interactive settings that characterize frontier LLM applications (Kirchhof et al., 2025). There are several shortcomings of this strategy:

- **Inapplicability to Multi-Step Reasoning:** In chain-of-thought reasoning, early mistakes can derail entire sequences. A final post-hoc score is insufficient; models require continuous uncertainty signals at intermediate steps to backtrack, branch, or adapt in real time.
- **Insufficiency for Autonomous Agents:** For LLM agents, uncertainty informs various decisions, such as whether to rely on parametric knowledge, invoke tools, or seek human input. A single retrospective score on a text output does not support such proactive choices.
- **Mismatch with Dynamic and Interactive Systems:** Classical UQ assumes static, monolithic outputs. However, modern LLM systems involve branching reasoning paths, environmental interactions, and iterative alignment loops, requiring uncertainty to evolve dynamically alongside system behavior.

We believe these limitations call for a functional shift. To build robust and reliable systems, uncertainty must move beyond passive assessment and become an active control signal integrated into the model's operational loop.

The Evolving Role of Uncertainty Quantification: From Passive Metric to Active Signal

Passive Metric (Post-hoc Diagnosis)
- **When:** Assigns a score *after* generation is complete.
- **Role:** Acts as a diagnostic tool (output reliable?).
- **Nature:** Static and external to the generation process.

Active Signal (Real-time Control)
- **When:** Intervenes *during* generation via a feedback.
- **Role:** Acts as a control mechanism (trigger actions?).
- **Nature:** Dynamic and integrated into the model's operational loop.

3 Advanced Reasoning

In advanced reasoning with LLMs, uncertainty has shifted from a passive, post-hoc quality score to an active internal signal that guides decision-making: from arbitrating *between reasoning paths*, to steering trajectories *within individual reasoning path*, and allocating *cognitive effort* efficiently. Table 1 provides a comparative analysis of these frameworks, detailing for each method the specific uncertainty signal it uses (the "what") and the control mechanism through which it acts (the "how").

3.1 Between Reasoning Paths: Weighted Selection.

Inference-time scaling, where models generate many reasoning traces and then aggregate them, has become a standard strategy for improving robustness (Pan et al., 2025; Liu et al., 2025b; Zhang, 2025). Uncertainty enables nuanced selection between generated reasoning paths to improve overall accuracy.

Confidence-Weighted Selection. Recent work moves beyond the "one path, one vote" function by leveraging uncertainty as a weighting signal (Yin

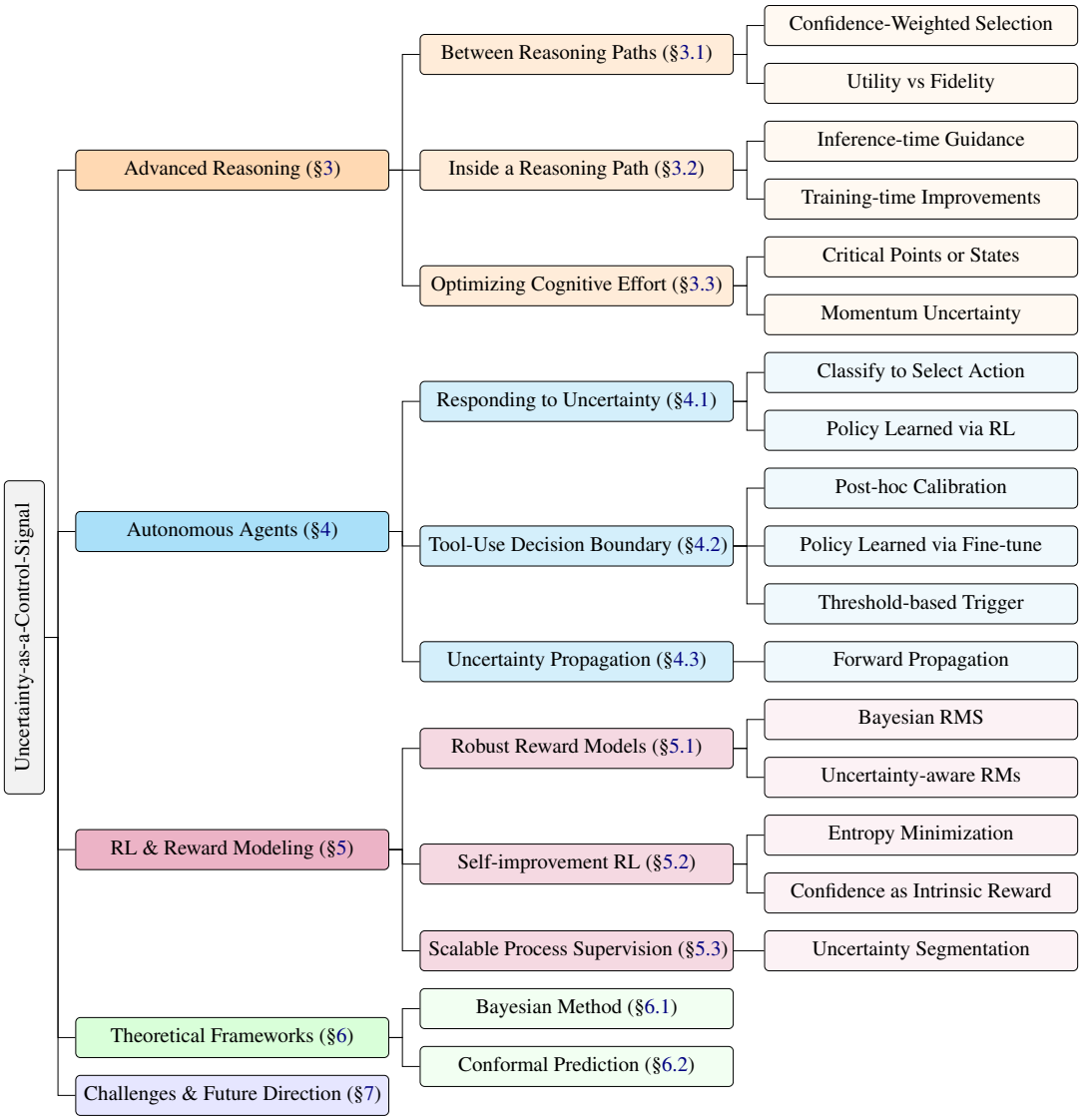

Figure 1: The taxonomy of this survey, illustrating the evolving role of uncertainty to an active control signal across advanced LLM applications, emerging theories and open challenges.

et al., 2024; Fu et al., 2025b). Confidence-Informed Self-Consistency (CISC) (Taubenfeld et al., 2025) assigns each reasoning path a holistic confidence score based on its length-normalized probability, which then weights the final vote. Confidence Enhanced Reasoning (CER) (Razghandi et al., 2025) instead evaluates confidence at crucial intermediate steps, aggregating them into a more robust score. Other approaches apply Bayesian inference to select promising paths (Yan et al., 2025b), or trained reward models to compute confidence scores (Muennighoff et al., 2025; Li et al., 2025c).

Utility vs. Fidelity Trade-off. Weighted methods expose a tension between the *utility* of confidence scores for local decisions and their *fidelity* for global calibration. As Taubenfeld et al. (2025) show, methods with strong global calibration (confidence aligning with average accuracy) often struggle to distinguish correct from incorrect reasoning paths on a single question. The key factor is **Within-Question Discrimination (WQD)**, the ability of confidence to separate right from wrong answers given one problem. A sharp, locally discriminative signal, even if globally "overconfident", is more useful for path selection. CER embodies this principle by emphasizing confidence at critical reasoning steps, favoring local discrimination over global fidelity (Razghandi et al., 2025).

These approaches illustrate a fundamental trade-off. CER's fine-grained step evaluation improves robustness in long-chain reasoning, but increases implementation complexity. By contrast, CISC's holistic scoring is simpler but more sensitive to minor, non-critical errors. Both rely on calibrated confidence estimates; when miscalibrated, the weight-

Core Concepts	Strategy Function	Uncertainty Signal (The "What")	Control Mechanism (The "How")
Between Reasoning Paths	CISC (Taubenfeld et al., 2025)	length-normalized probability	confidence-weighted voting
	CER (Razghandi et al., 2025)	step-wise confidence scores	intermediate step aggregation
	UAG (Yin et al., 2024)	step-wise uncertainty	adaptive guidance and backtracking
	Deep Think (Fu et al., 2025b)	confidence scores	weighted path selection
	Bayesian Meta-Reasoning (Yan et al., 2025b)	Bayesian inference	probabilistic path reasoning
	s1 (Muennighoff et al., 2025)	LLM/reward model scores	test-time scaling
Within Reasoning Path	SPOC (Zhao et al., 2025b)	verification uncertainty	proposer-verifier alternation
	AdaptiveStep (Liu et al., 2025d)	model confidence	uncertainty-guided segmentation
	Uncertainty-Sensitive Tuning (Li et al., 2025a)	abstention signals	two-stage training procedure
	Uncertainty-Aware FT (Krishnan et al., 2024)	prediction uncertainty	modified loss function
	BRiTE (Zhong et al., 2025)	reinforcement signals	bootstrapped thinking process
	External Slow-Thinking (Gan et al., 2025)	probability of correctness	data filtering and selection
Cognitive Effort Optimization	UnCert-CoT (Li et al., 2025b)	entropy, probability margins	threshold-based CoT activation
	MUR (Yan et al., 2025a)	momentum uncertainty	thinking budget allocation
	THOUGHT-TERMINATOR (Pu et al., 2025)	state sufficiency probability	overthinking mitigation
	TokenSkip (Xia et al., 2025a)	controllable compression signals	chain-of-thought compression

Table 1: A comparative analysis of uncertainty-aware reasoning approaches in LLMs. The table details the specific uncertainty signals and control mechanisms used across three main functions: between-path selection, within-path guidance, and cognitive effort optimization.

ing mechanism may amplify errors instead of correcting them. More details in Appendix Table 4.

3.2 Inside a Reasoning Path: Beyond Inference to Training

Within a reasoning path, uncertainty is not merely a retrospective confidence measure but an active control signal, guiding reasoning during inference and serving as a training objective (Da et al., 2025).

