International Journal of Computer Information Systems and Industrial Management Applications
ISSN 2150-7988 Volume 18(21s) 2026 pp. 2055-2062
https://doi.org/10.70917/ijcisim-2026-5944
Article

Meta-learning in Artificial Intelligence: Syatematic Literature Review and Its Application in Simulated Environments

Silvana Yacchirema[1], Arturo Bedon[2], Ramiro Delgado[3]

[1,2]Universidad Central, Carrera Sistemas de Información, Quito-Ecuador, sayacchirema@uce.edu.ec, cbedon@uce.edu.ec
3 Universidad de las Fuerzas Armadas ESPE, Carrera Software, Quito-Ecuador, rndelgado@espe.edu.ec

Abstract: Currently, the increasing development and training of artificial intelligence, using various machine learning algorithms present one of the biggest challenges for those involved in their development, namely; selecting the right learning algorithm for each problem. This is because selecting the correct learning algorithm traditionally relies on expert knowledge of many different machine learning algorithms which can take considerable amounts of time to experiment with. although the use of meta-learning has been proposed as an alternative method to automate this task by utilizing past experience. there are still issues related to the feasibility of various meta-learning methods and how theoretical applications relate to practical implementations. therefore, this study will seek to overcome these deficiencies and through a systematic literature review, assess the current state of meta-learning methodologies to determine which methodology(s) have the potential to be both the most effective and feasible for future use. to further support these findings, an experimental prototype application will be proposed that utilizes the methodology(ies) identified as being most effective in simulated environments under various conditions, providing evidence for the importance of autonomous decision making when it comes to AI based learning.
Keywords: Meta-learning, artificial intelligence, learning algorithms.

1. Introduction

The expansion of Artificial Intelligence (AI) and machine learning in the industrial sector has made it possible to solve various complex problems; however, traditional machine learning has the drawback of requiring algorithm selection and hyperparameter tuning, as the cost is high because the model must be retrained from scratch for each new task. In this regard, meta-learning emerges as a solution with greater potential, defined as learning from experience by running different algorithms on various datasets to learn new tasks (Kaiyi Ji J. Y., 2022). For example, meta-learning techniques can reduce the time required to adapt to a new task, as well as support AutoML systems by identifying the best algorithms based on previous tasks, thereby "accelerating the AutoML process and achieving better results in the same amount of time" (Monteagudo, 2019). Thus, meta-learning, based on prior knowledge, provides a warm start to the search for solutions rather than relying on searches that require significant computational power (such as exhaustive cross-validation) (Dr. Ashish Kr. Chakraverti, 2020).

Practical limitations remain a significant issue with current techniques. Recent studies indicate that many meta-learning tools in AutoML are limited to certain subsets of algorithms, such as those tied to a particular library, and lack sufficient adaptability to integrate diverse methods into complex multifactorial problems. Specifically, Monteagudo et al. (2019) note that "these meta-learning tools are not flexible enough to address practical problems that require the combination of algorithms and systems of different natures." Furthermore, the effectiveness of meta-methods depends largely on the quality of features and known information; for example, similarity-based approaches for datasets require specifying neighborhood parameters and maintaining extensive metadata caches (Llerena, 2021).


Copyright: © 2026 by the authors

Given the multitude of approaches (gradient-based optimization, algorithm classification methods, integrated AutoML systems, etc.), along with the lack of universal solutions (by analogy to "There's no such thing as a free lunch"), it is imperative to conduct a Systematic Literature Review (SLR). This study synthesizes the existing literature on meta-learning and AutoML and highlights the strengths and weaknesses of each method. This completes the rationale for the SLR in order to build a fundamental foundation to guide more innovative experimental work in richly configurable simulated environments.

Objective

Identify and analyze the most effective meta-learning algorithms in order to select the most suitable model for its application in an adaptive simulated environment.

