

Cross-cutting Observations in Artificial Metacognition

Paulo Shakarian

Syracuse University Institute for Artificial Intelligence
Syracuse, NY
pashakar@syr.edu

Abstract

Artificial metacognition is the study of AI systems that monitor their internal processes and control allocation of computational resources. In this paper, we describe three observations that cut across recent studies. These include (1) symbolic representation enables metacognitive cues, (2) metacognitive detection is easier than correction, and (3) multiple perceptual models can enable correction. We provide a general reference framework that can capture the ideas behind several different studies, explore each of these observations, and discuss directions for future research.

1 Introduction

Metacognition is defined in the psychology literature as “the processes that monitor our ongoing thought processes and control the allocation of mental resources” (Ackerman and Thompson 2017) and is well-studied in that discipline (Hemmon 1911; Flavell 1979; Nelson and Narens 1994; Metcalfe and Shimamura 1994). Adapting these ideas to artificial intelligence has given rise to *artificial metacognition* (Cox 2005; Cox and Raja 2011), which can be defined as the study of AI systems that monitor their internal processes and control allocation of computational resources. This area has seen a resurgence in the past few years (Shinn et al. 2023; Kricheli et al. 2024; Zhang et al. 2025; Shakarian 2026). In this paper, we summarize three cross-cutting observations from recent artificial metacognitive studies, in particular:

1. Symbolic representation enables metacognitive cues
2. Detection is easier than correction without new information
3. Multiple perceptual models can enable correction

The first item is important because metacognitive cues are what is used to trigger a metacognitive process. The design of these cues is often a challenge, because if the information were available to the AI system, it would simply use it directly without need for metacognition. However, a symbolic representation affords additional analysis such as identification of excessive symbols (Yang et al. 2025; Ngu, Lee, and Shakarian 2024) or inconsistency (Kricheli et al. 2024; Leiva et al. 2026). The second item provides insight into the design of metacognitive systems. This observation is supported by

Preprint version.

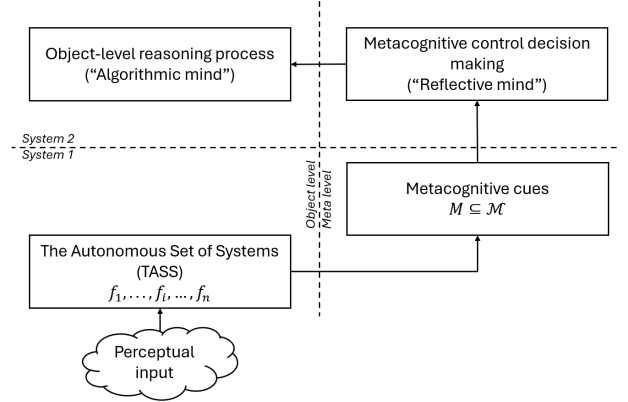


Figure 1: Cognitively Inspired Reference Architecture for Artificial Metacognition. The “algorithmic mind” and “reflective mind” follow the tripartite model of Stanovich (2009).

a formal proof which accounts for some empirical results on correction. It also suggests that fast correction without reasoning may not be feasible when relying on a single model. The third observation revisits the correction issue and identifies several results that suggest the use of multiple models combined with reasoning can provide a path to correction.

In this paper, we provide a reference framework, discuss each of these items in turn, and outline directions for future research.

2 Metacognitive Reference Architecture

By way of technical preliminaries, we present a general framework for artificial metacognition. This framework is primarily focused on metacognitive processes for handling perceptual information, as most work on artificial metacognition has focused on handling perceptual errors. We will discuss implications of the results of this paper for meta-reasoning, the metacognitive processes around reasoning (Ackerman and Thompson 2017), later in the paper.

The architecture is depicted in Figure 1. The framework has two axes. The first axis, following Nelson and Narens (1990, 1994), separates the object level (in this case, perception) from the meta level (processes that assess the object level). The other axis is the System 1–System 2 (dual process

theory) divide (Wason and Evans 1974; Kahneman 2012) where we distinguish near-instantaneous processes that require low working memory (System 1) from deliberate processes that require more working memory (System 2). Perceptual input (e.g., raw data x) is initially processed by a System 1 process on the object level which has been labeled as “The Autonomous Set of Systems” or TASS (Evans and Stanovich 2013). This is represented as a series of functions $f_1, \dots, f_n$ (Shakarian, Simari, and Bastian 2025; Leiva et al. 2026). Normally, these will be well-trained models and when the perceptual input (e.g., x) is understood, we will use the notation f_i instead of $f_i(x)$ to refer to the set of propositions (i.e., class labels) produced by model f_i on the object. So, for example, $\alpha \in f_i$ means that model f_i predicted label α . We use gt for the set of ground truth labels for the object. Hence, the statement $\alpha \in f_i, \alpha \notin gt$ means that model f_i incorrectly predicted class α for the object. This allows for a probabilistic characterization of precision and recall for a given model that we will use in this paper:

$$\text{Precision: } P_{\alpha,i} := \mathbf{P}(\alpha \in gt \mid \alpha \in f_i) \quad (1)$$

$$\text{Recall: } R_{\alpha,i} := \mathbf{P}(\alpha \in f_i \mid \alpha \in gt) \quad (2)$$

The following example from Shakarian, Simari, and Bastian (2025) illustrates a toy scenario in this notation.

