

Article

Not peer-reviewed version

---

# Evolutionary Ensembles of Artificial Agents for Enhanced Mineralogical Analysis

---

[Paolo Dell'Aversana](#) \*

Posted Date: 30 October 2024

doi: 10.20944/preprints202410.2434.v1

Keywords: Deep Learning; Mineral,; Classification; Thin section; Machine Learning; Ensemble



Preprints.org is a free multidiscipline platform providing preprint service that is dedicated to making early versions of research outputs permanently available and citable. Preprints posted at Preprints.org appear in Web of Science, Crossref, Google Scholar, Scilit, Europe PMC.

Copyright: This is an open access article distributed under the Creative Commons Attribution License which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

Article

# Evolutionary Ensembles of Artificial Agents for Enhanced Mineralogical Analysis

Paolo Dell'Aversana

Eni S.p.A; paolo.dell'avversana@eni.com

**Abstract:** This paper presents a novel machine learning framework that applies evolutionary ensembles of artificial agents to mineralogical analysis and classification. The approach is based on hybridization techniques that combine diverse machine learning algorithms, creating large and effective communities of agents. These progressively mature and improve through crossover, hybridization, and selection, addressing the challenges of mineral recognition and classification from thin section images. By combining multiple machine learning techniques, the ensemble of agents autonomously improves by evolving to adapt and enhance its ability to identify mineral species and classify different types of alterations. We detail the method, provide examples using synthetic and real data, and explore the potential to improve mineralogical analysis workflows through this dynamic, self-improving system.

**Keywords:** deep learning; mineral; classification; thin section; machine learning; ensemble

## 1. Introduction

In mineralogy, one of the central challenges is the accurate identification and classification of mineral species from thin section images under a petrographic microscope. These images can present significant variability due to factors such as optical properties, complex textures, alterations and overlaps between mineral phases. Automated methods are increasingly essential to complement traditional manual classification, which can be inaccurate and time-consuming [1–12]. Deep learning models like Convolutional Neural Networks (CNNs), can automatically extract relevant features from images that may not be immediately obvious to human observers. In mineralogy, this means the models can learn from large datasets of labeled mineral images to recognize subtle differences in texture, color, and patterns that are characteristic of different mineral species. Furthermore, machine learning algorithms can generalize across variations in lighting conditions, orientation, and image quality, leading to more consistent classification results than manual methods. For example, CNNs can take thin section images as input and learn the unique visual properties of each mineral species. This approach eliminates the need for manually crafted feature extraction, which is prone to error and may not capture the full complexity of mineral textures. Of course, while neural networks are powerful tools for image-based mineral classification, they represent just one subset of the broader field of machine learning [13]. There are many other algorithms that can be equally useful or even superior for certain aspects of mineralogical analysis, especially when combined in cooperative ensembles. Each algorithm has its strengths and limitations. By forming cooperative ensembles, we can combine the advantages of multiple algorithms to create a more versatile and reliable mineral classification system. In such ensembles, different algorithms are tasked with addressing the same goal—mineral classification—but from different perspectives.

In this framework, this paper proposes an innovative approach by forming “evolutionary communities” of artificial agents. These agents cooperate, compete, and evolve, employing a diverse range of machine learning techniques such as convolutional neural networks, decision trees, random forest, naïve bayes, support vector machine and a variety of other methods. The agents, through continuous interactions and hybridizations, are capable to evolve for improving their performance over time. Evolutionary algorithms (EAs) [14–21] have been explored in broader geological

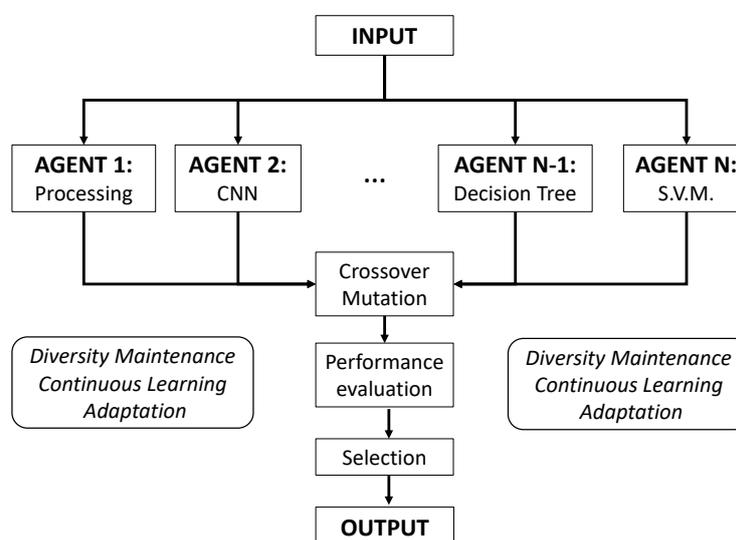
applications such as the optimization of drilling routes and mineral deposit predictions. In these contexts, EAs allow for flexibility in problem-solving, but they have not been widely applied to the direct classification of mineral images. Combining evolutionary strategies with an ensemble machine learning approach holds the promise of creating a flexible, continuously improving system for mineral identification. Indeed, the introduction of evolutionary agent communities in mineralogy could offer significant advantages. First, agents evolve and adapt to new mineralogical datasets, making the system increasingly robust over time. Second, by combining neural networks, decision trees, and other methods, agents can improve performance in recognizing complex mineral textures and structures. Third, the dynamic nature of the system ensures adaptability to new imaging conditions, such as variations in illumination or sample preparation techniques. In the following sections, we will demonstrate how our approach can effectively address all these challenges. First, we will introduce the details of the methodology and the workflow. Then, we will apply the method to simulated data. Finally, we will discuss applications to real data through increasingly complex classification tests.

