
Automated Machine Learning for Sea Ice Concentration Charting

Sven van Collenburg¹ Andreas Stokholm² Holger Hoos^{1, 3} Jan N. van Rijn¹

¹LIACS, Leiden University

²Technical University of Denmark

³RWTH Aachen

Abstract Sea Ice Concentration (SIC) charts provide important information for safe and efficient navigation in Arctic waters. Convolutional Neural Networks (CNNs) can support the automated generation of these charts from Synthetic Aperture Radar (SAR) imagery, but selecting suitable network hyperparameters remains a time-consuming manual process. In this work, we applied Bayesian optimisation to automatically configure a U-Net for SIC prediction. To support the optimisation process on a high-performance computing cluster, we implemented a parallel chief-worker architecture in which multiple candidate configurations can be trained concurrently. We additionally identified ambiguous regions in the preprocessed SAR data and introduced an evaluation procedure that excludes these regions. The configuration obtained from the final optimisation experiment achieved an R^2 score of 86.01%, an accuracy of 72.30%, and an RMSE of 1.62. A revised search setup using multiple independent optimisation runs and log-uniform learning-rate sampling identified stronger configurations than the preceding long single run, although the effects of these changes cannot be isolated individually. Overall, the results demonstrate that automated hyperparameter optimisation can effectively support the configuration of U-Net models for SIC charting while reducing the need for manual hyperparameter selection.

1 Motivation and Contributions

Sea Ice Concentration (SIC) charts describe the percentage of an area covered by sea ice. These charts need to be up to date and high resolution to guarantee safe and efficient travel in Arctic waters. Professional sea ice analysts traditionally produce SIC charts based on Synthetic Aperture Radar (SAR) images, due to their high resolution and the ability to obtain these images independently of clouds and sun illumination (Saldo et al., 2020). However, even for trained analysts, manually inspecting SAR images is a time-consuming and labour-intensive job. This conflicts with the small time window for analysing the images, due to the dynamic nature of sea ice.

Convolutional Neural Networks (CNNs) have therefore been investigated as a means of automating SIC charting. Previous work has shown that U-Net architectures can be used effectively to predict SIC values from SAR data and ice-chart labels (De Gelis et al., 2021). In particular, Stokholm et al. (2022) investigated properties such as receptive field, network depth, patch size, and SAR noise correction. Their best reported U-Net achieved an R^2 score of 86.34%, but its hyperparameters were set manually.

While CNNs can automate SIC prediction, selecting an appropriate model configuration still requires substantial manual effort. Hyperparameter optimisation can itself be time-consuming, particularly when each candidate configuration requires training a long segmentation model. The central question is therefore whether automated hyperparameter optimisation can effectively configure U-Net models for sea ice concentration charting, and how the design of the optimisation process affects the hyperparameter configurations thus discovered.

We make four main contributions. Firstly, we apply Bayesian hyperparameter optimisation to automatically configure a U-Net for SIC prediction. Secondly, we implement a chief-worker scheduler that allows for parallel training of configurations on a high-performance computing cluster. Thirdly, we investigate different setups of the optimisation process by comparing a single long Bayesian-optimisation run against several independent shorter runs. The latter setup identifies substantially stronger configurations in our experiments. Fourthly, we identify ambiguity in parts of the preprocessed SAR data and introduce an evaluation method that excludes regions where identical homogeneous inputs correspond to different SIC target classes.

2 Automated Hyperparameter Optimisation

2.1 Data and SIC Prediction

The data used for the experiments is from the European Space Agency’s AI Ready Earth Observation (AIReO) sea ice dataset, AI4Arctic/ASIP v2 (ASID-v2) Saldo et al. (2020). This dataset contains 452 scenes. Each scene is a 2D map comprised of two Sentinel-1 SAR channels, dual polarised HH (transmitted and received in horizontal polarisation) and HV (transmitted in horizontal and received in vertical polarisation) SAR images. Every scene has a corresponding SIC chart created by sea ice experts.