2. Methods

The methodology used in this first phase of the research was a SLR. This methodology was designed to answer a specific question and to identify, analyze, and interpret the available evidence in an unbiased and reproducible manner (Carrizo, 2018). Therefore, the following question was formulated: Which meta-learning algorithms have demonstrated the greatest effectiveness in recent years, and how can they be applied in an adaptive simulated environment? This question guided the search for relevant information, which will serve as the foundation for the second phase of the study, in which the experimental application will be developed based on the most suitable learning algorithm identified by the SLR.

The PRISMA method was used, and two digital databases.

Starting with the ACM Digital Library, where two search strings were used. The first: "meta-learning" AND "artificial intelligence" AND "models"; this search yielded 379 initial results, of which, after applying a filter for the 7-year range between 2018 and 2025, 324 results were obtained, of which 2 were selected as they aligned with the research objective.

The second search string applied to the same repository: meta-learning AND "artificial intelligence" AND (engineering OR software), yielded significantly more results—11,877 documents. After applying a filter for publication years between 2018 and 2025, 8,095 results were obtained; after reading their abstracts, 5 documents were selected for inclusion in the RSL.

The next repository in which the search was conducted was Google Scholar, where the same search string was applied: meta-learning AND "artificial intelligence" AND (engineering OR software). The initial results yielded a total of 57,700 articles. After applying a date range filter for the years 2018–2025, the results were narrowed down to 21,400. Following the application of the inclusion criteria, two documents useful for the research were selected.

Finally, in the same Google Scholar repository, the search term "meta-learning" AND "algorithm" was entered, which yielded the highest number of results so far with 104,000 results. When filtered by date range between 2018 and 2025, along with a filter for the Spanish language to prioritize them in this case, the results were reduced to 124. Of these, after reading the abstracts and checking for open access, two documents were selected.

The following criteria were taken into account for this selection:

Inclusion criteria:

1. Relevant Topics:
    - We sought articles, theses, reviews, or even conference papers that explicitly address meta-learning, specifically in the field of artificial intelligence as applied to software engineering, including, for example, the development, maintenance, optimization, or adaptability of intelligent systems.
    - Also, works that use meta-learning methods or algorithms, such as MAML, Reptile, ProtoNets, and similar ones, applied to problems—whether real or simulated—in software or intelligent systems.

- Studies demonstrating practical applications or theoretical frameworks that combine meta-learning with other techniques, such as machine learning, deep learning, or AutoML, all applied to software problems in some way.

2. Time Frame:
   - Publications from 2018 to 2025, to ensure that the information is up-to-date.
3. Access to Publications:
   - Documents that are freely or openly accessible to facilitate consultation and citation.

Exclusion Criteria:

1. Unrelated Topics:
   - Documents that do not mention meta-learning or related machine learning techniques.
   - Works focused exclusively on AI topics unrelated to software engineering.
   - Articles on similar areas that do not apply meta-learning.
2. Outside the Time Range:
   - Publications prior to 2018 or after 2025
3. Access:
   - Documents not available via open access platforms such as IEEE or with limited access.
4. Language:
   - Documents in languages other than Spanish or English.

This process was carried out through stages of identification, selection, eligibility, and inclusion, as shown in Figure 1.

Figure 1: Flowchart of the Prisma Method

3. Results

The review identified 11 records—7 in the ACM Digital Library and 4 in Google Scholar—the details of which are shown in Table 1 below.