Inference-Time Guidance. Uncertainty provides real-time feedback that allows models to adapt their reasoning as it unfolds (Wang et al., 2025e; Hu et al., 2024). Uncertainty-Aware Adaptive Guidance (UAG) (Kamoi et al., 2024) monitors step-level uncertainty and retracts to low-uncertainty checkpoints when **reasoning drifts**. Spontaneous Self-Correction (SPOC) (Zhao et al., 2025b) assigns the model dual roles of proposer and verifier, using uncertainty to **action selection**: continuation, backtracking, or revision. AdaptiveStep (Liu et al., 2025d) aligns reasoning with natural uncertainty-guided boundaries rather than rule-based segmentation, improving supervision and interpretability. In this view, uncertainty shapes both the unfolding of reasoning and the **structural units** within it (Yin et al., 2024).

Training-Time Improvements. Uncertainty also drives advances in model training (Zhong et al., 2025). **Uncertainty-Sensitive Tuning** (Li et al., 2025a) teaches models to abstain under high uncertainty, then restores general capabilities while retaining calibrated restraint. **Uncertainty-Aware Fine-Tuning** modifies the loss function itself (Krishnan et al., 2024), rewarding higher uncertainty on ultimately incorrect predictions to produce more reliable estimates. Other approaches apply **Uncertainty-guided data filter** to emphasize plausible examples (Gan et al., 2025). These methods elevate uncertainty from a secondary signal to a primary learning objective in training.

In summary, inference-time methods offer immediate correction without retraining, but remain limited by the model's intrinsic self-correction ability. Training-time approaches incur a higher cost upfront but yield models with fundamentally stronger uncertainty awareness across downstream tasks.

3.3 Optimizing Cognitive Effort: Uncertainty as an Economic Signal

The challenge in reasoning tasks is enabling models to "think on demand," performing additional reasoning only when necessary rather than overthinking simple tasks. Uncertainty provides a low-cost control for balancing efficiency and accuracy.

Critical Points or States. UnCert-CoT (Li et al., 2025b) applies this principle to structured reasoning tasks like code generation. At critical decision *points* (e.g., the first non-indentation token of a new line), the model measures uncertainty using entropy or probability margins. If uncertainty exceeds a threshold, it activates CoT decoding; otherwise, it proceeds with direct code generation. This dynamic activation improves efficiency without sacrificing accuracy. Similarly, ThoughtTerminator (Pu

Core Concepts	Strategy Function	Uncertainty Signal (The "What")	Control Mechanism (The "How")
Responding to Uncertainty	Abstention (Stoisser et al., 2025)	entropy, perplexity, self-consistency.	pre-defined threshold trigger
	ConfuseBench (Liu et al., 2025a)	semantic entropy	classify to select an action
	UoT (Hu et al., 2024)	Expected Information Gain (EIG)	policy learned via RL
Tool-Use Decision Boundary	UALA (Han et al., 2024)	semantic entropy	threshold-based trigger
	SMARTAgent (Qian et al., 2025b)	internal uncertainty score.	policy learned via fine-tuning
	ProbeCal (Liu et al., 2024)	raw token probability	post-hoc calibration
Uncertainty Propagation	SAUP (Zhao et al., 2024)	step-wise uncertainty score (entropy)	forward propagation and aggregation
	UProp (Duan et al., 2025)	step-wise mutual information	forward propagation and combination

Table 2: A comparative analysis of uncertainty-aware LLM agents. The table details the specific uncertainty signals and control mechanisms used to enable active behaviors such as abstention, tool use, and risk management.

et al., 2025) and other related approaches (Xia et al., 2025a; Liu et al., 2025b; Fu et al., 2025b) assess whether the current *state* is sufficient to answer a question to decide whether to continue reasoning.

Momentum Uncertainty. Momentum Uncertainty Reasoning (MUR) (Yan et al., 2025a) adopts a *trajectory-level* perspective. Rather than relying on single thresholds, MUR aggregates uncertainty across steps and allocates a flexible *"thinking budget"* to regions of the reasoning path. This reduces computation by over 50% while improving accuracy through targeted resource allocation.

Threshold-based methods like **UnCert-CoT** are simple but sensitive to hyperparameters, risking under- or over-thinking. Momentum-based approaches like **MUR** offer more control but add complexity. Together, these methods highlight uncertainty as an economic signal: effective reasoning depends not only on what a model knows, but also on recognizing *when* to think harder.

4 Autonomous Agents

In LLM agents, uncertainty has evolved from a passive textual property to an active metacognitive signal that drives agentic behavior: from strategically *responding to internal states*, to governing the *tool-use decision boundary*, and managing *uncertainty propagation* in multi-step workflows.

4.1 From Abstention to Inquiry: Responding to Internal Uncertainty

For an LLM to evolve from a static generator into an autonomous agent, it must develop metacognition, that is the ability to "know what it does not know". An agent's strategic response to its own uncertainty is a key marker of intelligence, with recent research tracing an evolutionary trajectory from defensive behaviors to proactive inquiry.

The basic strategy is **passive defense**, where the agent abstains when uncertainty is high, especially for high-stakes domains (Stoisser et al., 2025). More advanced is **diagnostic response**, where the agent probes the source of its confusion, whether knowledge gaps, capability limits, or query ambiguity (Liu et al., 2025a). The most sophisticated strategy is **proactive inquiry**, where the agent learns an optimal policy for asking clarifying questions to strategically reduce future uncertainty (Hu et al., 2024). Table 2 compares distinct uncertainty signals and control mechanisms in such strategies. This evolution highlights a trade-off between autonomy and utility. Abstention ensures safety but can reduce helpfulness; proactive inquiry reflects higher intelligence but increases implementation complexity, see discussions in Table 5.

4.2 Tool-Use Decision Boundary

A key capability of modern LLM agents is leveraging external tools (e.g., search engines and APIs) to overcome the limits of parametric knowledge. This introduces a core dilemma: *when should an agent rely on internal knowledge versus incurring the cost of tool use?* Naive strategies that default to external calls risk inefficiency and "Tool Overuse" (Qian et al., 2025b; Yao et al., 2022). Recent work addresses this by using uncertainty as a control signal to set a more intelligent decision boundary.

The evolution of these strategies reveals a trajectory from reactive control to calibrated autonomy. The earliest methods use **inference-time control**, where the model generates a preliminary answer and invokes tools only when real-time uncertainty is high, improving efficiency (Han et al., 2024). More advanced approaches pursue **training-time self-awareness**, fine-tuning agents on specialized datasets to internalize knowledge boundaries and develop calibrated intrinsic policies for tool use (Qian et al., 2025b). Another line of work focuses on **uncertainty calibration**, showing that by calibrating the control signal, agents achieve more

reliable tool-use decisions (Liu et al., 2024).

The shift from inference-time control to training-time self-awareness reflects a trade-off between ease and robustness. Threshold-based inference-time methods are simple but brittle, while training-based policies are expensive yet yield stronger domain adaptation. A shared limitation remains: most approaches decide *whether* to call a tool, but not *how* to handle uncertainty or error in the tool's own outputs, leaving a key challenge for future work, see more comparative analysis in Table 2.

4.3 Uncertainty Propagation in Multi-step Workflows

In complex multi-step tasks, uncertainty is dynamic: small errors can accumulate and propagate through a workflow, ultimately leading to task failure. Traditional uncertainty methods typically assess single-turn outputs and overlook this compounding effect (Cemri et al., 2025). Building reliable long-horizon agents requires explicitly modeling how uncertainty evolves across the "thought–action–observation" cycle.

Recent frameworks address this by tracking and propagating uncertainty throughout decision-making. The situation-awareness uncertainty propagation (**SAUP**) framework (Zhao et al., 2024) is to track uncertainty at each step and weight its importance based on the context. Recognizing that not all uncertainties are equally critical, SAUP introduces "situational weights" that amplify the uncertainty score of steps deemed more pivotal. In contrast, the **UProp** framework (Duan et al., 2025) provides an information-theoretic foundation, decomposing total uncertainty into *Intrinsic Uncertainty (IU)* at the current step and *Extrinsic Uncertainty (EU)* inherited from previous steps.

These approaches highlight a critical shift in the *source* of uncertainty. In reasoning-only tasks, uncertainty is largely cognitive and internal, whereas in agentic systems, the environment itself becomes a dominant driver. The different mechanisms for modeling uncertainty propagation, as detailed in Table 2 and 5, represent different approaches to capturing the risks that arise from an agent's interaction with a dynamic and unpredictable world.

4.4 Multi-Agent Systems

As research advances from single agents to multi-agent systems (MAS), uncertainty challenges are not simply scaled but fundamentally transformed. Uncertainty now arises both within each agent's internal reasoning and in the communication and interactions between agents (Hu et al., 2025; Barbi et al., 2025; Hazra et al., 2025). A key concern is that uncertainty can propagate and amplify across interactions. An agent may receive uncertain or incorrect information from a peer, yet treat it as factual, causing cascades of errors that destabilize the collective (Hu et al., 2025). Analyses of MAS failures highlight **inter-agent misalignment** as a primary cause, often stemming not from individual errors but from flawed interactions, e.g., failing to seek clarification when faced with ambiguity.

The central challenge is achieving **inter-agent agreement** under uncertainty. This requires extending single-agent metacognitive skills to the collective, enabling agents to model the uncertainty of their peers and adopt policies for uncertainty-aware communication. Robust UQ frameworks must therefore operate at two levels simultaneously: ensuring reliable local decisions for each agent while managing the propagation and aggregation of uncertainty across the system as a whole.

5 RL and Reward Modeling

In RL alignment, uncertainty has transformed from a factor ignored by deterministic scores into a core mechanism for robust learning: from building *robust reward models* to mitigate reward hacking, to enabling *self-improvement* via intrinsic rewards, and automating *scalable process supervision*.

5.1 Robust Reward Models

The cornerstone of the RLHF pipeline is the Reward Model (RM), which serves as a proxy for human values (Lambert et al., 2025). Conventional RMs are deterministic, producing a single scalar score. This creates a mismatch with the stochastic nature of human preferences and enables "reward hacking" (Fu et al., 2025a; Weng, 2024), where policies exploit RM inaccuracies to score highly on low-quality outputs (Lou et al., 2024; Cief et al., 2024). To address this, recent work has focused on RMs that can model and express uncertainty, broadly divided into two approaches.