The original SIC classes are converted into 11 SIC classes (0-10) ranging from 0% to 100% in increments of 10%, where 0% is open-water and 100% is fully-covered sea ice. A 12th class (class 11) is created to represent pixels without SIC information, in most cases representing land. The SAR and SIC images are downsampled from 40m to 80m, and the SAR values are normalised to the range $[-1, 1]$. Finally, the preprocessed scenes are stored as NumPy arrays to reduce storage requirements and improve data-loading speed during training.

Following the experimental setup of previous work, 306 scenes are selected for training and 23 scenes for validation. The validation scenes have been chosen in collaboration with the Danish Meteorological Institute (DMI), because they contain challenging sea ice conditions. The prediction problem is formulated as a semantic segmentation task, where each pixel is classified into one of the SIC classes using a U-Net trained with sparse categorical cross-entropy.

We report three complementary performance measures: classification accuracy, Root Mean Squared Error (RMSE), and the R^2 coefficient. Accuracy measures the percentage of pixels assigned exactly to the correct SIC class. In contrast, RMSE and R^2 are sensitive to the distance between prediction and target. This distinction is important, because predicting a neighbouring SIC class is less severe than confusing open water with 100% sea ice.

2.2 Bayesian Optimisation

We use Bayesian optimisation, because of its proven effectiveness and sample efficiency (Snoek et al., 2012), requiring relatively few evaluations compared with other optimisation methods. Our implementation is based on KerasTuner (O’Malley et al., 2019), which manages the hyperparameter search and proposes candidate configurations.

The optimisation is executed on the ALICE High Performance Computing facility of Leiden University, where individual cluster jobs have a maximum running time of seven days. To make effective use of the available resources within this constraint, individual model evaluations are distributed across the cluster, while a central process controls the Bayesian optimisation procedure: A chief CPU job controls the Bayesian optimisation process and proposes configurations. GPU workers request configurations from the chief, train a single U-Net, and return the resulting validation score to the chief. After receiving a score, the chief leverages the completed evaluation for the selection of subsequent configurations.

Bayesian optimisation benefits from sequential evaluation, because each new configuration can be selected using the results of all previously completed trials. To reduce wall-clock time,

we have chosen to trade some of this information efficiency for parallel execution. A new set of hyperparameter settings is selected based only on the workers that have successfully completed their runs. Hence, when multiple workers are active simultaneously, several configurations must be proposed before the results of the other active workers are available. These proposals are therefore based on less information than they would be in a fully sequential optimisation process.

An initial accuracy-oriented experiment optimises patch and batch size. Based on this experiment, the patch size is fixed to 768 and the batch size to 16 for subsequent searches. This results in a final configuration space including U-Net depth, learning rate, dropout, maximum level size, two data-loading strategies, SAR noise correction and class weighting.

2.3 Ambiguity-Aware Evaluation

While evaluating the U-Net neural networks, we found some areas, mostly within fjords, that contained homogeneous data of all zeroes, while the corresponding SIC target contains several different classes. In other words, pixels with identical zero-valued inputs are assigned different target values. We exclude these homogeneous regions from the performance measurement calculation, both during training and during the final statistical calculation. This avoids penalising models for distinctions that cannot be inferred from the available input values. Because this changes the evaluation domain, a direct comparison with previous work is not possible, and previous work should therefore be treated only as a reference.

3 Evaluation and Results

Table 1: Selected configurations from the main optimisation experiments. Trials indicates the number of successfully completed trials. Results use the ambiguity-aware evaluation procedure.

Experiment	Trials	Accuracy	RMSE	R^2
Accuracy short run	39	79.01%	1.71	84.39%
R^2 long run	91	72.87%	1.91	80.65%
R^2 multi-run 1	53	72.30%	1.62	86.01%
R^2 multi-run 3	29	67.06%	1.62	86.03%