No.	Country	Title	Author(s)	Topics Covered
1	China	MetaLog: Generalizable Cross-System Anomaly Detection from Logs with Meta-Learning	Chenyangguang Zhang, Tong Jia, Guopeng Shen, Pinyan Zhu, Ying Li.	The model 'learns to learn' general patterns of anomalies. Optimization based on MAML
2	Germany	Auto-sklearn 2.0: hands-free AutoML via meta-learning	Matthias Feurer, Katharina Eggensperger, Stefan Falkner, Marius Lindauer, Frank Hutter.	Improve AutoML so that it is more efficient and easier to use. Meta-learning without meta-features and bandit-type strategies for resource allocation
3	China	FMLM: A federated meta-learning method for learning multimodal data features	Meng Yue, Weiyun Li.	Application in FMLM

				Training models that learn to learn for rapid adaptation with limited data.
4	Australia	Meta-learning Approaches for FewShot Learning: A Survey of Recent Advances	Hassan Gharoun, Fereshteh Momenifar, Fang Chen, Amir H. Gandomi.	Models that quickly adapt to new tasks with few examples. Key methods based on metrics, in-memory processing, and optimization. Current challenges in meta-learning.
5	Finland	Cross-System Categorization of Abnormal Traces in Microservice-Based Systems via Meta-Learning	Yuqing Wang, Mika V. Mäntylä, Serge Demeyer, Mutlu Beyazıt, Joanna Kisaakye, Jesse Nyyssölä.	Training with limited data. Quick adaptation to new systems. High accuracy in real-world scenarios
6	Japan	Character Expressions in Meta-Learning for Extremely Low Resource Language Speech Recognition	Rui Zhou, Akinori Ito, Takashi Nose.	Multilingual Meta-Learning MAML (Model-Agnostic Meta-Learning) for multilingual pre-training.
7	United States	Theoretical convergence of multistep model-agnostic meta-learning.	Kaiyi Ji, Junjie Yang, Yingbin Liang.	Convergence Analysis of Multi-Stage MAML Hyperparameters in MAML Advanced meta-learning variants.
8	India	Learning How to Learn: Meta Learning Approach to Improve Deep Learning	Dr. Ashish Kr. Chakraverti, Sugandha Chakraverti, Dr. Yashpal Singh.	The Path to Artificial General Intelligence (AGI) Meta-Reinforcement Learning
9	India	Meta-Learning Theory, Algorithms and Applications.	Lan Zou	Fundamentals of Meta-Learning. Key Approaches Practical Applications
10	Cuba	Una estrategia de MetaLearning para flujos genéricos de AutoML.	Loraine Monteagudo García.	Meta-Learning to Optimize AutoML. Intelligent Selection of Algorithm Workflows.
11	Peru	Un nuevo mecanismo de Regularización de algoritmos de aprendizaje con pocas muestras.	Marcell Llerena Paricahua.	Meta-learning for few-shot learning. Regularization mechanism to improve generalization.

Table 1 Details of Selected Records

4. Discussion

The following section presents a critical analysis of the main meta-learning approaches identified through the literature review. We sought to describe, in detail, their most representative characteristics, as well as to highlight their most significant limitations. Additionally, we discussed the applications in which they are most frequently implemented. All of this was presented with the intention of highlighting the existing and often overlooked gap between theoretical proposals and their actual practical application.

Optimization-based meta-learning (MAML and Reptile): The objective of meta learning is to optimize hyper parameters (such as initialization of neural networks) so that we can rapidly learn about a new task. MAML is a model agnostic method because it does not rely on the use of any particular architecture. It is based on optimization through two nested loops to find parameters that could be adjusted with one or a small number of gradient descent steps. One of the advantages of MAML is that it has good few shot adaptation ability. A major disadvantage of MAML however is the high computational cost: since the MAML uses second order gradients, it has the potential to lead to "task memorization" and thus has low accuracy in many situations (Hassan Gharoun, 2024). The Reptile algorithm is a variation of MAML where first order gradients are used: unlike MAML, where the inner loop is optimized explicitly, Reptile achieves this goal by finding target parameters that minimize the average Euclidean distance between initial and updated parameters when trained on each task. As a result, the computational load associated with MAML is reduced, while there is no longer a need for computing the Hessian matrix, however, it is possible that the convergence rate or stability of Reptile will be lower than those found in MAML. The process used by both Reptile and FOMAML to increase generalized task learning by maximizing internal gradient correlation also accounts for why they have relatively similar performance to one another; however, as demonstrated previously, Reptile outperforms FOMAML due to its increased efficiency (Llerena, 2021). Overall, methods such as Reptile and MAML work well in supervised machine learning problems with limited sample sizes (i.e. classification and/or regression in controlled environments); however, the question of whether these algorithms will perform effectively at scale is still being researched.