## 2. Materials and Methods

### 2.1. The Structure of the Artificial Ensemble

Our evolutionary ensemble is composed of numerous agents, each specialized in different machine learning techniques and methods for data analysis (Figure 1). These include:

- Pre-processing algorithms and techniques for feature engineering
- Convolutional Neural Networks (CNNs) for pattern recognition in mineralogical textures.
- Decision Trees for distinguishing mineral species based on characteristic features.
- Additional Machine Learning algorithms, such as Random Forest, Recurrent Neural Networks, Support Vector Machine and so forth.
- Probabilistic Models for handling uncertainties in image data and classification tasks.



**Figure 1.** key blocks of the Structure of the Artificial Evolutionary Community.

The ensemble is divided into subgroups, with some agents focused on data pre-processing and analysis (such data normalization, image segmentation for identifying boundaries between different minerals in thin section images, feature engineering, and so forth), while others focus on mineral identification and classification, based on mineral properties usable as features (shape, colors, optical properties etc.).

## 2.2. Evolutionary Mechanisms

The agents evolve through several mechanisms including cross-over, mutation and selection. In crossover, the agents exchange parts of their learned features and model architectures, combining strengths from CNNs and other machine learning algorithms, for example, to improve mineral recognition. Through continuous mutation, random variations are introduced in agents, modifying network architectures, or adding new decision-making criteria, and/or changing key hyper-parameters. Finally, agents are evaluated based on their performance on a range of tasks, including image segmentation accuracy, mineral classification precision, and texture differentiation. The top-performing agents are selected for reproduction, while the weaker agents are eliminated.

## 2.3. Diversity Maintenance

Maintaining a diverse set of agents is crucial to avoid overfitting to specific mineral species or textures. This is achieved through several criteria. With speciation, agents are clustered based on their model architectures and training histories, ensuring that different methods explore distinct solutions. Using a criterion of "homogeneity penalization", if agents within the community become too similar, penalties are applied to reduce the likelihood of further replication, encouraging the exploration of novel approaches.

## 2.4. Continuous Learning and Adaptation

Commonly, when analyzing a large data base of thin section images, new mineralogical datasets are periodically introduced into the system, requiring agents to adapt. This allows the ensemble of agents to handle changing conditions in the classification of minerals and to accommodate new challenges: these can be the identification of rare minerals or the differentiation of minerals with similar optical properties. Our ensemble framework is designed to incorporate new data into the existing database, enabling efficient preprocessing, feature extraction, and integration of variable information into the workflow.

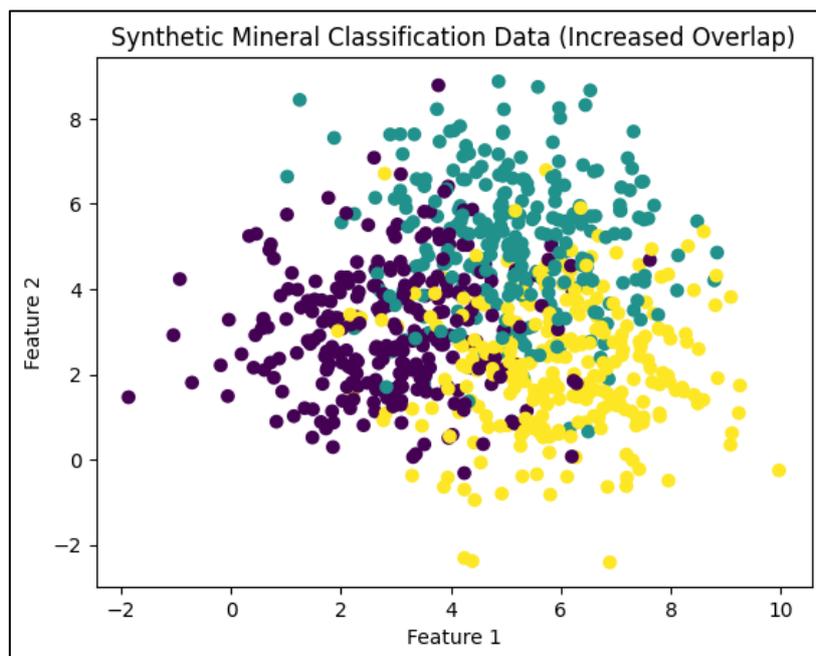
In the following two sections, we will show all these aspects through illustrative tests characterized by an increasing level of complexity. First, we will focus on methodological aspects using simulated data. Then, we will move to classification and pattern recognition tasks using real data sets.

# 3. Synthetic Data Example

In the first (synthetic) test, our evolutionary approach was employed to train a diverse set of machine learning models on a simulated classification problem characterized by highly overlapping class distributions in the feature space. The primary goal was to enhance classification performance over generations by evolving a population of agents, which included both classical machine learning models and deep neural networks. The neural network models, or "NN agents," underwent selection, crossover, and mutation to simulate an evolutionary process, allowing them to adapt their architecture (number of hidden layers, number of neurons, learning rate and so forth) to the challenging feature space. Classical machine learning models, like SVM, Random Forest Classifier, or AdaBoost Classifier, do not have layers like neural networks (e.g., CNNs or RNNs). Consequently, crossover and mutation functions are only applied between neural network-based models. The other machine learning algorithms are allowed to mute in terms of their specific hyperparameters.