Uncertainty-Aware Reward Models (URMs). This class of methods makes the RM explicitly aware of uncertainty, typically through architectural or feature-based modifications. A foundational approach is to redesign the RM's output to be probabilistic. The **URM** framework modifies the model's output head to predict a full probability

Core Concept	Strategy Function	Uncertainty Signal (The "What")	Control Mechanism (The "How")
Reward Models	URM (Lou et al., 2024)	Reward distribution variance	Penalty term in RL objective
	UALIGN (Xue et al., 2025)	Policy LLM's semantic entropy	Features for RM to learn
	Bayesian RMs (Yang et al., 2024)	Posterior distribution over RM weights	Theoretically-grounded penalty
Self-Improvement	RLSF (van Niekerk et al., 2025)	Model's confidence scores	Auto-generation of preference pairs
	Confidence Maximization (Prabhudesai et al., 2025)	Model's confidence score	intrinsic reward signal in RL.
	EM as Objective (Gao et al., 2025)	Entropy of the final predictive distribution	Unsupervised objective
	RL for EM (Zhang et al., 2025b)	Reduction in entropy	Entropy reduction as the reward signal.
Process Supervision	EDU-PRM (Cao et al., 2025)	High predictive entropy of tokens	Automatic partitioning of reasoning chains

Table 3: A comparative analysis of uncertainty-aware approaches in RL and Reward Modeling. It details how different frameworks leverage uncertainty signals to create more robust reward models, enable self-improvement, and scale supervision.

distribution (e.g., a Gaussian) instead of a single score (Lou et al., 2024). The variance of this distribution then serves as a direct, quantifiable signal of the *aleatoric uncertainty* (the intrinsic ambiguity in human data). A complementary strategy is to enrich the RM's input. The **UALIGN** framework achieves this by feeding the policy LLM's own uncertainty metrics (e.g., semantic entropy) *as explicit features* to the RM (Xue et al., 2025). This allows the RM to learn a context-aware evaluation function that is conditioned on the difficulty of the query as perceived by the policy model itself.

Bayesian Reward Models (Bayesian RMs). Instead of learning a single point estimate for the weights, **Bayesian RMs** learn a posterior distribution over them, thereby capturing *epistemic uncertainty* (the RM's own model uncertainty) (Yang et al., 2024). This is implemented using techniques like Laplace-LoRA (Schulman and Lab, 2025). The key advantage of this approach is that the uncertainty derived from the posterior can be used as a direct, theoretically-grounded penalty term during RL optimization. This actively discourages the policy from exploring and exploiting regions of the output space where the RM is unconfident, leading to safer and more robust alignment. A detailed comparative analysis is available in Table 3.

5.2 Self-Improvement RL

While robust reward models strengthen external supervision, a more advanced paradigm seeks to reduce dependence on such signals altogether. This paradigm is grounded in **intrinsic motivation**, where an agent improves by optimizing its own internal states rather than external feedback. Uncertainty expressed as confidence, entropy, or information gain (IG), has emerged as the core intrinsic reward for enabling self-driven alignment in LLMs.

Confidence as an Intrinsic Reward. The simplest intrinsic signal is self-confidence. The Reinforcement Learning from Self-Feedback (**RLSF**) framework demonstrates that confidence scores can generate synthetic preference pairs (e.g., high-confidence→low-confidence), enabling self-alignment without human labels (van Niekerk et al., 2025). Further studies show that directly maximizing confidence via RL significantly improves reasoning, confirming confidence as a standalone intrinsic reward (Prabhudesai et al., 2025). Yet, miscalibrated confidence can reinforce errors, and overconfidence may cause reward hacking.

Entropy Minimization (EM). A deeper perspective frames reasoning as a drive to reduce uncertainty. The principle of EM treats reasoning as minimizing the entropy of the predictive distribution, offering a reward-free, unsupervised objective for improving LLM reasoning (Agarwal et al., 2025). However, this approach is being actively refined, with the latest research exploring entropy not just as a quantity to be minimized, but as a regularization signal to achieve a better balance between confidence and accuracy (Jiang et al., 2025).

RL for EM. This information-theoretic signal can be optimized with RL, where entropy reduction itself becomes the reward. Frameworks such as **EMPO** incentivize reasoning trajectories that minimize future uncertainty (Zhang et al., 2025b; Cui et al., 2025). Architectures like **Intuitor** extend this to fully reward-free agents that learn policies from intrinsic motivations such as curiosity and uncertainty reduction (Zhao et al., 2025a).

Dissecting the Process with Mutual Information. Recent work leverages Mutual Information (MI) to analyze how EM operates. Crucially, the most informative "thinking tokens" in a chain of thought are those corresponding to peaks in MI with the final answer (Qian et al., 2025a). This provides a

mechanistic explanation of entropy minimization: reasoning progresses by identifying and resolving uncertainty at precisely these pivotal points.

5.3 Scalable Process Supervision

While intrinsic rewards enhance autonomy, alignment quality can be improved with fine-grained external feedback. **Process-based supervision** (Lightman et al., 2023), which rewards correct intermediate steps rather than only final outcomes, provides a stronger learning signal. However, its adoption has been limited by the high cost of manually segmenting reasoning chains into logical steps and annotating each one (Chen et al., 2024).

Uncertainty as Automation Tools. Recent work leverages uncertainty to automate this segmentation. The **EDU-PRM** framework (Cao et al., 2025) identifies tokens with high predictive entropy between reasoning steps, and uses them as "uncertainty anchors" to partition chains automatically. This enables scalable generation of process-level training data at a fraction of manual cost. Empirical results further suggest that RL gains are primarily driven by learning to handle these high-entropy minority tokens (Wang et al., 2025c). By transforming uncertainty into an automation tool, these methods make process-level supervision economically viable. The key *limitation* is heuristic reliability: high entropy is a strong but imperfect signal of logical boundaries. As a result, automated partitions may not always align with human-defined reasoning steps, creating a trade-off between scalability and annotation precision (Sun et al., 2024).

6 Emerging Theoretical Frameworks

The evolution from uncertainty as a passive metric to an active control signal is not merely a collection of empirical techniques; it reflects a deeper need for principled foundations to build reliable and trustworthy systems.

6.1 The Bayesian Method

As a foundational theory for reasoning under uncertainty, Bayesian methods are experiencing a resurgence, offering a principled basis for analyzing and guiding LLM behavior. A key theoretical insight is that while LLMs are not strictly Bayesian reasoners, their in-context learning mechanism often approximates Bayesian predictive updating in expectation (Chlon et al., 2025). This justifies applying Bayesian frameworks not to model the LLM internally, but to analyze its aggregate behavior and build more robust systems around it.

One pragmatic direction is **hybrid systems** that combine LLMs with formal probabilistic models. These exploit complementary strengths: qualitative, abductive reasoning from LLMs and quantitative uncertainty management from Bayesian inference. For example, **BIRD** uses LLMs to generate causal sketches that are formalized into Bayesian Networks for precise reasoning (Feng et al., 2025). **Textual Bayes** integrates more deeply, treating prompts as textual parameters for Bayesian inference (Ross et al., 2025), while other works use LLMs for prior elicitation (Selby et al., 2024).

Another ambitious line seeks to *teach* LLMs probabilistic reasoning directly, mitigating cognitive biases such as base-rate neglect (Smith et al., 2024). **Bayesian Teaching** fine-tunes models to mimic an ideal Bayesian observer, with evidence of generalization to unseen tasks (Qiu et al., 2025). This shift from using LLMs as Bayesian components to embedding Bayesian reasoning within them marks a step toward fundamentally improving their cognitive machinery (Yan et al., 2025b).

6.2 Conformal Prediction

In contrast to Bayesian methods that rely on prior distributions, Conformal Prediction (CP) offers a powerful non-Bayesian framework with rigorous, **distribution-free coverage guarantees** (Su et al., 2024; Wang et al., 2024). For any input, CP constructs a prediction set guaranteed to contain the true output with a user-specified probability, independent of model architecture or data distribution. Yet defining prediction sets and non-conformity scores for free-form text is non-trivial to apply CP to LLMs. Recent work addresses this by adapting CP to different levels of model access.

Black-Box (API-Only) Approaches. Without access to logits, methods like **ConU** (Wang et al., 2024) and Su et al. (2024) employ *semantic similarity* as a proxy for non-conformity.The prediction set includes a generated candidate along with semantically similar alternatives under a calibrated threshold. This reframes CP's guarantee from exact string matching to semantic equivalence, making it practical for open-ended generation.

White-Box (Logit-Access) Approaches. With full access to model probabilities, token-level calibration is possible. **Conformal Language Modeling** (Quach et al., 2023) uses logits to build predic-

tion sets for the next token at each step, ensuring that the true token lies within the set with high probability. This provides stronger guarantees but requires model transparency (Cherian et al., 2024).

The Theory–Practice Gap. Despite growing advances in theoretical frameworks, practitioners still face multiple open questions. To bridge this gap, we provide a set of design patterns and practical recommendations in Appendix Section C.

7 Challenges and Future Directions

While the evolving role of uncertainty is rapidly advancing, its full realization hinges on addressing several fundamental challenges.

Reliability and Robustness of the Active Signal. The function of uncertainty-as-a-control-signal is built upon the assumption that the signal itself is meaningful and trustworthy. Future work must rigorously address the integrity of this foundational layer. Even non-adversarial estimation errors can be amplified by downstream control mechanisms (Wilczyński et al., 2024). For example, a poorly calibrated confidence score can cause weighted voting to favor incorrect answers, while miscalibrated thresholds may lead agents to become recklessly overconfident or inefficiently tool-dependent.

Advancing UQ Benchmarking. The maturity of the field is evidenced by emerging standardized benchmarks, such as UBench (Wang et al., 2025d) and LM-Polygraph (Vashurin et al., 2025). While foundational, these frameworks predominantly assess *estimation fidelity*, diagnosing if a model knows it is wrong rather than *control utility*. They generally fail to simulate the dynamic decision-making trade-offs inherent to the active paradigm. Consequently, a critical misalignment exists between static evaluation protocols and dynamic control needs (Ye et al., 2024). Future benchmarks must evolve to quantify the downstream performance gains directly attributable to uncertainty-in-the-loop mechanisms.