Example 2.1 Let f_{car} be a model that, given a sample x , returns a subset of labels $\{\text{ford}, \text{toyota}, \text{dodge}, \text{us}, \text{japan}\}$. So, for example, for perceptual input x , $f_{car}(x) = \{\text{dodge}, \text{us}\}$.

The next portion of the architecture is the set $\mathcal{M}$ of potential signals known as metacognitive “cues” (Ackerman 2019; Thompson, Prowse Turner, and Pennycook 2011).¹ The set of cues present for a given perceptual input x we will denote $M(x)$ (and again, we will use M when x is understood). Following the intuition of cognitive psychologists that cues trigger higher cognitive processes (Evans and Stanovich 2013), if cue c is present in $M(x)$, then we disregard the return value of $f_i(x)$. We note that in Figure 1 the cues, part of “metacognitive monitoring” (Ackerman 2019), are still part of System 1. This is because these processes are part of “automatic” metacognition (Flavell 1979). Cues can trigger a metacognitive control process (Ackerman and Thompson 2017) in which the agent makes decisions based on the cues, often to allocate additional cognitive resources (Thompson 2009) or to communicate cognitive state (Shea et al. 2014). In artificial metacognition, metacognitive cues have been instantiated in various ways, such as an outlier detector (Huang et al. 2023), a model supervised on errors (Cornelio et al. 2023), a model supervised on errors of reasoning traces in LLMs (Tyen et al. 2024), an additional output layer (Hu et al. 2025), alternative class labels (Kricheli et al. 2024), or alternative models (Lee et al. 2024).

In the presence of a metacognitive cue, we can recompute precision and recall values as follows:

$$\text{Precision: } P_{\alpha,i}^c := \mathbf{P}(\alpha \in gt \mid \alpha \in f_i, c \notin M) \quad (3)$$

$$\text{Recall: } R_{\alpha,i}^c := \mathbf{P}(\alpha \in f_i, c \notin M \mid \alpha \in gt) \quad (4)$$

¹The term “conditions” was used in some work (Kricheli et al. 2024; Xi et al. 2025).

These interpretations of precision and recall lead to what Shakarian, Simari, and Bastian (2025) define as an *error-detecting* metacognitive cue with respect to model i , class α when the original precision (Expression 1) is less than or equal to the precision when we account for the cue (Expression 3). Several algorithms for learning rules instantiating error-detecting metacognitive cues and maximizing the increase in precision have been introduced (Kricheli et al. 2024; Lee et al. 2024; Xi et al. 2025). The idea in those papers is that there is some set of candidate cues $\mathcal{M}'$ and we use such approaches to select $\mathcal{M} \subseteq \mathcal{M}'$ that is most effective. We note that rules are not the only way to instantiate metacognitive cues, but we will often refer to them as they provide transparency in the understanding of the artificial metacognitive processes.

3 Symbolic Representation Enables Metacognitive Cues

Cognitive psychologists have identified the communication of cognitive state as one of the functions of metacognition (Shea et al. 2014). We can think of this communication of cognitive state as being in the form of a metalanguage which critiques the underlying thought process. Here, we think of the cognitive state as the output of the model or agent. For example, in psychology, the metacognitive cue of *expectation violation* (Anderson and Fincham 2014) could lead to the identification of an inconsistency in the cognitive state of a human. In other words, this metacognitive cue may lead a human to search for an inconsistency between what is perceived and some domain knowledge.

In artificial metacognition, this type of expectation violation also occurs. For example, aligning with how humans experience expectation violation, Cornelio et al. (2023) found that a learned model, called a “mask predictor,” could identify errors in visual Sudoku that would lead to a correction mechanism that included a symbolic solver (e.g., returning 0 in the output vector position corresponding with the part of the symbolic output deemed incorrect). This aligns with the view of cognitive psychologists such as Evans and Stanovich (2013), where a fast, autonomous metacognitive cue would cause a person to allocate more working memory toward a task. In terms of Figure 1, given mask predictor mp from Cornelio et al. (2023), we define the metacognitive cue $M(x) = \{c_i \text{ such that } mp(f(x))[i] = 0\}$.

However, it also seems that with an artificial metacognitive system the representation itself can be used to derive a metacognitive cue. Let us return to our example (adapted from Shakarian, Simari, and Bastian 2025) and look at a sample metacognitive cue.

Example 3.1 Consider model f_{car} from Example 2.1 and the metacognitive cue $c_1 := \text{toyota} \wedge \text{us}$. Intuitively, this cue occurs when the input is classified as both *toyota* and *us* by the model.