## 3.1. Dataset Creation

A synthetic dataset was generated to simulate a three-class classification problem, where the data for each class was sampled from overlapping Gaussian distributions in a 2D feature space. The dataset consisted of 1,000 samples, with features (simulated percentage of different types of oxides) from each class exhibiting substantial overlap, thereby increasing the complexity of the classification task (Figure 2). This setup created a challenging scenario where the boundaries between classes were difficult to distinguish, testing the robustness of the models.



**Figure 2.** Scatter plot of three different classes highlighted with different colors in a 2-feature space.

### 3.2. Agents

The ensemble of agents consisted of the following models with a typical architecture including:

- A set of Convolutional Neural Networks (CNN) including: Input Layer, Convolutional Layers, ReLU Activation function, Pooling Layers (MaxPooling), all repeated multiple times. Furthermore the CNN models include: Flattening Layer, Fully Connected Layers, Softmax Output Layer for classification.
- A set of Residual Neural Networks (ResNet): A neural network architecture that includes residual connections, which allow the model to retain information across layers and prevent gradient vanishing during training.
- A set of Decision Trees: A non-parametric, tree-based model for classification.
- A set of Random Forests (RF): An ensemble method based on Decision Trees, which averages multiple trees to improve generalization.
- A set of Support Vector Machines (SVM): A kernel-based classifier aimed at finding a decision boundary that maximizes the margin between classes.
- A set of Naive Bayes (NB): A probabilistic classifier based on Bayes' theorem, assuming feature independence.
- A set of Adaptive Boosting (AdaBoost): An ensemble method that builds a strong classifier by sequentially training weaker classifiers.

### 3.3. Evolutionary Algorithm

The evolutionary process was implemented over 20 generations with a population size of 20 agents. In each generation, agents were trained on the training set and evaluated on a test validation set using classification accuracy as the fitness measure. Neural networks (CNN and ResNet) were trained using backpropagation over multiple epochs, while the other machine learning models were trained using their respective learning algorithms. The top-performing agents from each generation were selected for reproduction. Selection was based on accuracy, with the top 50% of agents retained for crossover. For neural networks, a crossover mechanism was employed where the internal hyperparameters (layers, number of neurons for each layer, learning rate, drop-out, etc.) of two parent agents were swapped. This genetic operation was applied to classical models (like Decision Tree or Random Forest), acting on their specific hyper-parameters, but without any cross over mechanisms with neural networks. The next generation was formed by combining the top-

performing agents from the previous generation with newly created agents from the crossover and mutation process. This ensured a mix of exploitation (using the best solutions) and exploration (introducing diversity through mutation).

### 3.4. Results

The evolutionary training led to a steady improvement in average accuracy over generations. The performance of the models was tracked, showing the adaptive behavior of the agents as they evolved to better handle the challenging feature space. After 20 generations, the best-performing agent was evaluated on the test set, and its classification results were visualized. A classification report was generated to provide a detailed assessment of the final model's performance. Figure 3 shows the increasing trend of the average validation accuracy vs. generation ID number. When this trend reaches a certain level of stationarity, the evolutionary process is stopped. At that point, the most evolved hybrid algorithm is selected and used for classification. In this specific simulation, the synthetic data were classified with a final validation accuracy ranging between 75% and 85%. In the next test, we will see how accuracy can be improved.

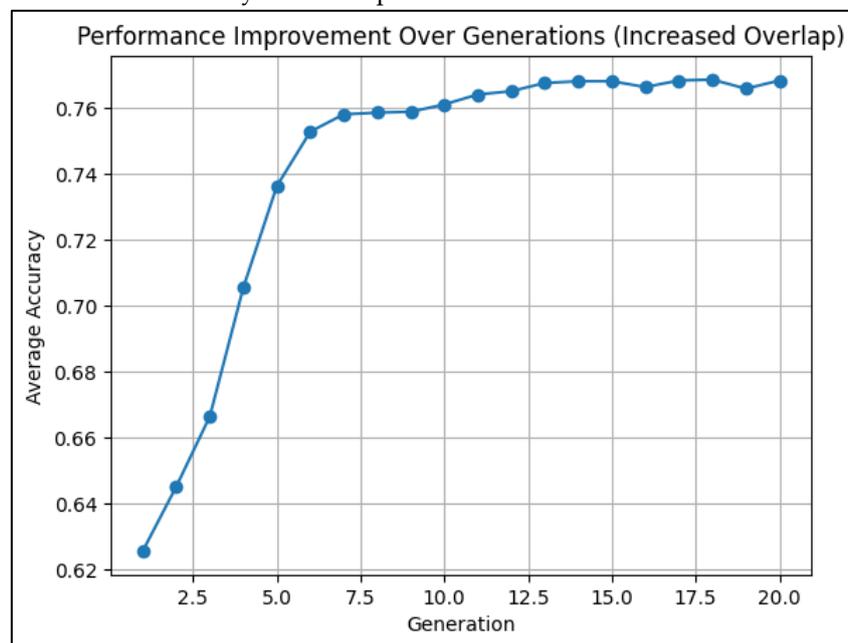


Figure 3. Average validation accuracy vs. generation ID number.