Metric networks (embedding, prototypical networks, and relationship networks): Methods based on metric spaces transform the data into a high-dimensional feature space (using a neural network) where the similarity between samples is evaluated based on distance. For example, prototypical networks calculate a prototype for each class as the average of the embeddings of the training samples, and classify a new sample based on the shortest distance to those prototypes (Rui Zhou, 2024). Their main strength is that they do not require an extensive fine-tuning step in the target phase; once the embedding space is trained, classification is fast and straightforward (Zou, 2022). However, they critically depend on the embedding space accurately reflecting the classes: if test tasks differ significantly from training tasks or there is insufficient data for certain classes, their performance declines. Furthermore, as they are geared toward classification with few classes, they are less effective when tasks involve complex combinations of algorithms or pipelines (as in heterogeneous AutoML). Llerena's review details these models (including Siamese networks and relational networks) as examples of model-based meta-learning.

AutoGOAL and heterogeneous AutoML systems: AutoGOAL is a free and open source auto-ml system that stands out due to the combination of multiple methodologies and tools (e.g., linear classifiers, nlp, neural networks etc.), to create machine learning pipelines. In his research paper Monteagudo Garcia (2020), added a meta-learning element to AutoGOAL: in this case, the author used an algorithm that selects the top k-pipelines based on dataset features for use in the optimization phase. The strength of this methodology lies in its capacity to take advantage of the heterogeneity present within AutoGOAL in order to share knowledge across different tasks. A major limitation of this methodology includes building a "knowledge base" of all possible pipeline configurations run against numerous datasets, selecting the value of k, and also the limitations of AutoGOAL itself, since AutoGOAL can only be applied to domains that are currently supported by AutoGOAL. Overall, AutoGOAL represents a practical way of applying meta-learning with positive experimental results however additional development would be needed in order to apply AutoGOAL to future applications.

Ranking and portfolio approaches: These methods learn to rank or select algorithms/pipelines based on their historical performance. A simple example is Average Ranking, which aggregates past rankings of algorithms on similar tasks to recommend the best ones for the new task. A key advantage is that it can operate without complex

meta-features, allowing costly AutoML processes to be initialized quickly (for example, by avoiding the calculation of many dataset metrics). Other methods such as Predictive Clustering Trees (PCTs) or CofiRank create learned ranking models. Modern AutoML systems (Auto-PyTorch, Auto-sklearn 2.0, ATM) incorporate these ideas through portfolios of offline-optimized configurations (Matthias Feurer, 2022). Their strengths include simplicity and fast initialization; however, they require large, representative prior datasets to be effective, and often do not adapt in real time to data that differs significantly from what they have seen. For example, Auto-sklearn 2.0 abandons meta-features in favor of a pre-built portfolio to ensure at least one good configuration. In general, these approaches are promising in simulated scenarios with many available training datasets, although their performance may decline in novel domains where the prior ranking does not generalize (Meng Yue, 2025).

Bayesian Optimization (SMBO): Although not a meta-learning method in the strict sense, Bayesian optimization (using sequential substitution models) is key in AutoML. Its strength lies in systematically exploring large hyperparameter spaces, balancing exploration and exploitation. For example, tools such as BOHB (Bayesian Optimization with Hyperband) (Kaiyi Ji J. Y., 2022) combine Bayesian models with multi-level evaluation budgets. However, Bayesian optimization can require many model evaluations, making it costly. As studies point out, hyperparameter search is generally slow, as the search space is vast and evaluating even a single algorithm can take hours. Furthermore, unless prior knowledge is incorporated, these methods restart the process from scratch for each new problem. In practical applications, Bayesian optimization is powerful for fine-tuning once good initial candidates are available, but its integration with meta-learning (for example, to transfer distributions to new problems) (Yuqing Wang, 2025) remains an active area of research.