The metacognitive cue in Example 3.1 can be viewed in several ways. First, this could arise from a function that evaluates the output of the model, as in Cornelio et al. (2023) (e.g., the mask predictor). If we use a black box approach, as in that work, c_1 would not be defined as in Example 3.1 but

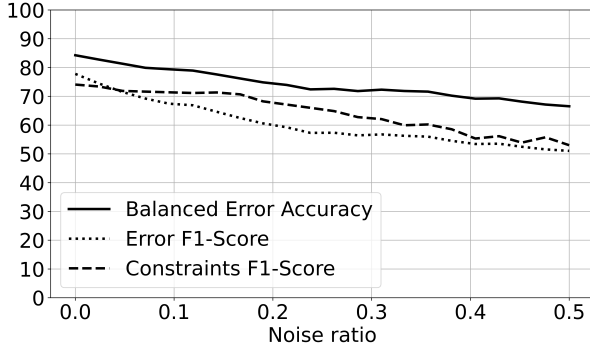


Figure 2: Balanced Error Accuracy, Error F1-score, and constraints F1-score results of the metacognitive cue-learning approach of Kricheli et al. (2024) (“Focused EDR”) for varying noise ratios on the ImageNet50 dataset. Constraints F1 measures agreement between the recovered constraints and a held-out ground-truth set. From Kricheli et al. (2024).

rather as an output of a secondary model. However, the work of Kricheli et al. (2024) uses a rule-learning approach (“Error Detection Rules” or EDR) in which such a constraint can be learned from the errors of the model. This approach could not only serve as an *error-detecting* cue, but that it could also recover ground truth constraints (see Figure 2 from Kricheli et al. (2024)). We note that during training, the process is the reverse of what was observed in the cognitive psychology literature about metacognition: the representation was used to derive the cue. However, the cue’s use during inference in Cornelio et al. (2023) more resembles the view of cognitive psychology. We note that Kricheli et al. (2024) did not use the cue to make corrections at inference time, but rather used the cue to re-train the model with constraints. When compared to Cornelio et al. (2023), the inference-time correction mechanism is absent in Kricheli et al. (2024). Later work on error-detection rules presented reasoning-based correction (Leiva et al. 2026).

Kricheli et al. (2024) recovered a logical constraint while Cornelio et al. (2023) detected an unspecified error in part of the symbolic output. While it is easy to assume that the black-box model may just be identifying inconsistencies like c_1 in Example 3.1, there is another possibility. It could be that the symbols returned by the model are likely to indicate an error. In our running example, the cue $toyota \wedge japan$ could also be an error-detecting cue (e.g., the model was not well-trained on Toyota cars). Another place where a string of symbols, though correct, may indicate an error would be with generative AI approaches. For example, in Shakarian et al. (2023) it was noted that long chains of addition produced by an LLM attempting to solve a math word problem were indicative of error. Likewise, such symbolic output can be evaluated with heuristics as done in Ngu, Lee, and Shakarian (2024) by applying Gini and entropy measures to symbol counts. These techniques were used together to create metacognitive cues in EDCIM, an LLM wrapper to solve math word problems (Yang et al. 2025). An advantage of this approach was that the precise issue with the symbols (lack of diversity,

explosion of a certain symbol type) was fully explainable, and that information could be reported back to the LLM in a corrective step. Another rule-based metacognitive system known as RECOVER (Cornelio and Diab 2024) also leverages metacognitive rules to alert the system to failure and leverage symbolic output to improve LLM-generated recovery plans. An example of the corrective re-prompting used in EDCIM is shown in Figure 3.

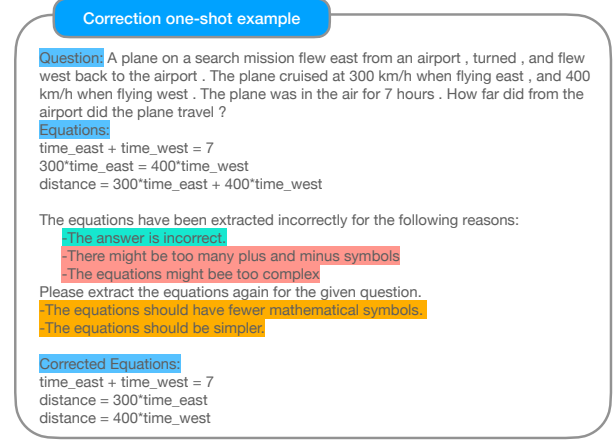


Figure 3: One-shot example for Answer Correction from Yang et al. (2025).

Another approach in which the representation of state can be used to derive a cue would involve an evaluation of the model output with a logical reasoner against a set of constraints as done in Yang, Neary, and Topcu (2024). Here again c_1 would not be explicit, but the emergence of an inconsistency itself would be the cue. This approach may require the presence of domain knowledge, but in some recent work one can rely on the internal consistency of the model output (Leiva et al. 2026). We also note that this observation is only stating that symbolic representation can enable cues, it does not preclude other sources of cues. Statistical methods such as out-of-distribution detection (Hendrycks and Gimpel 2017) and conformal prediction (Angelopoulos and Bates 2023) can also provide cues. We have seen evidence of such techniques employed in a metacognitive strategy with abductive learning. Here an outlier detection model was used to trigger reasoning (Huang et al. 2023). The cognitive psychology literature documents a variety of cues that humans use, and we would expect a general intelligence to have a metacognitive capability that can employ cues of different types.