## 4. Real Data Example

### 4.1. Dataset

To improve the effectiveness of our ensemble approach, we allowed more intensive hybridization between neural network agents and classical machine learning agents in the evolutionary training code. We implemented crossover and mutation functions that apply to, and between, both types of agents. For example, for that purpose we can create a hybrid model by combining the output of a neural network with the decision-making capabilities of a classical Decision Tree algorithm.

We tested this improved approach directly on a real data set. This consists of hundreds of thin section images obtained through a petrographic microscope, including samples from igneous rocks. The dataset contains high-resolution images, each labelled with expert annotations identifying minerals as well as their textural relationships (e.g., grain boundaries, intergrowths) or possible alterations (such as hydrothermal alterations). First, we applied our method of evolutionary ensembles of artificial agents to classify different types of alterations. Figure 4 shows some examples of the different types of alterations here considered.



**Figure 4.** Left image: Olivine crystal with Bowlingite in fractures. Image at N//, 10x (long side = 2mm). Central image: Olivine crystal with Iddingsite at the edge. Image at NX, 2x (long side = 7mm). Right image: Deeply sericitized feldspar crystals in hydrothermally altered dacite. Image at NX, 2x (long side = 7mm).

These high-resolution images were analyzed to classify and better understand the mineralogical changes associated with specific alterations [22–27]. We used images of fine-grained Bowlingite (commonly referred to as saponite), where alterations result in the addition of iron (Fe) and water (H<sub>2</sub>O) and the loss of magnesium (Mg). These transformations are significant as they indicate hydrothermal activity that could signal the presence of valuable mineral deposits. Furthermore, we examined Iddingsite, which forms because of the alteration of olivine in highly oxidizing environments under moderate temperatures and low pressures. This process, known as Pseudomorphism, involves a chemical transformation while the mineral retains its original form. The detection of Iddingsite in our samples can point towards environments with potentially exploitable iron deposits. Lastly, we analyzed images showing the alteration of feldspars into Sericite, a fine-grained aggregate of muscovite, illite, and paragonite. Sericite forms through the circulation of hydrothermal fluids and can indicate proximity to ore deposits, such as those containing gold or silver. The distinct silky texture of Sericite, often visible in these altered rocks, provides crucial insight into mineral-rich zones. By leveraging these images, our tests focus on classifying and analyzing these alterations, which are key indicators for guiding mining exploration efforts.

#### 4.2. Workflow

The following is a breakdown of the main workflow steps, focusing on the testing and evaluation aspects.

- The test begins by loading images from a specified directory structure. Each subdirectory corresponds to a different class of mineral images. The images are resized and normalized before being stored in arrays.
- The class images and their labels are encoded into numerical values, which allows them to be used in machine learning algorithms.
- The dataset is divided into training and validation sets. A specified percentage of the data (80% by default) is used for training the models, while the remaining data (20%) is reserved for validating their performance.
- The testing and evaluation of the machine learning models happen through the following steps:
  - Agent Creation: Two types of agents are created: a Convolutional Neural Network (CNN) and a Decision Tree classifier.
  - Evaluation Function: an evaluation function is defined, which computes the performance of the agents.
  - CNN Evaluation: the CNN is trained on the training set, and its accuracy is calculated on the validation set which returns the accuracy.
  - Decision Tree Evaluation: The Decision Tree is trained, and its accuracy is evaluated separately.
- Evolutionary Training Loop: the core of the evolutionary algorithm involves multiple generations. For each generation, all agents are evaluated, and their scores (accuracies) are collected.
- The average score across all agents is computed to assess the overall performance of the current generation.

- A selection process chooses the top-performing agents to carry over to the next generation.
- Crossover and mutation operations are applied to create new agents. Crossover combines architectural features of two agents, while mutation introduces random variations in each agent to help explore the solution space for optimal architectural configurations.
- The best-performing agent from each generation is stored for later analysis.
- Final Classification Results: after the evolutionary training process, the best evolved agent is used to make predictions on the validation set.
- A scatter plot visualizes the classification results, showing how the model categorized the validation images.
- A classification report is generated, which includes metrics like precision, recall, F1-score, and overall accuracy, providing a detailed evaluation of the model's performance on the validation data.

#### 4.3. Results

A final Evolutionary algorithm model was created by training all the above agents over N generations, with a fitness evaluation (based on accuracy), selection of the best-performing agents, crossover between models, and mutation of their architecture. The application of "elitism criteria" ensured the top agents are retained across generations. Elitism in evolutionary algorithms refers to a strategy where the best-performing individuals (or agents) in a population are preserved unchanged for the next generation. This ensures that the top solutions, which have the highest fitness or performance, are not lost during the crossover or mutation processes. This hybrid architecture leverages the strengths of both CNN (for image processing) and Decision Trees (for structured, interpretable decisions) while improving performance through genetic evolution.

Studying the validation accuracy trend is crucial because it indicates how well the hybrid models are able in classifying unseen data. Indeed, our evolutionary approach aims to improve this accuracy over successive generations, ultimately yielding a robust model for classification of the different types of alterations. Figure 5 shows the average score with increasing generation number in this test. In the evolutionary training process, the average score for a generation represents the mean accuracy of all agents in the population during that generation. We can see that validation accuracy reached 100% in less than 10 generations, indicating perfect classification results.