To organize these aspects, Table 2 is presented, which briefly compares each method studied based on the literature:

Focus	**Strengths**	**Weaknesses**	**Requirements**
MAML	Model-agnostic algorithm, enabling rapid few-shot learning.	Costly (second-order gradients) and prone to overfitting; limited accuracy without further improvements.	Neural networks or differentiable models; similar tasks during training.
Reptile	First-order variant of MAML: more computationally efficient.	Lower performance than full MAML; may require more iterations to converge.	Similar to MAML; does not require Hessian computation.
Metric networks (embedding)	They do not require any adaptation steps during the training phase (immediate classification). They perform well in few-shot learning if the classes are consistent.	They rely heavily on the quality of the embedding; they are difficult to scale to very different domains.	Set of tasks organized by category (e.g., support and consultation for prototypes).
AutoGOAL (Heterogeneous AutoML)	Combines multiple ML libraries in heterogeneous pipelines. Meta-learning improved results (error reduced by half).	It requires the generation and storage of prior data streams; it depends on the AutoGOAL system (limited to the domains it supports).	Knowledge base of executed pipelines; integration with AutoGOAL.
Rankings / Portfolios	Quick initialization (no need for computationally intensive meta-features);	A representative metadata set is required; this is inflexible if new tasks	Performance history across multiple datasets; discretized algorithms.

	leverages historical performance data.	differ; the parameter k or portfolio size must be adjusted.	
Bayesian Optimization	Efficiently explores large spaces of hyperparameters (SMBO); supports evaluation budgets.	Very time-consuming (requires many evaluations); it usually restarts each time (without knowledge transfer).	Objective function (performance estimator); ability to test configurations.

Table 2 Approaches to Metacognition

After reviewing the various approaches summarized in Table 2, it is clear that meta-learning offers a wide range of strategies that can be adapted to different software development contexts. While algorithms such as MAML and Reptile stand out for their agility in rapid adaptation tasks within homogeneous environments, approaches like AutoGOAL demonstrate a more integrative and practical approach, aimed at automating model selection and configuration in complex systems. Meanwhile, metric networks and ranking systems or portfolios optimize tasks where representative prior knowledge and consistent class structures are available.

However, beyond their theoretical formulation, the true value of these approaches lies in their practical applications. For example, MetaLog (Chenyangguang Zhang, 2024) was developed for anomaly detection in distributed system logs, outperforming supervised models trained on the full dataset thanks to its ability to learn from previous tasks. With just 1% of labeled data, MetaLog achieved an F1 score above 80% and outperformed traditional transfer learning approaches by 20 percentage points (Chenyangguang Zhang, 2024).

Similarly, TraFaultDia (Yuqing Wang, 2025) is applied in the context of microservices, a highly dynamic environment where faults can arise at different points in the system. This approach is able to diagnose faults using only 10 examples per class, achieving an accuracy of over 93% within the same system and 92% when the model is transferred to a different one. This demonstrates highly effective generalization, even under variable conditions, which is crucial for engineering applications in continuous production.

Another relevant example documents an innovative method applied to the domain of speech recognition in low-resource languages (Rui Zhou, 2024). In their work, they pre-train a model using meta-learning with speech data from ten different languages, and then fine-tune it with just fifteen minutes of audio from the target language using an alphabetic unification strategy based on phonetic rules of the Latin alphabet.

This technique succeeded in reducing the character error rate (CER) to less than 30%, demonstrating a substantial improvement in tasks with limited data (Rui Zhou, 2024). Although this case comes from speech processing, its approach is directly relevant to embedded software environments or adaptive systems in low-resource languages, as it shows how meta-learning can generate efficient models with extremely limited data.