4 Detection Is Easier than Correction without New Information

A study by Huang et al. (2024) showed strong empirical evidence against intrinsic LLM self-correction. In that study, they found that several previously explored methods for LLM self-correction failed when issues such as prompt design, unfair baselines, and use of oracles were more rigorously ac-

counted for. A later work, EDCIM (Yang et al. 2025) provided a useful approach as a model router, i.e., if a metacognitive cue was identified (e.g., see Figure 3), then a more powerful/expensive model was re-prompted. In this way, a smaller LLM could do most of the work and the more expensive model was used sparingly. However, it seems that this setup could still not escape the result of Huang et al. (2024). If the same model was re-prompted, the results did not improve over the baseline “CRITIC” LLM correction routine (Gou et al. 2024), which verifies outputs via external tools (e.g., a code interpreter) rather than metacognitive cues.

A similar finding came about with an attempt to create “correction” rules, an extension of the EDR framework. Correction rules were used successfully for a binary classification problem (Lee et al. 2024), but they did not provide a significant performance increase when extended to multi-class classification (Xi et al. 2025). However, the limited improvement in classification occurred despite good performance in error detection. This is evidenced by an examination of the precision–recall tradeoff of the combined detection–correction approach, shown for one of the experiments in Figure 4. Here we see an increase in precision (computed as per Expression 3) but little change in F1.

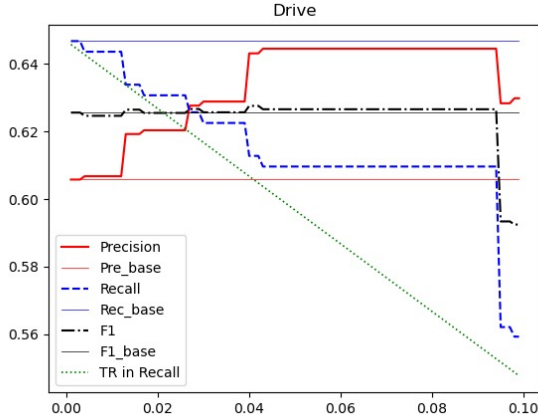


Figure 4: Results from Xi et al. (2025) after application of error detection and correction rules to the LRCN trajectory classification model (for the “Drive” class) as a function of hyperparameter ϵ (which can be interpreted as allowable recall reduction). Note improved precision but little change in F1.

We now show why the results of studies like Huang et al. (2024) and Yang et al. (2025) may be the best achievable with a single model. Let us consider the following example.

Example 4.1 Consider f_{car} of Example 2.1, and suppose that its precision, measured over the whole population, is $P_{dodge} = 0.75$, $P_{ford} = 0.55$, and $P_{toyota} = 0.65$. Now consider a set S of eleven samples that f_{car} cannot tell apart. Suppose the ground truth over S is distributed as shown in Figure 5. Let c_2 be the cue associated with these samples. Clearly, c_2 is error detecting: disregarding these samples

would improve precision, since the best the model can do is classify all of them as *dodge*. This makes c_2 a useful warning to a user of f_{car} . However, without another model, the best possible “correction” would be to label these samples as *dodge*, as any other choice would lead to worse accuracy.

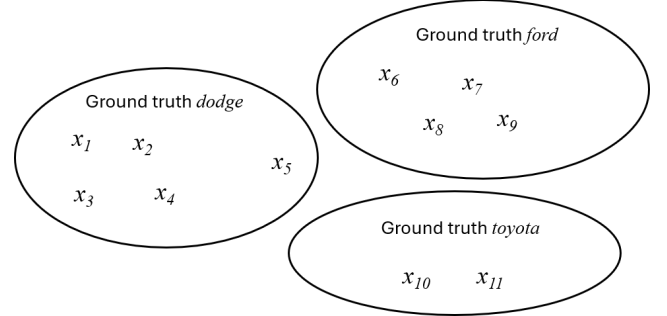


Figure 5: The eleven samples of Example 4.1. All samples are associated with cue c_2 and appear the same to f_{car} . Ground truth class boundaries are shown in solid lines.

Examples of this sort led to an analytical result proved in a “limits of reclassification” theorem in Shakarian, Simari, and Bastian (2025) (Theorem 4.2 of that paper), the statement of which is below.

Limits of Reclassification Theorem. Given disjoint classes α, β (i.e., the model can return one or the other but not both), cue c and model f , if we have that $\mathbf{P}(\beta \in gt \mid \alpha \in f, c \in M) \leq \mathbf{P}(\beta \in gt \mid \beta \in f)$, then: $\mathbf{P}(\beta \in gt \mid \beta \in f) \geq \mathbf{P}(\beta \in gt \mid \beta \in f \vee (\alpha \in f, c \in M))$.

This statement generalizes Example 4.1: the precondition compares the precision of the target class β (the class we wish to correct to) on the cued samples with its overall precision. In Example 4.1 the classes *toyota* and *ford* both meet the precondition, so the precision for those classes would decrease (thus decreasing overall accuracy). This gives no option for correction. However, this result does not account for the use of different models or representations, which is the focus of the next section.

5 Multiple Perceptual Models Can Enable Correction

The negative result of the prior section does not mean that correction is not possible. The theorem’s precondition holds most clearly when the samples look identical to the classifier; correction becomes possible when that condition is broken, as the following approaches show.