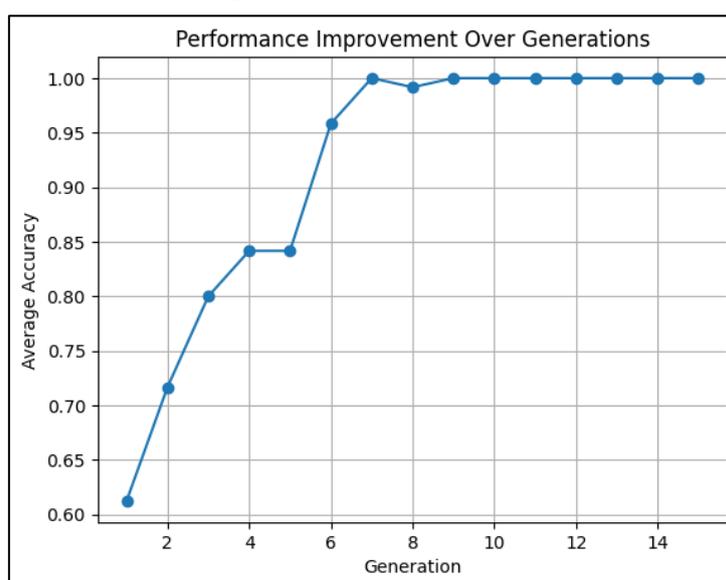
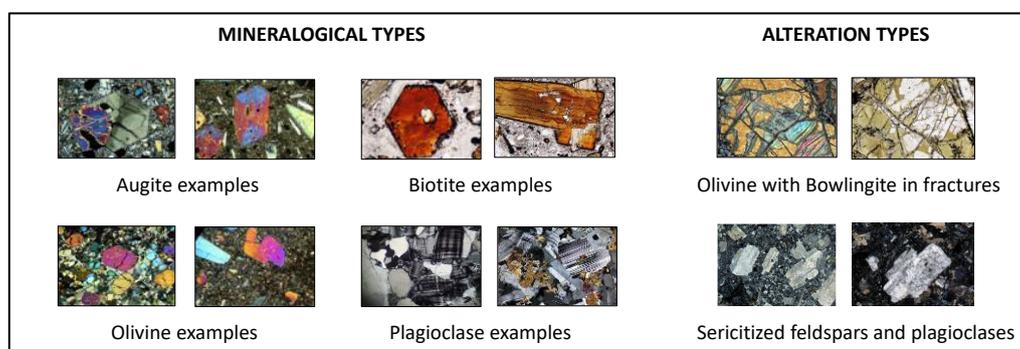


Figure 5. Average validation accuracy vs. generation ID number.

#### 5. A More Complex Test

To test the effectiveness of our evolutionary approach, we conducted a more comprehensive and challenging experiment on real data. The objective was not only to classify different types of

mineralogical alterations, such as Bowlingite, Iddingsite, and Sericite, but also to simultaneously classify various mineral classes (Figure 6). This dual-task problem increases the complexity of the classification test and provides a more realistic and robust evaluation of our approach in mining exploration scenarios, where mineralogical and alteration patterns often co-exist and influence each other.



**Figure 6.** Examples of thin sections used in the simultaneous classification of mineral species and alteration types.

### 5.1. Hybrid Model Architecture

In this enhanced version of the experiment, we expanded the hybrid machine learning architecture by incorporating Convolutional Neural Networks (CNNs), Decision Trees, and Support Vector Machines (SVMs) into the same algorithmic ensemble. Each model type brings unique strengths. CNNs have the objective to identify subtle patterns in the microstructures of altered minerals. On the other side, Decision Trees are adept at handling non-linear decision boundaries, making them useful for feature-based mineral classification tasks. Finally, SVMs are addressed to provide robust classification for high-dimensional datasets, and in this case, offer a strong complement to the CNN's ability to handle image data, improving classification accuracy of mineral classes.

All the images were preprocessed by normalizing pixel values and resizing to a uniform dimension to ensure compatibility across all models. Labels corresponding to both the type of alteration and the mineral class were assigned to each image. The evolutionary algorithm was initialized with a population of hybrid agents. Each agent was a combination of a CNN, Decision Tree, and SVM model, randomly configured with different architectures and hyperparameters.

CNN models were initialized with varying convolutional layers, pooling layers, and dense layers to extract image features, while Decision Trees and SVM models were initialized with different tree-depth levels and kernel types, respectively. Each hybrid agent in the population was evaluated on the classification task using both training and validation datasets. The CNN component processed the image data, while the Decision Tree and SVM components worked on derived features. The overall accuracy of each agent was recorded.

### 5.2. Hybrid Workflow

Based on performance, the top-performing agents were selected for the next generation. Elitism was applied by carrying over the best few agents directly without alteration to ensure that high-performing solutions were not lost due to stochastic variation. For the remaining agents, crossover operations were applied. For CNNs, this involved swapping convolutional and dense layers between two agents. For Decision Trees and SVMs, the hyperparameters such as tree-depth or kernel type were exchanged. This crossover allowed the model to explore new combinations of architectures and algorithms.