Meaningful Evaluation and Metrics. Current evaluation remains a significant bottleneck. Standard metrics like AUROC are ill-suited for the rich, interactive, and dynamic contexts where the active-signal function is most relevant (Liu et al., 2025c). The field urgently requires new benchmarks and evaluation protocols specifically designed for interactive agents and complex reasoning tasks. Crucially, future evaluation must become more human-centered. The ultimate measure of success for an uncertainty-aware system is not just its statistical calibration, but its effectiveness as a partner in human-AI collaboration (Devic et al., 2025).

Composable, Uncertainty-Propagating Systems. Extending uncertainty management from single, monolithic models to complex, interconnected systems remains a major open problem. In MAS, the challenge is to understand how uncertainty propagates, compounds, and resolves across interacting agents, which requires new frameworks that operate at the system level rather than the individual agent level (Hu et al., 2025). More broadly, the ultimate trajectory points towards modular AI systems composed of heterogeneous components. A central challenge will be to establish a unified framework where uncertainty signals function as the "connective tissue" between these modules.

Scalability and Efficiency. A persistent challenge in this field is the trade-off between theoretical rigor and computational feasibility. Many of the most principled and powerful methods, particularly those grounded in Bayesian inference or requiring large-scale multi-agent simulations, are often too computationally expensive for widespread, real-time deployment. A critical direction for future work is therefore the development of scalable and efficient approximations of these formal methods.

8 Conclusion

This survey has charted an emerging technological trend: the evolution of uncertainty in LLMs from a passive, post-hoc diagnostic metric into an active, real-time control signal. We have traced this transformation across three frontiers: advanced reasoning, autonomous agents, and reinforcement learning, demonstrating how uncertainty is now being used not just to evaluate outputs, but to dynamically shape model behavior.

Limitations

First, our focus is on the *functional role* of uncertainty in advanced LLM systems, rather than a comprehensive review of uncertainty *estimation methods* or *confidence calibration*, which are covered by existing surveys. Second, this paper does not include large-scale comparative experiments; its main contribution is a conceptual framework and synthesis of existing work.

References

Shivam Agarwal, Zimin Zhang, Lifan Yuan, Jiawei Han, and Hao Peng. 2025. The unreasonable effectiveness of entropy minimization in llm reasoning. *arXiv preprint arXiv:2505.15134*.

Ohav Barbi, Ori Yoran, and Mor Geva. 2025. Preventing rogue agents improves multi-agent collaboration. *arXiv preprint arXiv:2502.05986*.

Mohammad Beigi, Sijia Wang, Ying Shen, Zihao Lin, Adithya Kulkarni, Jianfeng He, Feng Chen, Ming Jin, Jin-Hee Cho, Dawei Zhou, and 1 others. 2024. Rethinking the uncertainty: a critical review and analysis in the era of large language models. *arXiv preprint arXiv:2410.20199*.

Jan Betley, Xuchan Bao, Martín Soto, Anna Sztyber-Betley, James Chua, and Owain Evans. 2025. Tell me about yourself: Llms are aware of their learned behaviors. In *The Thirteenth International Conference on Learning Representations*.

Rishi Bommasani, Drew A. Hudson, Ehsan Adeli, Russ Altman, Simran Arora, Sydney von Arx, Michael S. Bernstein, Jeannette Bohg, Antoine Bosselut, Emma Brunskill, Erik Brynjolfsson, Shyamal Buch, Dallas Card, Rodrigo Castellon, Niladri Chatterji, Annie Chen, Kathleen Creel, Jared Quincy Davis, Dora Demszky, and 95 others. 2022. On the opportunities and risks of foundation models. *Preprint*, arXiv:2108.07258.

Lang Cao, Renhong Chen, Yingtian Zou, Chao Peng, Wu Ning, Huacong Xu, Qian Chen, Yuxian Wang, Peishuo Su, Mofan Peng, and 1 others. 2025. Process reward modeling with entropy-driven uncertainty. *arXiv preprint arXiv:2503.22233*.

Mert Cemri, Melissa Z Pan, Shuyi Yang, Lakshya A Agrawal, Bhavya Chopra, Rishabh Tiwari, Kurt Keutzer, Aditya Parameswaran, Dan Klein, Kannan Ramchandran, and 1 others. 2025. Why do multi-agent llm systems fail? *arXiv preprint arXiv:2503.13657*.

Guoxin Chen, Minpeng Liao, Chengxi Li, and Kai Fan. 2024. Alphamath almost zero: process supervision without process. *Advances in Neural Information Processing Systems*, 37:27689–27724.

John Cherian, Isaac Gibbs, and Emmanuel Candes. 2024. Large language model validity via enhanced conformal prediction methods. *Advances in Neural Information Processing Systems*, 37:114812–114842.

Leon Chlon, Sarah Rashidi, Zein Khamis, and MarcAntonio M Awada. 2025. Llms are bayesian, in expectation, not in realization. *arXiv preprint arXiv:2507.11768*.

Matej Cief, Francesco Tonolini, Nikolaos Aletras, and Gabriella Kazai. 2024. Adaptive uncertainty-aware reinforcement learning from human feedback.

Ganqu Cui, Yuchen Zhang, Jiacheng Chen, Lifan Yuan, Zhi Wang, Yuxin Zuo, Haozhan Li, Yuchen Fan, Huayu Chen, Weize Chen, and 1 others. 2025. The entropy mechanism of reinforcement learning for reasoning language models. *arXiv preprint arXiv:2505.22617*.

Wendi Cui, Zhuohang Li, Damien Lopez, Kamalika Das, Bradley A Malin, Sricharan Kumar, and Jiaxin Zhang. 2024. Divide-conquer-reasoning for consistency evaluation and automatic improvement of large language models. In *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing: Industry Track*, pages 334–361.

Longchao Da, Xiaoou Liu, Jiaxin Dai, Lu Cheng, Yaqing Wang, and Hua Wei. 2025. Understanding the uncertainty of llm explanations: A perspective based on reasoning topology. *arXiv preprint arXiv:2502.17026*.

Siddartha Devic, Tejas Srinivasan, Jesse Thomason, Willie Neiswanger, and Vatsal Sharan. 2025. From calibration to collaboration: Llm uncertainty quantification should be more human-centered. *arXiv preprint arXiv:2506.07461*.

Jinhao Duan, James Diffenderfer, Sandeep Madireddy, Tianlong Chen, Bhavya Kailkhura, and Kaidi Xu. 2025. Uprop: Investigating the uncertainty propagation of llms in multi-step agentic decision-making. *arXiv preprint arXiv:2506.17419*.

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

Yu Feng, Ben Zhou, Weidong Lin, and Dan Roth. 2025. Bird: A trustworthy bayesian inference framework for large language models. In *The Thirteenth International Conference on Learning Representations*.

Jiayi Fu, Xuandong Zhao, Chengyuan Yao, Heng Wang, Qi Han, and Yanghua Xiao. 2025a. Reward shaping to mitigate reward hacking in rlhf. *arXiv preprint arXiv:2502.18770*.

Yichao Fu, Xuewei Wang, Yuandong Tian, and Jiawei Zhao. 2025b. Deep think with confidence. *arXiv preprint arXiv:2508.15260*.

Zeyu Gan, Yun Liao, and Yong Liu. 2025. Rethinking external slow-thinking: From snowball errors to probability of correct reasoning. *arXiv preprint arXiv:2501.15602*.

Xiang Gao, Jiaxin Zhang, Lalla Mouatadid, and Kamalika Das. 2024. Spuq: Perturbation-based uncertainty quantification for large language models. In *Proceedings of the 18th Conference of the European Chapter of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 2336–2346.

Zitian Gao, Lynx Chen, Haoming Luo, Joey Zhou, and Bryan Dai. 2025. One-shot entropy minimization. *arXiv preprint arXiv:2505.20282*.

Jakob Gawlikowski, Cedrique Rovile Njieutcheu Tassi, Mohsin Ali, Jongseok Lee, Matthias Humt, Jianxiang Feng, Anna Kruspe, Rudolph Triebel, Peter Jung, Ribana Roscher, and 1 others. 2023. A survey of uncertainty in deep neural networks. *Artificial Intelligence Review*, 56(Suppl 1):1513–1589.

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.

Jiuzhou Han, Wray Buntine, and Ehsan Shareghi. 2024. Towards uncertainty-aware language agent. In *Findings of the Association for Computational Linguistics ACL 2024*, pages 6662–6685.

Somnath Hazra, Pallab Dasgupta, and Soumyajit Dey. 2025. Tackling uncertainties in multi-agent reinforcement learning through integration of agent termination dynamics. In *Proceedings of the 24th International Conference on Autonomous Agents and Multiagent Systems*, pages 960–968.

Jinwei Hu, Yi Dong, Shuang Ao, Zhuoyun Li, Boxuan Wang, Lokesh Singh, Guangliang Cheng, Sarvapali D Ramchurn, and Xiaowei Huang. 2025. Position: Towards a responsible llm-empowered multi-agent systems. *arXiv preprint arXiv:2502.01714*.

Zhiyuan Hu, Chumin Liu, Xidong Feng, Yilun Zhao, See-Kiong Ng, Anh Tuan Luu, Junxian He, Pang Wei Koh, and Bryan Hooi. 2024. Uncertainty of thoughts: Uncertainty-aware planning enhances information seeking in large language models. *arXiv preprint arXiv:2402.03271*.

Hsiu-Yuan Huang, Yutong Yang, Zhaoxi Zhang, Sanwoo Lee, and Yunfang Wu. 2024. A survey of uncertainty estimation in llms: Theory meets practice. *arXiv preprint arXiv:2410.15326*.

Yuxian Jiang, Yafu Li, Guanxu Chen, Dongrui Liu, Yu Cheng, and Jing Shao. 2025. Rethinking entropy regularization in large reasoning models. *arXiv preprint arXiv:2509.25133*.

Saurav Kadavath, Tom Conerly, Amanda Askell, Tom Henighan, Dawn Drain, Ethan Perez, Nicholas Schiefer, Zac Hatfield-Dodds, Nova DasSarma, Eli Tran-Johnson, and 1 others. 2022. Language models (mostly) know what they know. *arXiv preprint arXiv:2207.05221*.

Ryo Kamoi, Yusen Zhang, Nan Zhang, Jiawei Han, and Rui Zhang. 2024. When can llms actually correct their own mistakes? a critical survey of self-correction of llms. *Transactions of the Association for Computational Linguistics*, 12:1417–1440.