The Limits of Reclassification Theorem does not comment on correction via logic-based repair, as done in Cornelio et al. (2023). Likewise, other neurosymbolic approaches take a similar strategy: STLnet (Ma et al. 2020) at inference time and abductive learning (Dai et al. 2019) at training time. These approaches can be viewed as leveraging properties of returned logic (such as inconsistency or non-entailment)

driving the repair procedure. The aforementioned papers require domain knowledge encoded in the logic.

There is an LLM-analog to logic-based repair noted in Tyen et al. (2024). Here an additional classifier is used to identify reasoning trace mistakes from the LLM, and when these are pointed out with the classifier, the authors showed that the LLM is able to perform correction. However, we note that in the general case (without the use of an additional classifier) that LLM self correction without additional knowledge generally does not lead to improved results. See Kamoi et al. (2024) for a survey.

The Limits of Reclassification Theorem also does not address the case where multiple models are used. We can think of this as providing a counterpoint to Example 4.1, as done in Example 5.1.

Example 5.1 Consider f_{car} of Example 2.1, and a second model f'_{car} . This model performs worse than the first, but partitions the eleven samples (identified by cue c_2) of Figure 5 differently, as shown by the classification boundary in Figure 6. In group A of Figure 6, f'_{car} returns *dodge* and in group B it returns *ford*. We note that there are other samples not depicted (that are not associated with c_2) which look identical to f'_{car} in these two groups (but samples in group A look different to f'_{car} than samples in group B).

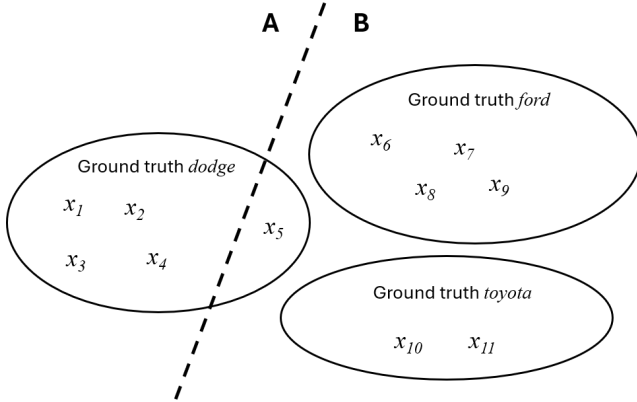


Figure 6: The eleven samples of Example 5.1. All samples are associated with cue c_2 and appear the same to f_{car} . Ground truth class boundaries are shown in solid lines. The dashed line shows the classification boundary of model f'_{car} .

In Example 5.1 the second model (f'_{car}) does not perform as well as the first. We also note that c_2 only provides useful error detection for over half of the samples (group B). However, it encounters the same problem as with the first model (there is no good correction for the predictions of f'_{car} within group B).² That said, we can see that this clearly provides more information once we have the second model. For example, we can think of a cue c_3 defined as follows: $c_3 := c_2 \wedge (ford \in f'_{car})$. We note that this new cue violates

²If we consider cue c_2 and f'_{car} predicting *ford*, then the precision would be 0.57, but picking *toyota* or *dodge* makes the precision for that population worse.

the precondition of the Limits of Reclassification Theorem. This cue not only detects an error in the first model, but provides an easy correction to f_{car} . This is the strategy employed by Lee et al. (2024), where additional models are used as metacognitive cues. In that work, the authors trained additional models in an arbitrary manner, so that model-class label pairs could serve as candidate cues, creating set $\mathcal{M}'$ (the set from which an algorithm will pick the cues to use, $\mathcal{M}$).

This resolves the problem of identifying potential failure modes or correction opportunities a priori. It differs from traditional ensembles in that there is no requirement for models to vote on a particular class label. Rather, the use of the model can be determined directly from the learned metacognitive cues. In this way, the designer can train models on different datasets and/or with different hyperparameters and use them as candidate cues. This leaves the rule learner to figure out which ones will be the most useful.

However, the paradigm presented in Lee et al. (2024) does have some shortcomings. First, it requires the identification of a lead model. Here, the other models only serve as cues. This differs from a perspective in cognitive psychology that there is a collection of System 1 perceptual systems referred to as “The Autonomous Set of Systems” or TASS (Evans and Stanovich 2013). In Lee et al. (2024), what should be in the TASS is relegated to use as a metacognitive cue, which raises the question: if one trains a collection of models for a given task, how do we know which should be the perceptual model and which ones should be cues? A second limitation with this setup is that there is no reasoning about what to do in a failure case. However, this is mostly due to the fact that Lee et al. (2024) examine a binary classification problem. To mirror the concepts of the cognitive psychologists (Ackerman and Thompson 2017; Evans and Stanovich 2013), we would want the cues to trigger a reasoning process. This also makes sense, because if we simply use other models as cues, it is not clear which one we use to assign a class label in the case where multiple cues are present (associated with different class labels).