Random mutations were introduced to further diversify the population. For CNNs, this might involve changing the number of filters in a convolutional layer or adding/removing layers. For Decision Trees and SVMs, it might involve altering tree-depth or kernel parameters. These mutations

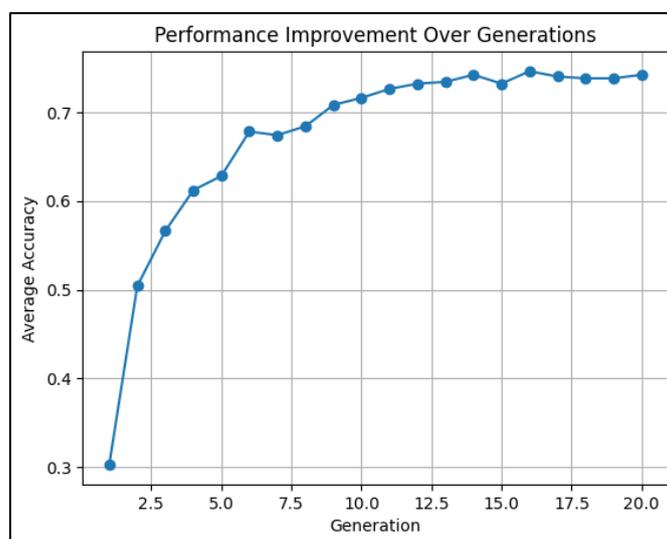
helped avoid premature convergence and introduced new structural configurations to improve generalization. The hybrid models were trained using an evolutionary loop that spanned 15-20 generations. With each generation, the population of agents evolved by gradually improving their classification performance through selection, crossover, and mutation. Throughout the evolutionary process, the performance of the models was tracked, with accuracy improving as the generations progressed.

### 5.3. Final Model and Evaluation

After several generations, the best-performing hybrid model was selected as the final classifier. This model effectively combined CNN's strength in feature extraction from images with Decision Trees and SVM's strengths in classifying more structured, non-image features. The final model was tested on a reserved validation set of images, and its performance was measured using standard metrics such as accuracy, precision, recall, and F1-score. The model achieved high accuracy in both classifying the type of alteration and the mineral class.

### 5.4. Results and Visualization:

The evolution of performance was visualized over the generations, showing a progressive increase in accuracy until the model stabilized around an average accuracy of about 75% (Figure 7). This time, the classification results was not perfect as in the previous test, because the objective was more challenging. In fact, the final hybrid model provided not only the classification of mineral species, but also clear identification of complex alteration patterns within the data. We think that this result can be still improved by increasing the size of training data, however it is encouraging by itself. It illustrates the effectiveness of using an evolutionary algorithm to train a hybrid ensemble of CNNs, Decision Trees, and SVMs, ultimately producing a solution that outperformed traditional, single-model approaches.



**Figure 7.** Average validation accuracy vs. generation ID number.

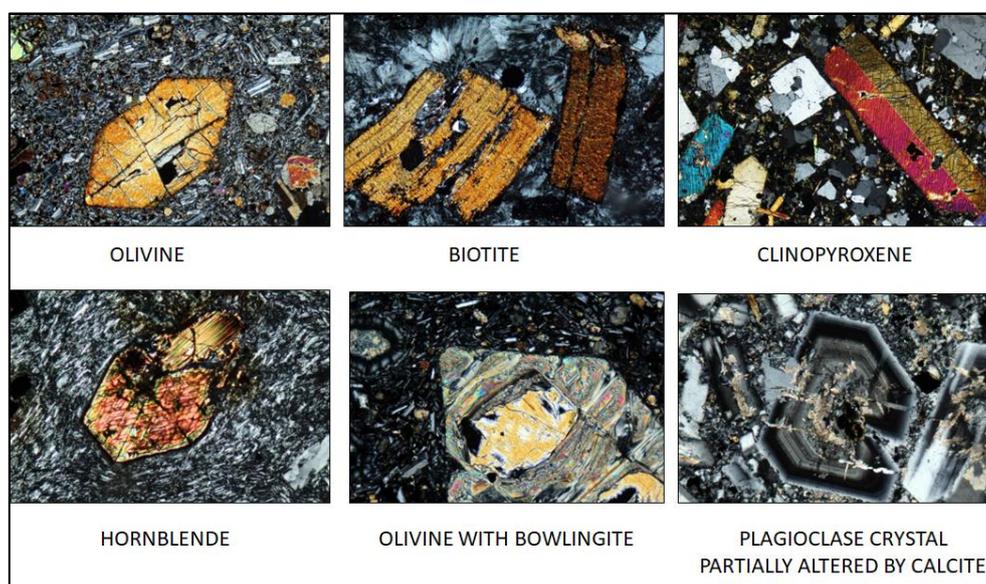
## 6. Evolutive Ensemble of Self-Aware Deep Neural Networks

Additional improvements can be obtained by combining more advanced models in the same machine learning ensembles, using self-aware deep neural networks recently introduced [28–31]. We recall that self-aware deep neural networks are designed to monitor their own performance and adapt in real-time, mimicking certain cognitive functions like error detection and adaptive learning. In ensemble machine learning, this self-awareness deep learning (briefly SAL) methodology allows models to dynamically adjust their own architecture without any human intervention, through continuous self-monitoring of performances. These networks can evolve following the same

evolutionary workflow described earlier, by selecting and hybridizing the best models, making the ensemble more robust. The additional advantage is that, using the SAL approach, allows further options to improve individual agent even before applying cross-over, mutation and hybridization with the other agents. This leads to more flexible, accurate, and resilient systems for complex tasks across various domains. To verify the effectiveness of such an evolutive ensemble of self-aware deep neural networks, we performed additional tests on the same data set described earlier (JPEG images of mineral thin sections). The improved version of our code implements an evolutionary training approach with self-aware mechanisms to adapt agents during the learning process. At the same time, it combines Naïve Bayes, Decision Tree, Random Forest, Adaptive Boosting and SVM agents in a hybrid system. Self-reflection and self-improvement mechanisms are introduced to monitor the performance of each individual agent after evaluation. If an agent's accuracy falls below a defined threshold (e.g., accuracy = 60%), it triggers the necessary adjustments. Examples of adjustments are the following:

- CNN agents increase their dense layer units, as well as the number of neurons.
- Decision Trees adjust their max depth.
- SVMs modify their regularization parameter.