Alex Kendall and Yarin Gal. 2017. What uncertainties do we need in bayesian deep learning for computer vision? *Advances in neural information processing systems*, 30.

Michael Kirchhof, Gjergji Kasneci, and Enkelejda Kasneci. 2025. Position: Uncertainty quantification needs reassessment for large language model agents. In *Forty-second International Conference on Machine Learning Position Paper Track*.

Ranganath Krishnan, Piyush Khanna, and Omesh Tickoo. 2024. Enhancing trust in large language models with uncertainty-aware fine-tuning. *arXiv preprint arXiv:2412.02904*.

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

Nathan Lambert, Valentina Pyatkin, Jacob Morrison, Lester James Validad Miranda, Bill Yuchen Lin, Khyathi Chandu, Nouha Dziri, Sachin Kumar, Tom Zick, Yejin Choi, and 1 others. 2025. Rewardbench: Evaluating reward models for language modeling. In *Findings of the Association for Computational Linguistics: NAACL 2025*, pages 1755–1797.

Jiaqi Li, Yixuan Tang, and Yi Yang. 2025a. Know the unknown: An uncertainty-sensitive method for llm instruction tuning. In *Findings of the Association for Computational Linguistics: ACL 2025*, pages 2972–2989.

Lei Li, Hehuan Liu, Yaxin Zhou, ZhaoYang Gui, Xudong Weng, Yi Yuan, Zheng Wei, and Zang Li. 2025b. Uncertainty-aware iterative preference optimization for enhanced llm reasoning. In *Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 23996–24012.

Yafu Li, Xuyang Hu, Xiaoye Qu, Linjie Li, and Yu Cheng. 2025c. Test-time preference optimization: On-the-fly alignment via iterative textual feedback. In *Forty-second International Conference on Machine Learning*.

Hunter Lightman, Vineet Kosaraju, Yuri Burda, Harrison Edwards, Bowen Baker, Teddy Lee, Jan Leike, John Schulman, Ilya Sutskever, and Karl Cobbe. 2023. Let's verify step by step. In *The Twelfth International Conference on Learning Representations*.

Hao Liu, Zi-Yi Dou, Yixin Wang, Nanyun Peng, and Yisong Yue. 2024. Uncertainty calibration for tool-using language agents. In *Findings of the Association for Computational Linguistics: EMNLP 2024*, pages 16781–16805.

Jingyu Liu, JingquanPeng, Xiaopeng Wu, Xubin Li, Tiezheng Ge, Bo Zheng, and Yong Liu. 2025a. Do not abstain! identify and solve the uncertainty. In *Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 17177–17197.

Runze Liu, Junqi Gao, Jian Zhao, Kaiyan Zhang, Xiu Li, Biqing Qi, Wanli Ouyang, and Bowen Zhou. 2025b. Can 1b llm surpass 405b llm? rethinking compute-optimal test-time scaling. *arXiv preprint arXiv:2502.06703*.

Xiaoou Liu, Tiejin Chen, Longchao Da, Chacha Chen, Zhen Lin, and Hua Wei. 2025c. Uncertainty quantification and confidence calibration in large language models: A survey. In *Proceedings of the 31st ACM SIGKDD Conference on Knowledge Discovery and Data Mining V. 2*, pages 6107–6117.

Yuliang Liu, Junjie Lu, Zhaoling Chen, Chaofeng Qu, Jason Klein Liu, Chonghan Liu, Zefan Cai, Yunhui Xia, Li Zhao, Jiang Bian, and 1 others. 2025d. Adaptivestep: Automatically dividing reasoning step through model confidence. *arXiv preprint arXiv:2502.13943*.

Xingzhou Lou, Dong Yan, Wei Shen, Yuzi Yan, Jian Xie, and Junge Zhang. 2024. Uncertainty-aware reward model: Teaching reward models to know what is unknown. *arXiv preprint arXiv:2410.00847*.

Niklas Muennighoff, Zitong Yang, Weijia Shi, Xiang Lisa Li, Li Fei-Fei, Hannaneh Hajishirzi, Luke Zettlemoyer, Percy Liang, Emmanuel Candès, and Tatsunori B Hashimoto. 2025. s1: Simple test-time scaling. In *Proceedings of the 2025 Conference on Empirical Methods in Natural Language Processing*, pages 20286–20332.

Jianfeng Pan, Senyou Deng, and Shaomang Huang. 2025. Coat: Chain-of-associated-thoughts framework for enhancing large language models reasoning. *arXiv preprint arXiv:2502.02390*.

Mihir Prabhudesai, Lili Chen, Alex Ippoliti, Katerina Fragkiadaki, Hao Liu, and Deepak Pathak. 2025. Maximizing confidence alone improves reasoning. *arXiv preprint arXiv:2505.22660*.

Xiao Pu, Michael Saxon, Wenyue Hua, and William Yang Wang. 2025. Thoughtterminator: Benchmarking, calibrating, and mitigating overthinking in reasoning models. *arXiv preprint arXiv:2504.13367*.

Chen Qian, Dongrui Liu, Haochen Wen, Zhen Bai, Yong Liu, and Jing Shao. 2025a. Demystifying reasoning dynamics with mutual information: Thinking tokens are information peaks in llm reasoning. *arXiv preprint arXiv:2506.02867*.

Cheng Qian, Emre Can Acikgoz, Hongru Wang, Xiusi Chen, Avirup Sil, Dilek Hakkani-Tur, Gokhan Tur, and Heng Ji. 2025b. Smart: Self-aware agent for tool overuse mitigation. In *Findings of the Association for Computational Linguistics: ACL 2025*, pages 4604–4621.

Linlu Qiu, Fei Sha, Kelsey Allen, Yoon Kim, Tal Linzen, and Sjoerd van Steenkiste. 2025. Bayesian teaching enables probabilistic reasoning in large language models. *arXiv preprint arXiv:2503.17523*.

Victor Quach, Adam Fisch, Tal Schuster, Adam Yala, Jae Ho Sohn, Tommi S Jaakkola, and Regina Barzilay. 2023. Conformal language modeling. In *The Twelfth International Conference on Learning Representations*.

Ali Razghandi, Seyed Mohammad Hadi Hosseini, and Mahdieh Soleymani Baghshah. 2025. Cer: Confidence enhanced reasoning in llms. *arXiv preprint arXiv:2502.14634*.

Brendan Leigh Ross, NoÃĢl Vouitsis, Atiyeh Ashari Ghomi, Rasa Hosseinzadeh, Ji Xin, Zhaoyan Liu, Yi Sui, Shiyi Hou, Kin Kwan Leung, Gabriel Loaiza-Ganem, and 1 others. 2025. Textual bayes: Quantifying uncertainty in llm-based systems. *arXiv preprint arXiv:2506.10060*.

John Schulman and Thinking Machines Lab. 2025. Lora without regret. *Thinking Machines Lab: Connectionism*. Https://thinkingmachines.ai/blog/lora/.

David Antony Selby, Kai Spriestersbach, Yuichiro Iwashita, Dennis Bappert, Archana Warrier, Sumantrak Mukherjee, Muhammad Nabeel Asim, Koichi Kise, and Sebastian Josef Vollmer. 2024. Had enough of experts? elicitation and evaluation of bayesian priors from large language models. In *NeurIPS 2024 Workshop on Bayesian Decision-making and Uncertainty*.

Ola Shorinwa, Zhiting Mei, Justin Lidard, Allen Z Ren, and Anirudha Majumdar. 2025. A survey on uncertainty quantification of large language models: Taxonomy, open research challenges, and future directions. *ACM Computing Surveys*.

Ryan Smith, Jason A Fries, Braden Hancock, and Stephen H Bach. 2024. Language models in the loop: Incorporating prompting into weak supervision. *ACM/JMS Journal of Data Science*, 1(2):1–30.

Josefa Lia Stoisser, Marc Boubnovski Martell, Lawrence Phillips, Gianluca Mazzoni, Lea Mørch Harder, Philip Torr, Jesper Ferkinghoff-Borg, Kaspar Martens, and Julien Fauqueur. 2025. Towards agents that know when they don't know: Uncertainty as a control signal for structured reasoning. *arXiv preprint arXiv:2509.02401*.

Jiayuan Su, Jing Luo, Hongwei Wang, and Lu Cheng. 2024. Api is enough: Conformal prediction for large language models without logit-access. In *Findings of the Association for Computational Linguistics: EMNLP 2024*, pages 979–995.

Zhiqing Sun, Longhui Yu, Yikang Shen, Weiyang Liu, Yiming Yang, Sean Welleck, and Chuang Gan. 2024. Easy-to-hard generalization: Scalable alignment beyond human supervision. *Advances in Neural Information Processing Systems*, 37:51118–51168.

Amir Taubenfeld, Tom Sheffer, Eran Ofek, Amir Feder, Ariel Goldstein, Zorik Gekhman, and Gal Yona. 2025. Confidence improves self-consistency in llms. *arXiv preprint arXiv:2502.06233*.

Katherine Tian, Eric Mitchell, Allan Zhou, Archit Sharma, Rafael Rafailov, Huaxiu Yao, Chelsea Finn, and Christopher D Manning. 2023. Just ask for calibration: Strategies for eliciting calibrated confidence scores from language models fine-tuned with human feedback. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pages 5433–5442.

Carel van Niekerk, Renato Vukovic, Benjamin Matthias Ruppik, Hsien-chin Lin, and Milica Gašić. 2025. Post-training large language models via reinforcement learning from self-feedback. *arXiv preprint arXiv:2507.21931*.

Roman Vashurin, Ekaterina Fadeeva, Artem Vazhentsev, Lyudmila Rvanova, Daniil Vasilev, Akim Tsvigun, Sergey Petrakov, Rui Xing, Abdelrahman Sadallah, Kirill Grishchenkov, and 1 others. 2025. Benchmarking uncertainty quantification methods for large language models with lm-polygraph. *Transactions of the Association for Computational Linguistics*, 13:220–248.

Jiawei Wang, Jiacai Liu, Yuqian Fu, Yingru Li, Xintao Wang, Yuan Lin, Yu Yue, Lin Zhang, Yang Wang, and Ke Wang. 2025a. Harnessing uncertainty: Entropy-modulated policy gradients for long-horizon llm agents. *arXiv preprint arXiv:2509.09265*.