A different way to deal with the problem is to treat all perceptual models as part of the TASS and reason jointly over their outputs, the metacognitive cues, and domain knowledge. This is the approach taken in Leiva et al. (2026), depicted in Figure 7 where there are n machine learning models which are considered along with metacognitive cues and domain knowledge to reach a class assignment. Like Lee et al. (2024), it uses multiple models (though here it relies on a reasoning process to do so). However, unlike that work, it does not require any model to have primacy. It also uses metacognitive cues, as in earlier work (Kricheli et al. 2024; Xi et al. 2025; Yang et al. 2025). Its cue candidates (determining set $\mathcal{M}'$), however, must be engineered. Further, like the neurosymbolic work (Cornelio et al. 2023; Ma et al. 2020; Dai et al. 2019) it can leverage domain knowledge as well. All of these inputs are considered together in a reasoning process (using abductive inference) at inference time. However, unlike the proposal of cognitive scientists (Ackerman and Thompson 2017; Evans and Stanovich 2013), the reasoning process here happens for *all* perceptual input,

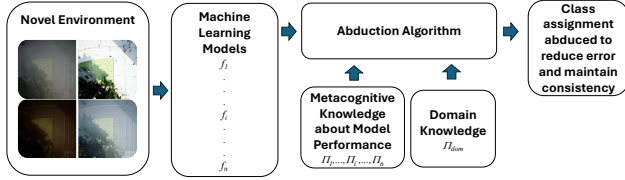


Figure 7: Overview of Consistency-based Abductive Reasoning Approach from Leiva et al. (2026). Here n machine learning models perceive a novel environment. Their results are considered with domain knowledge and metacognitive information about the models (learned independently and with the same training data) to abduce a set of results that exhibit consistency and reduce perceptual errors.

rather than being triggered by a metacognitive cue. Despite these shortcomings, Leiva et al. (2026) show that multiple models with lower precision (due to errors detected via metacognitive cues) can provide superior performance when combined. Specifically, their approach outperformed both a voting-based ensemble and the best single model on object detection.

6 Directions for Future Research

The observations of this paper also suggest further areas of exploration. First, symbolic representations can provide metacognitive cues, but what about other types of representations? For example, can vector representations of the output layers of the models also provide potential cues (Ma et al. 2025)? There is also some evidence that activation patterns could provide the same (de Sousa Ribeiro and Leite 2021). Perhaps the work on mechanistic interpretability (Bereska and Gavves 2024) can be leveraged to provide cues as well. However, we note that symbolic representations provide inherently explainable cues, a property lost when a vector representation is used. Further, symbolic representations also afford logic-based repair as a correction mechanism. Losing this quality with vector representations would exacerbate the “detection is easier than correction” problem.

Second, the observations of this paper raise several questions on computational efficiency and the allocation of computational resources. For example, consider an LLM that returns an unbounded output of logical symbol statements. Determining with exact methods whether those statements are consistent is at least linear in the output. While tractable consistency checking is possible for certain fragments of logic, exact consistency checking is NP-complete even for propositional logic, and undecidable in general for first-order and higher-order logic. Part of the reason why metacognitive cues exist in humans is that they are used to essentially regulate computational resources. The development of fast methods to identify potential logical inconsistencies (e.g., Cornelio et al. 2023; Hu et al. 2025) may become more important.

Third, there is the question of what to do once a metacognitive signal is seen. In recent work on abductive ensembling (Leiva et al. 2026) the metacognitive cue is wrapped into a corrective process. While these methods improve re-

sults, they do not conserve computational resources. This is because the more expensive process occurs in any case, whether it be formal reasoning (Leiva et al. 2026) or expensive LLM calls (Gou et al. 2024).

Finally, artificial metacognition research has largely been centered on correction of perceptual (vision in particular) and language models. Comparatively little recent work, however, has addressed meta-reasoning (Ackerman and Thompson 2017). In particular, this would involve finding cues that can tell the system if reasoning should continue, accept current results, conduct more exploration, re-obtain sensory input, allocate more time, or allocate more memory. That said, there has been earlier work on meta-reasoning (Horvitz 1987; Russell and Wefald 1991; Cox and Raja 2011), which may see renewed interest, especially as global computational demands increase.

Acknowledgments

This research was supported by the Army Research Office (ARO) grant W911NF-24-1-0007.

References

- Ackerman, R. 2019. Heuristic Cues for Meta-Reasoning Judgments: Review and Methodology. *Psychological Topics*, 28(1): 1–20.
- Ackerman, R.; and Thompson, V. A. 2017. Meta-Reasoning: Monitoring and Control of Thinking and Reasoning. *Trends in Cognitive Sciences*, 21(8): 607–617.
- Anderson, J. R.; and Fincham, J. M. 2014. Extending problem-solving procedures through reflection. *Cognitive Psychology*, 74: 1–34.
- Angelopoulos, A. N.; and Bates, S. 2023. Conformal Prediction: A Gentle Introduction. *Foundations and Trends in Machine Learning*, 16(4): 494–591.
- Bereska, L.; and Gavves, E. 2024. Mechanistic Interpretability for AI Safety – A Review. *Transactions on Machine Learning Research*. Survey Certification, Expert Certification.
- Cornelio, C.; and Diab, M. 2024. Recover: A Neuro-Symbolic Framework for Failure Detection and Recovery. In *2024 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, 12435–12442. Abu Dhabi, United Arab Emirates: IEEE.
- Cornelio, C.; Stuehmer, J.; Hu, S. X.; and Hospedales, T. 2023. Learning where and when to reason in neuro-symbolic inference. In *The Eleventh International Conference on Learning Representations*.
- Cox, M. T. 2005. Metacognition in computation: A selected history. In *AAAI Spring Symposium: Metacognition in Computation*, 1–17.
- Cox, M. T.; and Raja, A. 2011. *Metareasoning: Thinking about thinking*. MIT Press.
- Dai, W.-Z.; Xu, Q.; Yu, Y.; and Zhou, Z.-H. 2019. Bridging machine learning and logical reasoning by abductive learning. *NeurIPS*, 32.