In summary, this SAL approach enables dynamic improvement by allowing the agents to evolve not only via evolutionary techniques (crossover, mutation), but also by self-tuning their internal architecture based on continuous feedback about agent's performance. To make the test particularly challenging, we used a dataset of images of thin sections of minerals belonging to various crystalline species related to igneous, sedimentary, and metamorphic rocks, also including many examples of alterations (some illustrative examples are provided in Figure 8). In total, we used about 400 images divided into ten different classes (in terms of crystalline species and types of alterations). The results of the test were very encouraging and are summarized in Table 1. We can see that classification performance indexes of the ensemble is relatively high, compared to the other individual agents.



**Figure 8.** Examples of thin sections used for the classification test with evolutive ensemble of self-aware deep neural networks.

**Table 1.** Comparison of performance indexes of SAL-Evolutive ensemble vs. other individual machine learning methods.

Classifiers	Area Under Curve (AUC)	Classification Accuracy (CA)	F1	Precision	Recall
<b>Evolutionary SAL-ensemble</b>	<b>0.951</b>	<b>0.799</b>	<b>0.799</b>	<b>0.798</b>	<b>0.799</b>
Decision Tree	0.74	0.611	0.603	0.625	0.611
Support Vector Machine	0.811	0.758	0.751	0.691	0.758
Random Forest	0.866	0.695	0.681	0.691	0.695
Naive Bayes	0.833	0.495	0.477	0.7	0.495
Adaptive Boosting	0.65	0.505	0.504	0.503	0.505

## 7. Conclusions

The evolutionary agent framework discussed in this paper offers significant improvements in mineral recognition and classification compared to traditional machine learning models. By continuously adapting to new mineralogical datasets, the entire machine learning ensemble evolves to handle challenging cases, such as identifying minerals or alterations with similar optical characteristics. A notable advantage of this approach is its ability to handle complex textures in thin section images. The agents' hybridization of techniques allows them to distinguish between similar textures, which can be difficult for traditional methods to classify accurately. This framework is highly scalable, making it suitable for large-scale mineralogical projects, such as automated analysis of big databases of thin sections across different rock types.

Additionally, we have integrated Self-Aware deep neural networks into the framework, further enhancing the evolutionary machine learning approach. These self-aware mechanisms allow the agents to adapt their architecture or hyperparameters based on their performance, empowering the system to improve over time. Tests on real-world datasets, focused on the classification of mineral images (thin sections), have confirmed the effectiveness of this enhanced approach.

Despite its advantages, challenges remain, particularly in terms of computational resource management. The evolutionary process, involving large agent populations and continuous adaptation, can be resource-intensive. Additionally, ensuring that the agents remain diverse enough to explore novel solutions without converging prematurely is a key issue to be addressed. Future work will explore optimizing the evolutionary process and testing this approach on even larger and more diverse mineralogical datasets.

To conclude, we remark that the ensemble-evolutionary method introduced in this paper can be applied in many other real-world scenarios, and not exclusively in mineral analysis or in mining exploration. The combination of different machine learning models into a single, evolving hybrid ensemble opens new avenues for handling complex, multi-faceted classification and prediction tasks.

**Author Contributions:** Paolo Dell'Aversana: Conceptualization, methodology, software; validation, formal analysis, investigation, writing, review and editing.

**Funding:** This research received no external funding.

**Data Availability Statement:** All the images (microscope mineral thin sections) discussed and shown in this paper are in the public domain and available from the web site of Alessandro Da Mommio, under CC (Creative Commons Licenses) conditions. Link: <http://www.alexstrekeisen.it/index.php>, accessed on 21 July 2023.

**Conflicts of Interest:** The author declares no conflicts of interest.

## References

1. Porwal, A.; Carranza, E.; Hale, M. Artificial neural networks for mineral-potential mapping: A case study from Aravalli Province, Western India. *Nat. Resour. Res.* 2003, 12, 155–171.
2. Karimpouli, S.; Tahmasebi, P.; Saenger, E.H. Coal cleat/fracture segmentation using convolutional neural networks. *Nat. Resour. Res.* 2020, 29, 1675–1685.
3. Juliani, C.; Ellefmo, S.L. Prospectivity Mapping of Mineral Deposits in Northern Norway Using Radial Basis Function Neural Networks. *Minerals* 2019, 9, 131.