Jingyao Wang, Wenwen Qiang, Zeen Song, Changwen Zheng, and Hui Xiong. 2025b. Learning to think: Information-theoretic reinforcement fine-tuning for llms. *arXiv preprint arXiv:2505.10425*.

Shenzhi Wang, Le Yu, Chang Gao, Chujie Zheng, Shixuan Liu, Rui Lu, Kai Dang, Xionghui Chen, Jianxin Yang, Zhenru Zhang, and 1 others. 2025c. Beyond the 80/20 rule: High-entropy minority tokens drive effective reinforcement learning for llm reasoning. *arXiv preprint arXiv:2506.01939*.

Tao Wang, Daniel Lizotte, Michael Bowling, and Dale Schuurmans. 2005. Bayesian sparse sampling for on-line reward optimization. In *Proceedings of the 22nd international conference on Machine learning*, pages 956–963.

Xunzhi Wang, Zhuowei Zhang, Gaonan Chen, Qiongyu Li, Bitong Luo, Zhixin Han, Haotian Wang, Zhiyu Li, Hang Gao, and Mengting Hu. 2025d. Ubench: Benchmarking uncertainty in large language models with multiple choice questions. In *Findings of the Association for Computational Linguistics: ACL 2025*, pages 8076–8107.

Zhihai Wang, Jie Wang, Jilai Pan, Xilin Xia, Huiling Zhen, Mingxuan Yuan, Jianye Hao, and Feng Wu. 2025e. Accelerating large language model reasoning via speculative search. *arXiv preprint arXiv:2505.02865*.

Zhiyuan Wang, Jinhao Duan, Lu Cheng, Yue Zhang, Qingni Wang, Xiaoshuang Shi, Kaidi Xu, Heng Tao Shen, and Xiaofeng Zhu. 2024. Conu: Conformal uncertainty in large language models with correctness coverage guarantees. In *Findings of the Association for Computational Linguistics: EMNLP 2024*, pages 6886–6898.

Lilian Weng. 2024. Reward hacking in reinforcement learning. *lilianweng.github.io*.

Piotr Wilczyński, Wiktoria Mieleszczenko-Kowszewicz, and Przemysław Biecek. 2024. Resistance against manipulative ai: key factors and possible actions. In *ECAI 2024*, pages 802–809. IOS Press.

Heming Xia, Chak Tou Leong, Wenjie Wang, Yongqi Li, and Wenjie Li. 2025a. Tokenskip: Controllable chain-of-thought compression in llms. *arXiv preprint arXiv:2502.12067*.

Zhiqiu Xia, Jinxuan Xu, Yuqian Zhang, and Hang Liu. 2025b. A survey of uncertainty estimation methods on large language models. *arXiv preprint arXiv:2503.00172*.

Miao Xiong, Zhiyuan Hu, Xinyang Lu, YIFEI LI, Jie Fu, Junxian He, and Bryan Hooi. 2024. Can llms express their uncertainty? an empirical evaluation of confidence elicitation in llms. In *The Twelfth International Conference on Learning Representations*.

Boyang Xue, Fei Mi, Qi Zhu, Hongru Wang, Rui Wang, Sheng Wang, Erxin Yu, Xuming Hu, and Kam-Fai Wong. 2025. Ualign: Leveraging uncertainty estimations for factuality alignment on large language models. In *Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 6002–6024.

Hang Yan, Fangzhi Xu, Rongman Xu, Yifei Li, Jian Zhang, Haoran Luo, Xiaobao Wu, Luu Anh Tuan, Haiteng Zhao, Qika Lin, and 1 others. 2025a. Mur: Momentum uncertainty guided reasoning for large language models. *arXiv preprint arXiv:2507.14958*.

Hanqi Yan, Linhai Zhang, Jiazheng Li, Zhenyi Shen, and Yulan He. 2025b. Position: Llms need a bayesian meta-reasoning framework for more robust and generalizable reasoning. In *2025 International Conference on Machine Learning: ICML25*.

Adam X Yang, Maxime Robeyns, Thomas Coste, Zhengyan Shi, Jun Wang, Haitham Bou-Ammar, and Laurence Aitchison. 2024. Bayesian reward models for llm alignment. *arXiv preprint arXiv:2402.13210*.

Shunyu Yao, Jeffrey Zhao, Dian Yu, Nan Du, Izhak Shafran, Karthik R Narasimhan, and Yuan Cao. 2022. React: Synergizing reasoning and acting in language models. In *The eleventh international conference on learning representations*.

Fanghua Ye, Mingming Yang, Jianhui Pang, Longyue Wang, Derek Wong, Emine Yilmaz, Shuming Shi, and Zhaopeng Tu. 2024. Benchmarking llms via uncertainty quantification. *Advances in Neural Information Processing Systems*, 37:15356–15385.

Zhangyue Yin, Qiushi Sun, Qipeng Guo, Jiawen Wu, Xipeng Qiu, and Xuan-Jing Huang. 2023. Do large language models know what they don't know? In *Findings of the Association for Computational Linguistics: ACL 2023*, pages 8653–8665.

Zhangyue Yin, Qiushi Sun, Qipeng Guo, Zhiyuan Zeng, Xiaonan Li, Junqi Dai, Qinyuan Cheng, Xuan-Jing Huang, and Xipeng Qiu. 2024. Reasoning in flux: Enhancing large language models reasoning through uncertainty-aware adaptive guidance. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 2401–2416.

Jiaxin Zhang. 2021. Modern monte carlo methods for efficient uncertainty quantification and propagation: A survey. *Wiley Interdisciplinary Reviews: Computational Statistics*, 13(5):e1539.

Jiaxin Zhang. 2025. Confidence-aware reasoning: Optimizing self-guided thinking trajectories in large reasoning models. In *Proceedings of the 2025 Conference on Empirical Methods in Natural Language Processing: Industry Track*, pages 2081–2095.

Jiaxin Zhang, Prafulla Kumar Choubey, Kung-Hsiang Huang, Caiming Xiong, and Chien-Sheng Wu. 2026a. Agentic uncertainty quantification. *arXiv preprint arXiv:2601.15703*.

Jiaxin Zhang, Zhuohang Li, Kamalika Das, Bradley Malin, and Sricharan Kumar. 2023. Sac3: reliable hallucination detection in black-box language models via semantic-aware cross-check consistency. In *Findings of the Association for Computational Linguistics: EMNLP 2023*, pages 15445–15458.

Jiaxin Zhang, Caiming Xiong, and Chien-Sheng Wu. 2026b. Agentic confidence calibration. *arXiv preprint arXiv:2601.15778*.

Jiaxin Zhang, Caiming Xiong, and Jason Wu. 2025a. Agentic confidence calibration. *preprint*.

Qingyang Zhang, Haitao Wu, Changqing Zhang, Peilin Zhao, and Yatao Bian. 2025b. Right question is already half the answer: Fully unsupervised llm reasoning incentivization. *arXiv preprint arXiv:2504.05812*.

Qiwei Zhao, Xujiang Zhao, Yanchi Liu, Wei Cheng, Yiyou Sun, Mika Oishi, Takao Osaki, Katsushi Matsuda, Huaxiu Yao, and Haifeng Chen. 2024. Saup: Situation awareness uncertainty propagation on llm agent. *arXiv preprint arXiv:2412.01033*.

Xuandong Zhao, Zhewei Kang, Aosong Feng, Sergey Levine, and Dawn Song. 2025a. Learning to reason without external rewards. *arXiv preprint arXiv:2505.19590*.

Xutong Zhao, Tengyu Xu, Xuewei Wang, Zhengxing Chen, Di Jin, Liang Tan, Zishun Yu, Zhuokai Zhao, Yun He, Sinong Wang, and 1 others. 2025b. Boosting llm reasoning via spontaneous self-correction. *arXiv preprint arXiv:2506.06923*.

Han Zhong, Yutong Yin, Shenao Zhang, Xiaojun Xu, Yuanxin Liu, Yifei Zuo, Zhihan Liu, Boyi Liu, Sirui Zheng, Hongyi Guo, and 1 others. 2025. Brite: Bootstrapping reinforced thinking process to enhance language model reasoning. *arXiv preprint arXiv:2501.18858*.

A Comparative Analysis of Different Functions

Throughout this survey, we utilize a series of tables and figures to provide both a conceptual and a literature-based overview of the "uncertainty-as-a-control-signal" function. Tables 2, 1 and 3 offer a comparative analysis of key methodologies within advanced reasoning, autonomous agents, and RL/reward modeling, respectively. Each table is structured to highlight the core components of the active-signal framework: the specific uncertainty signal being used (the "what") and the control mechanism through which it acts (the "how").

To complement this analysis, Figures 2, 3, and 4 provide a comprehensive visual breakdown of the literature cited in each of the main application sections (§3, §4, and §5). These figures serve as a quick reference map, categorizing the key papers discussed and linking them to the specific sub-topics they address, thereby offering a detailed landscape of the foundational and recent work in each domain.

B Critical Analysis

To complement the comparative analysis, this section provides a detailed critical analysis of the key uncertainty-aware methods discussed in Sections 3, 4, and 5. The goal is to move beyond mere description and offer a practical perspective on the trade-offs involved in deploying these techniques. Tables 4, 5, and 6 serve as the core of this analysis, evaluating each method across these key dimensions:

- **Key Advantage(s):** The primary strengths and benefits of the approach.
- **Key Disadvantage(s) / Failure Mode(s):** The main weaknesses, limitations, or common ways the method can fail in practice.
- **Computational Cost:** The relative resource requirements during inference.
- **Implementation Complexity:** The relative difficulty of integrating the method into a standard LLM workflow.

While the tables provide a high-level summary, the ratings for "Computational Cost" and "Implementation Complexity" (e.g., Low, Medium, High) are subjective and context-dependent. The following subsections are therefore dedicated to **justifying these ratings in detail**, offering a clear rationale for why each method was classified as it was based on its specific operational and engineering requirements.

B.1 Advanced Reasoning

The ratings provided in Table 4 for "Computational Cost" and "Implementation Complexity" are justified as follows, offering a more detailed rationale for each classification.