- de Sousa Ribeiro, M.; and Leite, J. 2021. Aligning Artificial Neural Networks and Ontologies towards Explainable AI. In *Proceedings of the Thirty-Fifth AAAI Conference on Artificial Intelligence (AAAI-21)*, volume 35, 4932–4940. AAAI Press.
- Evans, J. S. B.; and Stanovich, K. E. 2013. Dual-process theories of higher cognition: Advancing the debate. *Perspectives on Psychological Science*, 8(3): 223–241.
- Flavell, J. H. 1979. Metacognition and cognitive monitoring: A new area of cognitive–developmental inquiry. *American Psychologist*, 34(10): 906–911.
- Gou, Z.; Shao, Z.; Gong, Y.; Shen, Y.; Yang, Y.; Duan, N.; and Chen, W. 2024. CRITIC: Large Language Models Can Self-Correct with Tool-Interactive Critiquing. In *The Twelfth International Conference on Learning Representations (ICLR)*.
- Hendrycks, D.; and Gimpel, K. 2017. A Baseline for Detecting Misclassified and Out-of-Distribution Examples in Neural Networks. In *International Conference on Learning Representations (ICLR)*.
- Henmon, V. A. C. 1911. The relation of the time of a judgment to its accuracy. *Psychological Review*, 18(3): 186–201.
- Horvitz, E. J. 1987. Reasoning about Beliefs and Actions under Computational Resource Constraints. In *Proceedings of the Third Workshop on Uncertainty in Artificial Intelligence (UAI-87)*, 429–444. Seattle, WA.
- Hu, W.-C.; Dai, W.-Z.; Jiang, Y.; and Zhou, Z.-H. 2025. Efficient Rectification of Neuro-Symbolic Reasoning Inconsistencies by Abductive Reflection. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 39, 17333–17341.
- Huang, J.; Chen, X.; Mishra, S.; Zheng, H. S.; Yu, A. W.; Song, X.; and Zhou, D. 2024. Large Language Models Cannot Self-Correct Reasoning Yet. In *The Twelfth International Conference on Learning Representations (ICLR)*.
- Huang, Y.-X.; Dai, W.-Z.; Jiang, Y.; and Zhou, Z.-H. 2023. Enabling Knowledge Refinement upon New Concepts in Abductive Learning. *Proceedings of the AAAI Conference on Artificial Intelligence*, 37(7): 7928–7935.
- Kahneman, D. 2012. *Thinking, Fast and Slow*. London: Penguin.
- Kamoi, R.; Zhang, Y.; Zhang, N.; Han, J.; and Zhang, R. 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.
- Kricheli, J. S.; Vo, K.; Datta, A.; Ozgur, S.; and Shakarian, P. 2024. Error detection and constraint recovery in hierarchical multi-label classification without prior knowledge. In *Proceedings of the 33rd ACM International Conference on Information and Knowledge Management*, 3842–3846.
- Lee, N.; Ngu, N.; Sahdev, H. S.; Motaganahall, P.; Chowdhury, A. M. S.; Xi, B.; and Shakarian, P. 2024. Metal Price Spike Prediction via a Neurosymbolic Ensemble Approach. In *IEEE International Conference on Data Mining, ICDM 2024 - Workshops, Abu Dhabi, United Arab Emirates, December 9, 2024*, 106–110. IEEE.
- Leiva, M.; Ngu, N.; Kricheli, J. S.; Taparia, A.; Senanayake, R.; Shakarian, P.; Bastian, N. D.; Corcoran, J.; and Simari, G. 2026. Consistency-based Abductive Reasoning over Perceptual Errors of Multiple Pre-trained Models in Novel Environments. In *Proceedings of the Fortieth AAAI Conference on Artificial Intelligence (AAAI-26)*, volume 40 of *Proceedings of the AAAI Conference on Artificial Intelligence*, 19216–19223. Singapore: AAAI Press.
- Ma, M.; Gao, J.; Feng, L.; and Stankovic, J. 2020. STL-net: Signal Temporal Logic Enforced Multivariate Recurrent Neural Networks. In Larochelle, H.; Ranzato, M.; Hadsell, R.; Balcan, M.-F.; and Lin, H.-T., eds., *Advances in Neural Information Processing Systems*, volume 33, 14604–14614. Curran Associates, Inc.
- Ma, Y.; Ren, H.; Wang, B.; Jin, J.; Velipasalar, S.; and Qiu, Q. 2025. LVP-CLIP: Revisiting CLIP for Continual Learning with Label Vector Pool. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR) Workshops*, 231–240.
- Metcalfe, J.; and Shimamura, A. P., eds. 1994. *Metacognition: Knowing about Knowing*. Bradford Books. Cambridge, MA: MIT Press.
- Nelson, T. O.; and Narens, L. 1990. Metamemory: A Theoretical Framework and New Findings. In Bower, G. H., ed., *Psychology of Learning and Motivation*, volume 26 of *Psychology of Learning and Motivation*, 125–173. Academic Press.
- Nelson, T. O.; and Narens, L. 1994. Why Investigate Metacognition? In Metcalfe, J.; and Shimamura, A. P., eds., *Metacognition: Knowing about Knowing*, 1–25. Cambridge, MA: MIT Press.
- Ngu, N.; Lee, N.; and Shakarian, P. 2024. Diversity Measures: Domain-Independent Proxies for Failure in Language Model Queries. In *2024 IEEE 18th International Conference on Semantic Computing (ICSC)*, 176–182. Laguna Hills, CA, USA: IEEE.
- Russell, S. J.; and Wefald, E. H. 1991. Principles of Metareasoning. *Artificial Intelligence*, 49(1–3): 361–395.
- Shakarian, P. 2026. Toward Artificial Metacognition. In *Proceedings of the Fortieth AAAI Conference on Artificial Intelligence (AAAI-26)*, volume 40 of *Proceedings of the AAAI Conference on Artificial Intelligence*, 41000–41005. Singapore: AAAI Press.
- Shakarian, P.; Koyyalamudi, A.; Ngu, N.; and Mareedu, L. 2023. An Independent Evaluation of ChatGPT on Mathematical Word Problems (MWP). In Martin, A.; Fill, H.-G.; Gerber, A.; Hinkelmann, K.; Lenat, D.; Stolle, R.; and van Harmelen, F., eds., *Proceedings of the AAAI 2023 Spring Symposium on Challenges Requiring the Combination of Machine Learning and Knowledge Engineering (AAAI-MAKE 2023)*, volume 3433 of *CEUR Workshop Proceedings*. San Francisco, CA, USA: CEUR-WS.org.
- Shakarian, P.; Simari, G. I.; and Bastian, N. D. 2025. Probabilistic Foundations for Metacognition via Hybrid-AI. *Proceedings of the AAAI Symposium Series*, 5(1): 389–393.
- Shea, N.; Boldt, A.; Bang, D.; Yeung, N.; Heyes, C.; and Frith, C. D. 2014. Supra-personal cognitive control and