4. Sun, T.; Li, H.; Wu, K.; Chen, F.; Hu, Z. Data-driven predictive modelling of mineral prospectivity using machine learning and deep learning methods: A case study from southern Jiangxi province, China. *Minerals* 2020, 10, 102.
5. Izadi, H.; Sadri, J.; Bayati, M. An intelligent system for mineral identification in thin sections based on a cascade approach. *Comput. Geosci.* 2017, 99, 37–49.
6. Dell'Aversana, P. An Integrated Deep Learning Framework for Classification of Mineral Thin Sections and Other Geo-Data, a Tutorial. *Minerals* 2023, 13, 584. <https://doi.org/10.3390/min13050584>.
7. Dell'Aversana, P., Enhancing Deep Learning and Computer Image Analysis in Petrography through Artificial Self-Awareness Mechanisms. *Minerals* 2024, 14, 247.
8. She, Y.; Wang, H.; Zhang, X.; Qian, W. Mineral identification based on machine learning for mineral resources exploration. *J. Appl. Geophys.* 2019, 168, 68–77. [Google Scholar].
9. Liu, K.; Liu, J.; Wang, K.; Wang, Y.; Ma, Y. Deep learning-based mineral classification in thin sections using convolutional neural network. *Minerals* 2020, 10, 1096. [Google Scholar].
10. Slipek, B.; Młynarczyk, M. Application of pattern recognition methods to automatic identification of microscopic images of rocks registered under different polarization and lighting conditions. *Geol. Geophys. Environ.* 2013, 39, 373.
11. Aligholi, S.; Khajavi, R.; Razmara, M. Automated mineral identification algorithm using optical properties of crystals. *Comput. Geosci.* 2015, 85, 175–183.
12. Zhang, Y.; Li, M.; Han, S.; Ren, Q.; Shi, J. Intelligent Identification for Rock-Mineral Microscopic Images Using Ensemble Machine Learning Algorithms. *Sensors* 2019, 18, 3914.
13. Raschka, S.; Mirjalili, V. *Python Machine Learning: Machine Learning and Deep Learning with Python, Scikit-Learn, and TensorFlow*, 2nd ed.; Packt Publishing Ltd.: Birmingham, UK, 2017. [Google Scholar].
14. Arabas, J. a., Michalewicz, Z., and Mulawka, J. GAVaPS - a Genetic Algorithm with Varying Population Size. *Proc. of the First IEEE Conference on Evolutionary Computation* pp. 73-78. 1994.
15. Brest, J., Greiner, S., Boskovic, B., Mernik, M. and Zumer, V., 2006. Self-adapting control parameters in differential evolution: A comparative study on numerical benchmark problems. *IEEE transactions on evolutionary computation*, 10(6), pp.646-657.
16. Deb, K. and Sundar, J., 2006, July. Reference point based multi-objective optimization using evolutionary algorithms. In *Proceedings of the 8th annual conference on Genetic and evolutionary computation* (pp. 635-642).
17. Eiben, A. E.; Smith, J. E., 2015, *Evolutionary Computing: The Origins*, Natural Computing Series, Berlin, Heidelberg: Springer Berlin Heidelberg, pp. 13–24, doi:10.1007/978-3-662-44874-8\_2, ISBN 978-3-662-44873-1.
18. Lehman, J., & Stanley, K. O., 2011. Abandoning objectives: Evolution through the search for novelty alone. *Evolutionary Computation*, 19(2), 189-223.
19. Mehrabian, A.R. and Lucas, C., 2006. A novel numerical optimization algorithm inspired from weed colonization. *Ecological informatics*, 1(4), pp.355-366.
20. Mezura-Montes, E., Velázquez-Reyes, J. and Coello Coello, C.A., 2006, July. A comparative study of differential evolution variants for global optimization. In *Proceedings of the 8th annual conference on Genetic and evolutionary computation* (pp. 485-492).
21. Stanley, K. O., & Miikkulainen, R., 2002. Evolving neural networks through augmenting topologies. *Evolutionary Computation*, 10(2), 99-127.
22. Cox et al., 1979: *The Interpretation of Igneous Rocks*, George Allen and Unwin, London.
23. Howie, R. A., Zussman, J., & Deer, W. (1992). *An introduction to the rock-forming minerals* (p. 696). Longman.
24. Le Maitre, R. W., Streckeisen, A., Zanettin, B., Le Bas, M. J., Bonin, B., Bateman, P., & Lameyre, J. (2002). *Igneous rocks. A classification and glossary of terms*, 2. Cambridge University Press.
25. Middlemost, E. A., 1986. *Magma and magmatic rocks: an introduction to igneous petrology*.
26. Shelley, D., 1993. *Igneous and metamorphic rocks under the microscope: classification, textures, microstructures and mineral preferred-orientations*.
27. Vernon, R. H. & Clarke, G. L., 2008. *Principles of Metamorphic Petrology*. Cambridge University Press.
28. Dell'Aversana, P., Empowering Deep Learning through Self-Awareness Mechanisms for Applications in Geosciences, EAGE 2024 Conference & Ex.
29. Dell'Aversana, P., Oil production optimization based on Reinforcement Learning and Self-Aware Deep Neural Networks, EAGE 2024 Conference & Ex.
30. Dell'Aversana, P., Reservoir geophysical monitoring supported by artificial general intelligence and Q-Learning for oil production optimization[J]. *AIMS Geosciences*, 2024, 10(3): 641-661.
31. Dell'Aversana P. An introduction to Self-Aware Deep Learning for medical imaging and diagnosis. *Explor. Digit. Health Technol.* 2024; 2:218–34.

**Disclaimer/Publisher's Note:** The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.