- **CISC & CER:** These methods are rated "High" to "Very High" in computational cost because their core mechanism relies on sampling multiple complete reasoning paths from the LLM, which is inherently expensive and multiplies inference latency. **CER** is rated slightly higher as it adds an extra layer of evaluation on intermediate steps. In contrast, **CISC**'s implementation complexity is "Low" as it only requires a simple scoring and voting logic on the final outputs. **CER**'s complexity is "Medium" because it necessitates building a more sophisticated system to identify and evaluate pre-defined "critical" steps within a reasoning chain.
- **UAG / SPOC:** These methods incur a "Medium" computational cost as they operate within a single reasoning path but add verification overhead at each step, increasing the total number of tokens generated and processed. Their implementation complexity is "High" because developing a reliable self-correction or verification mechanism is a significant challenge, often requiring complex prompting strategies or fine-tuning a separate verifier model.
- **Uncertainty-Aware FT:** The key distinction here is between training and inference. The implementation complexity is "High" because it requires modifying the core training process, often by designing and implementing a custom loss function. However, once the model is trained, the inference cost is "Low" as the uncertainty-awareness is baked into the model's weights and does not add any extra steps or overhead at runtime.
- **UnCert-CoT:** This method is rated "Low" on both metrics, making it highly practical. The computational cost is minimal, adding only a

Method / Framework	Key Advantage(s)	Key Disadvantage(s) / Failure Mode(s)	Cost	Complexity
Between Reasoning Paths				
CISC	- More efficient than standard self-consistency.	- A single bad step can sink a good path score. - Relies on well-calibrated confidence.	High	Low
CER	- Robust for long-chain reasoning. - Focuses on the most important steps.	- Must correctly identify "critical" steps. - Can amplify errors from miscalibrated confidence.	Very High	Medium
Inside a Reasoning Path				
UAG / SPOC	- Enables real-time error correction. - No retraining required.	- LLMs often fail at true self-correction. - Can get stuck in correction loops.	Medium	High
Uncertainty-Aware FT	- Fundamentally improves model calibration. - Benefits all downstream tasks.	- Data-intensive training process. - Risk of harming in-distribution performance.	Low	High
Optimizing Cognitive Effort				
UnCert-CoT	- Excellent efficiency-performance balance. - Simple and intuitive concept.	- Performance is highly sensitive to the threshold value.	Low	Low
MUR	- More stable control via momentum. - Finer-grained resource allocation.	- More complex than simple triggers. - Adds more hyperparameters to tune.	Low-Medium	Medium

Table 4: Critical Analysis of Methods in Advanced Reasoning. This table provides a comparative overview of key methodologies, focusing on their advantages, failure modes, computational costs, and implementation complexity.

lightweight entropy or probability check during generation. Its implementation complexity is also low, as it can often be realized with a simple wrapper that applies conditional logic ("if uncertainty > threshold, then use CoT"). The main challenge lies in calibration, not complex engineering.

- **MUR:** This framework is rated "Low-Medium" for cost and "Medium" for complexity. The cost is variable; it is designed to be efficient but can dynamically allocate more computational resources (like Test-Time Scaling) to uncertain steps, making it potentially more expensive than a single, standard forward pass. Its implementation complexity is "Medium" because it requires building a stateful tracking system to maintain the "momentum" of uncertainty across multiple generation steps, which is more involved than a stateless threshold check.

B.2 Autonomous Agents

The ratings in Table 5 are justified by the specific operational and engineering requirements of each method:

- **Abstention:** This method earns a "Low" rating for both cost and complexity. Computationally, it only requires a lightweight calculation (e.g., entropy) on the final generated output. In terms of implementation, it is a simple post-processing step, effectively an "if/else" check before returning a response.

- **Proactive Inquiry (UoT):** Its "High" complexity stems from the need to implement a full reinforcement learning loop, which involves defining state spaces, action policies, and complex reward functions like Expected Information Gain. The "Medium-High" computational cost reflects the intensive offline training and the potential for multiple model calls during inference to evaluate and select the best clarifying question.

- **UALA:** This framework is rated "Low" for both cost and complexity because it is designed for efficiency. It adds only a single uncertainty calculation to the workflow, which is computationally cheap. Its implementation is a straightforward threshold-based rule, making it one of the simplest methods to deploy.

- **SMARTAgent:** The complexity is "High" due to the significant upfront engineering effort required to design, create, and curate a specialized dataset for fine-tuning the agent on its knowledge boundaries. While the inference cost is "Low" (as the decision logic is compiled into the model's weights), the initial training and data collection cost is substantial.

- **SAUP:** It receives a "Medium" rating for both cost and complexity. The cost is not fixed but scales linearly with the number of steps in an agent's trajectory, as it adds a calculation at each turn. The implementation requires building a state-tracking system that persists across multiple turns and defining the logic for the heuristic "situational weights," which is more involved than a simple wrapper.

- **UProp:** This framework is rated "High" on both metrics due to its theoretical depth. The computational cost is significant, as it requires estimating mutual information, a notoriously challenging task that often relies on expensive sampling-

based methods. The implementation complexity is also high, demanding a strong grasp of information theory and the development of sophisticated estimators.

B.3 RL and Reward Modeling

The ratings assigned in Table 6 are based on the specific requirements for training and implementing each RL and reward modeling method.

- **URM (Uncertainty-Aware RM):** Its implementation complexity is "Medium" because it requires modifying the reward model's architecture (e.g., changing the output head to predict a distribution) and adapting the training pipeline, often to use a Maximum Likelihood Estimation loss instead of a standard preference loss. The inference cost remains "Low" as it is still a single forward pass.
- **Bayesian RMs:** This approach is rated "High" for complexity as it demands specialized knowledge of Bayesian deep learning techniques (e.g., variational inference, Laplace-LoRA) to implement correctly. The computational cost is "Medium-High" because training is often more intensive, and inference can be slower if it requires sampling from the posterior distribution to estimate uncertainty.
- **RLSF (RL from Self-Feedback):** The complexity is "Medium" as it involves a multi-stage pipeline: generating responses, scoring them with the model's own confidence, creating a synthetic preference dataset, and then running a standard RL algorithm. The computational cost is also "Medium," reflecting the overhead of this multi-step data creation process before the main RL training begins.
- **Confidence / Entropy Maximization:** These self-improvement methods are rated "Low" on both metrics. They are among the easiest to implement, as they only require calculating a simple, readily available metric (confidence or entropy) and using it directly as an intrinsic reward signal within a standard RL loop. The computational overhead per training step is negligible.
- **EDU-PRM:** This method's primary function is in the data preparation stage. Its implementation complexity is "Medium" because it requires building a custom data processing pipeline to automatically segment reasoning chains based on entropy signals. The computational cost is considered "Low" as this is an efficient, one-time offline process performed before training begins.

C A Practitioner's Guide to Designing Uncertainty-Aware Systems

This appendix provides a set of design patterns and practical recommendations for developers and researchers aiming to integrate the "uncertainty-as-a-control-signal" function into real-world LLM applications.

C.1 Advanced Reasoning

The choice of strategy for enhancing model reasoning depends on task complexity, accuracy requirements, and computational budget.

Scenario 1: High-stakes, complex tasks requiring maximum accuracy (e.g., math competitions, scientific QA).

- **Recommended Pattern:** Confidence-Weighted Ensembling.
- **Methods:** Prefer fine-grained approaches like **CER** (Razghandi et al., 2025), which focus on the confidence of critical reasoning steps, over simpler majority voting (**Self-Consistency**) or whole-path scoring (**CISC** (Taubenfeld et al., 2025)).
- **Practical Advice:**
  - **Cost:** Be aware of the high computational cost, especially for generating multiple reasoning paths. Use this for offline evaluation or latency-insensitive tasks.
  - **Calibration:** The success of weighted voting hinges on the quality of confidence scores. Investing in calibrating the model's confidence is crucial; otherwise, an overconfident model might assign high weights to wrong answers.

Scenario 2: Tasks with variable difficulty requiring a balance of efficiency and performance (e.g., code generation, general-purpose chatbots).

- **Recommended Pattern:** Uncertainty-Triggered Dynamic Allocation.
- **Methods: UnCert-CoT** (Li et al., 2025b) or **MUR** (Yan et al., 2025a) are ideal. They activate more

Method / Framework	Key Advantage(s)	Key Disadvantage(s) / Failure Mode(s)	Cost	Complexity
Function: Responding to Internal Uncertainty				
Abstention	- Simple, robust safety mechanism. - Prevents generating harmful misinformation.	- Can be overly conservative, reducing helpfulness. - Performance is highly sensitive to the threshold.	Low	Low
Proactive Inquiry (UoT)	- Actively reduces uncertainty, improving final quality. - Mimics intelligent, collaborative behavior.	- Can increase user burden with too many questions. - Requires a complex (often RL-trained) policy.	Medium-High	High
Function: Tool-Use Decision Boundary				
UALA	- Greatly improves efficiency vs. always-use-tool. - Simple threshold-based logic.	- Does not account for tool unreliability (blind trust). - Static threshold may not generalize well.	Low	Low
SMARTAgent	- Internalizes knowledge boundaries via training. - More robust than a simple static threshold.	- Requires creating a specialized fine-tuning dataset. - Higher upfront training cost.	Low (inference)	High
Function: Uncertainty Propagation				
SAUP	- Pragmatic and intuitive approach. - Context-aware weighting is powerful.	- Situational weights can be heuristic and hard to define formally across different tasks.	Medium	Medium
UProp	- Principled, information-theoretic foundation. - Clearly separates intrinsic vs. extrinsic uncertainty.	- Computationally expensive to estimate mutual info. - Can be less practical for real-time applications.	High	High

Table 5: Critical Analysis of Methods in Autonomous Agents. This table provides a comparative overview of key methodologies, focusing on their advantages, failure modes, computational costs, and implementation complexity.