Finally, Auto-sklearn 2.0 (Feurer et al., 2022) demonstrates how meta-learning can be integrated into robust AutoML frameworks to accelerate the search for optimal models. By eliminating the need to extract meta-features and incorporating portfolio-based selection strategies, this approach reduced the relative error by 78% compared to its previous version, with a computational budget of just 10 minutes. Its direct applicability to software engineering processes where time and automation are critical reinforces its relevance.

Taken together, these implementations demonstrate that, while meta-learning algorithms have complex foundations, their adaptation to specific tasks within software development is feasible and highly beneficial. Approaches that combine data efficiency, robustness against heterogeneous tasks, and flexible structures—such as TraFaultDia or Auto-sklearn 2.0—emerge as the most promising for future experimental applications in simulated environments with variable conditions.

The reviewed approaches and applications show how meta-learning can be used to manage changing conditions (with little data) using software development as a model. This review is not only to identify the best models for

addressing changing scenarios with limited data but it provides the research base needed to develop an empirical application that will experimentally evaluate whether one of these approaches can learn from past experience in a simulated environment in order to make decisions that maximize performance under new/dynamic conditions. Ultimately, the goal is to provide a real-world evaluation of the applicability and efficacy of the models identified in this study.

Refrences

1. Carrizo, D. (november 2018). Estructuras metodológicas de revisiones sistemáticas de literatura en Ingeniería de Software: un estudio de mapeo sistemático. Retrieved from scielo: https://www.scielo.cl/scielo.php?script=sci_arttext&pid=S0718-33052018000500045.
2. Chenyangguang Zhang, T. J. (April 12, 2024). MetaLog: Generalizable Cross-System Anomaly Detection from Logs with Meta-Learning. Retrieved from ACM Digital Library: https://dl.acm.org/doi/10.1145/3597503.3639205.
3. Dr. Ashish Kr. Chakraverti, S. C. (2020). Learning How to Learn: Meta Learning Approach to Improve Deep Learning. Approach to Improve Deep Learning.
4. Hassan Gharoun, F. M. (July 25, 2024). Meta-learning Approaches for Few-Shot Learning: A Survey of Recent Advances. Retrieved from ACM Digital Library: https://dl.acm.org/doi/10.1145/3659943.
5. Kaiyi Ji, J. Y. (January 1, 2022). Theoretical convergence of multi-step model-agnostic metalearning. Retrieved from ACM Digital Library: https://dl.acm.org/doi/10.5555/3586589.3586618.
6. Llerena, M. (2021). Un nuevo mecanismo de regularización para algoritmos de aprendizaje con pocas muestras. Arequipa.
7. Matthias Feurer, K. E. (January 1, 2022). Auto-sklearn 2.0: hands-free AutoML via metalearning. Retrieved from ACM Digital Library: https://dl.acm.org/doi/10.5555/3586589.3586850.
8. Meng Yue, W. L. (Mayo 27, 2025). FMLM: A federated meta-learning method for learning multimodal data features. Retrieved from ACM Digital Library: https://dl.acm.org/doi/10.1145/3727353.3727385.
9. Monteagudo, L. (2019). Una estrategia de Meta -Learnig para flujos genéricos de AutoML. La Habana.
10. Rui Zhou, A. I. (June 7, 2024). Character Expressions in Meta-Learning for Extremely Low Resource Language Speech Recognition. Retrieved from ACM Digital Library: https://dl.acm.org/doi/10.1145/3651671.3651730.
11. Yuqing Wang, M. V. (June 19, 2025). Cross-System Categorization of Abnormal Traces in Microservice-Based Systems via Meta-Learning. Retrieved from ACM Digital Library: https://dl.acm.org/doi/10.1145/3715742.
12. Zou, L. (2022). Meta-Learning Theory, Algorithms and Applications. Project Manager.