metacognition. *Trends in Cognitive Sciences*, 18(4): 186–193.

Shinn, N.; Cassano, F.; Gopinath, A.; Narasimhan, K.; and Yao, S. 2023. Reflexion: Language agents with verbal reinforcement learning. *Advances in Neural Information Processing Systems*, 36: 8634–8652.

Stanovich, K. E. 2009. Distinguishing the reflective, algorithmic, and autonomous minds: Is it time for a tri-process theory? In Evans, J.; and Frankish, K., eds., *In Two Minds: Dual Processes and Beyond*. Oxford University Press.

Thompson, V. A. 2009. Dual-process theories: A metacognitive perspective. In Evans, J.; and Frankish, K., eds., *In Two Minds: Dual Processes and Beyond*, 171–195. Oxford University Press.

Thompson, V. A.; Prowse Turner, J. A.; and Pennycook, G. 2011. Intuition, reason, and metacognition. *Cognitive Psychology*, 63(3): 107–140.

Tyen, G.; Mansoor, H.; Carbune, V.; Chen, P.; and Mak, T. 2024. LLMs cannot find reasoning errors, but can correct them given the error location. In Ku, L.-W.; Martins, A.; and Srikumar, V., eds., *Findings of the Association for Computational Linguistics: ACL 2024*, 13894–13908. Bangkok, Thailand: Association for Computational Linguistics.

Wason, P.; and Evans, J. 1974. Dual processes in reasoning? *Cognition*, 3(2): 141–154.

Xi, B.; Scaria, K.; Bavikadi, D.; and Shakarian, P. 2025. Rule-Based Error Detection and Correction to Operationalize Movement Trajectory Classification. In *IJCAI Workshop on Spatio-Temporal Reasoning and Learning*.

Yang, Y.; Cornelio, C.; Leiva, M.; and Shakarian, P. 2025. Error Detection and Correction for Interpretable Mathematics in Large Language Models. *Proceedings of the AAAI Symposium Series*, 7(1): 283–292.

Yang, Y.; Neary, C.; and Topcu, U. 2024. Multimodal Pre-trained Models for Verifiable Sequential Decision-Making: Planning, Grounding, and Perception. In *Proceedings of the 23rd International Conference on Autonomous Agents and Multiagent Systems*, AAMAS '24, 2011–2019. Richland, SC: International Foundation for Autonomous Agents and Multiagent Systems.

Zhang, D.; Lei, J.; Li, J.; Wang, X.; Liu, Y.; Yang, Z.; Li, J.; Wang, W.; Yang, S.; Wu, J.; Ye, P.; Ouyang, W.; and Zhou, D. 2025. Critic-V: VLM Critics Help Catch VLM Errors in Multimodal Reasoning. In *Proceedings of the Computer Vision and Pattern Recognition Conference*, 9050–9061.