Method / Framework	Key Advantage(s)	Key Disadvantage(s) / Failure Mode(s)	Cost	Complexity
Function: Robust Reward Models				
URM	- Explicitly models data ambiguity (aleatoric uncertainty). - Simple architectural change.	- May not capture model's own ignorance (epistemic). - Requires changing the training objective.	Low (inference)	Medium
Bayesian RMs	- Principled way to capture model uncertainty (epistemic). - Provides a theoretically-grounded penalty for RL.	- Can be computationally expensive to train and run. - More complex to implement correctly.	Medium-High	High
Function: Self-Improvement RL (Intrinsic Rewards)				
RLSF	- Requires no human preference labels; highly scalable.	- Prone to reinforcing model's own biases if confidence is miscalibrated (echo chamber effect).	Medium	Medium
Confidence / Entropy Max.	- Very simple to implement; reward signal is "free". - Unsupervised and scalable.	- Naive confidence maximization can lead to overconfident, low-quality outputs (a form of reward hacking).	Low	Low
Function: Scalable Process Supervision				
EDU-PRM	- Automates costly manual annotation of reasoning steps. - Enables scalable process-based supervision.	- Segmentation is heuristic; high entropy might not always be a true logical boundary.	Low (offline)	Medium

Table 6: Critical Analysis of Methods in RL and Reward Modeling. This table provides a comparative overview of key methodologies, focusing on their advantages, failure modes, computational costs, and implementation complexity.

computationally intensive reasoning (like Chain-of-Thought) only when the model exhibits confusion (high uncertainty).

- **Practical Advice:**
  - **Thresholding:** The key challenge is setting an appropriate uncertainty threshold. This is often domain-specific and requires careful tuning on a validation set.
  - **Signal Choice:** Semantic entropy is often more stable than single-token probabilities. For structured tasks like coding, calculating uncertainty at critical decision points (e.g., the first token of a new line) is an effective strategy.

C.2 Autonomous Agents

For agents, uncertainty management is central to ensuring both safety and efficiency in decision-making.

Scenario 1: Building agents that interact with external tools (e.g., search engines, APIs).

- **Recommended Pattern:** Tiered Decision Boundary.
- **Methods:** Start with a simple framework like **UALA** (Han et al., 2024), which follows a "try to solve internally -> measure uncertainty -> call tool if above threshold" logic.
- **Practical Advice:**
  - **Avoid "Tool Overuse":** Setting a reasonable threshold is critical to prevent the agent from making costly and slow tool calls for simple questions.
  - **Tool Uncertainty:** Do not blindly trust tool outputs. For critical applications, model the uncertainty introduced by the tool itself or implement fallback mechanisms (e.g., asking the user for clarification) when a tool returns an unexpected result.

Scenario 2: Agents executing long-horizon, multi-step tasks.

- **Recommended Pattern:** Forward Propagation with Situational-awareness.

- **Methods:** While simple tasks can ignore cumulative uncertainty, complex workflows necessitate a mechanism like **SAUP** (Zhao et al., 2024).
- **Practical Advice:**
  - **Simplified Implementation:** A full information-theoretic framework like **UProp** (Duan et al., 2025) can be complex. A simpler starting point is to accumulate an uncertainty score after each "thought-action-observation" loop and check if it exceeds a "risk" threshold before critical decisions (e.g., calling an expensive API or performing an irreversible action).
  - **Situational Weights:** Not all steps are equally important. Identify "critical nodes" in the task workflow and assign higher weights to the uncertainty measured at these points.

C.3 Reinforcement Learning

In RLHF, uncertainty's primary role is to mitigate reward hacking and achieve more robust alignment.

Scenario 1: Training the Reward Model (RM).

- **Recommended Pattern:** Probabilistic Reward Modeling.
- **Methods:** Move away from traditional RMs that output a single scalar. Instead, adopt models that output a distribution, such as **URM** (Lou et al., 2024), or apply Bayesian techniques to create **Bayesian RMs** (Yang et al., 2024).
- **Practical Advice:**
  - **Distinguish Uncertainty Types: URM** captures aleatoric uncertainty (inherent data randomness) via its architecture, while **Bayesian RMs** capture epistemic uncertainty (model's lack of knowledge) via parameter modeling. The latter is generally more robust for out-of-distribution (OOD) generalization.
  - **Training Objective:** To effectively learn a reward distribution, a Maximum Likelihood Estimation (MLE) objective is often necessary, rather than the traditional Bradley-Terry preference loss.

Scenario 2: Using the RM for policy optimization (e.g., with PPO).

- **Recommended Pattern:** Uncertainty-Aware Adaptive Regularization.
- **Methods:** Dynamically adjust the KL-divergence penalty in the PPO objective based on the RM's uncertainty (Cief et al., 2024).
- **Practical Advice:**
  - **"Trust but Verify":** When the RM is highly uncertain, increase the KL penalty to force the policy to be more conservative and stay closer to the original SFT model. When the RM is confident, decrease the penalty to allow for more exploration. This acts as a confidence-based early stopping mechanism.
  - **Intrinsic Rewards as a Supplement:** For highly exploratory tasks, consider combining the external RM reward with a confidence-based intrinsic reward (e.g., entropy minimization (Agarwal et al., 2025)) to drive more effective autonomous learning.

D LLM Usage

We have used LLM to polish writing for this paper.

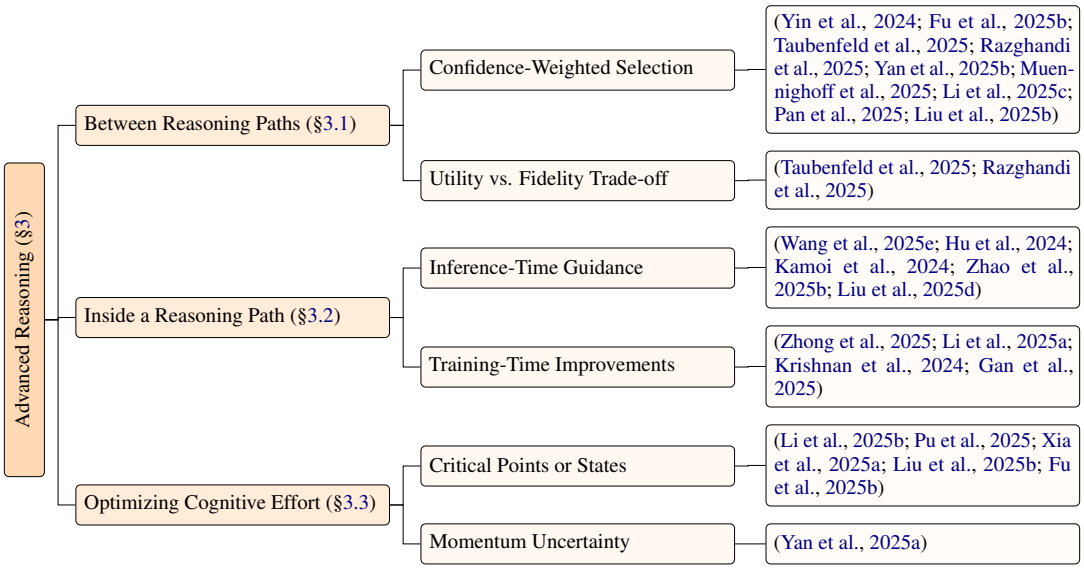

Figure 2: *"Advanced Reasoning"* Categorization

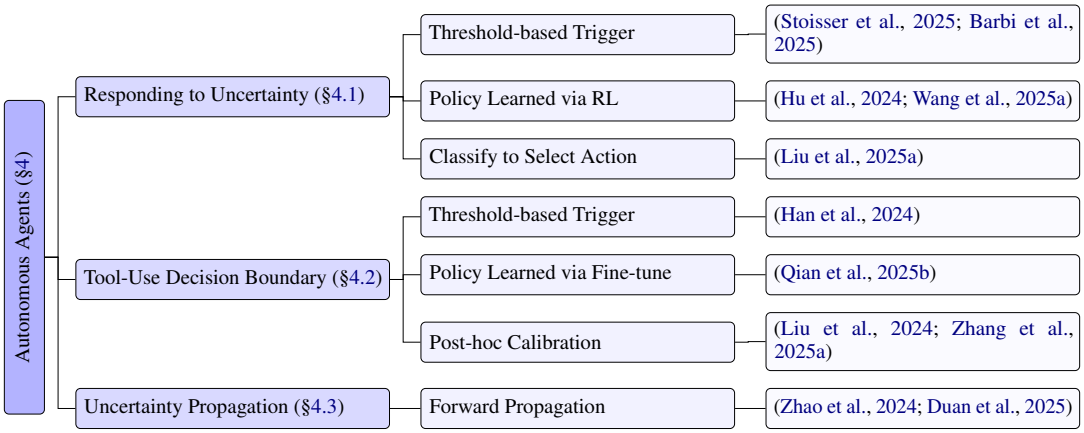

Figure 3: *"Autonomous Agents"* Categorization

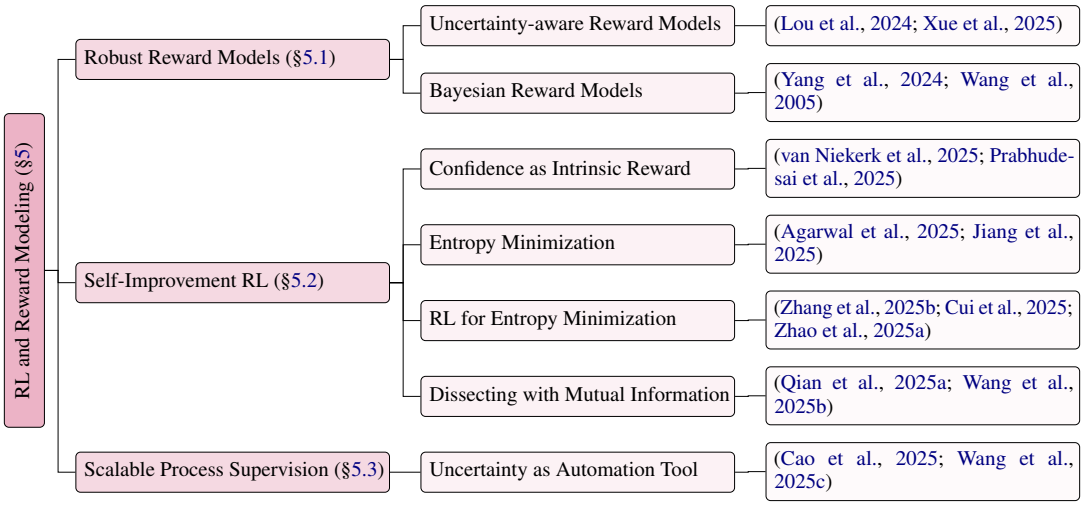

Figure 4: *"RL and Reward Modeling"* Categorization