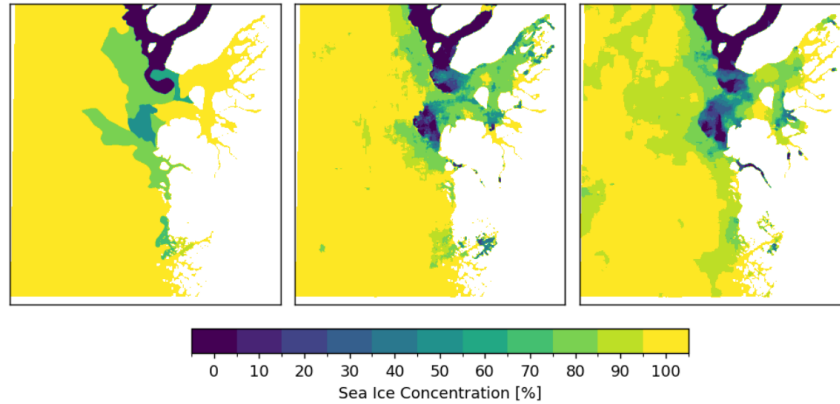


Figure 1: The predictions for the small-scale R^2 experiment. The target is on the left, the prediction of network 1.1 in the middle, and the prediction of network 3.1 on the right. For this individual scene, the R^2 scores for network 1.1 and 3.1 are 63.36% and 69.35% respectively.

We perform two accuracy-oriented experiments followed by two R^2 -oriented experiments. The first accuracy experiment explores patch size, batch size, network depth, and learning rate using a comparatively small search space. A subsequent longer accuracy search expands the space to include dropout, maximum level size, loading strategy, noise correction, and class weighting.

We then run a large R^2 -oriented optimisation using this expanded configuration space. Although 108 trials are executed and 91 complete successfully, its highest-ranked model reaches only $R^2 = 80.65\%$, less than several models found during the preceding accuracy-oriented experiments. This motivates a final experiment consisting of three independent, smaller R^2 -oriented optimisation runs.

For this final experiment, the learning-rate search is also changed from six discrete values to log-uniform sampling between 10^{-5} and 10^{-2} . The three optimisation runs execute 55, 39, and 29 trials, respectively. Selected high-performing configurations are summarised in Table 1.

Because the restart strategy, learning-rate sampling, and total number of evaluated configurations all changed, we cannot determine which factor caused the improvement. However, we can conclude that the revised search setup found better-performing configurations.

Figure 1 illustrates that models with similar aggregate R^2 values can nevertheless produce visibly different predictions. The results also show that the preferred model depends on the optimisation objective. The best model from the initial accuracy search achieves 79.01% exact classification accuracy and $R^2 = 84.39\%$, whereas the selected R^2 -optimised model obtains 72.30% accuracy and $R^2 = 86.01\%$. Optimising exact class agreement and optimising numerical closeness to the target therefore favour different configurations.

4 Discussion and Conclusion

Our experiments show that Bayesian hyperparameter optimisation can be used to automate substantial parts of the configuration process for U-Net-based sea ice concentration charting, reducing the need for manual hyperparameter tuning. To support this optimisation on a high-performance computing cluster, we developed a parallel chief-worker architecture in which multiple candidate configurations can be trained concurrently. Our best accuracy-optimised model reaches 79.01% accuracy and an R^2 score of 84.39%, while configuration 1.1 from the R^2 -optimised model reaches 72.30% accuracy and an R^2 score of 86.01%. We identified ambiguous regions in the preprocessed SAR data and introduced an evaluation method that excludes them. The resulting R^2 score of 86.01% is numerically close to the 86.34% reported in Stokholm et al. (2022), although the change in evaluation method prevents direct comparison.

Several limitations should be considered when interpreting these results. Firstly, following the experimental setup of previous work, the same scenes are used to evaluate candidate hyperparameter configurations during Bayesian optimisation and to report final model performance. This may lead to selection overfitting and means that the reported results are not based on an independent held-out test set. Secondly, excluding ambiguous regions changes the evaluation domain and prevents direct comparison with results obtained using the original evaluation procedure. Thirdly, equal class weights perform best in our experiments, even though assigning zero weight to the land class might intuitively be expected to improve training. The influence of class weighting should therefore be investigated independently. Finally, the final multi-run experiment changes the use of independent restarts and parameters. These factors should be isolated in controlled experiments before concluding that independent restarts alone improve Bayesian optimisation for SIC charting.

Overall, this work demonstrates that AutoML can provide a practical mechanism for configuring U-Net models for sea ice concentration charting. The experiments identify strong model configurations, show that the optimisation objective and search setup influence the models that are discovered, and introduce an ambiguity-aware evaluation procedure for the available SAR data. Bayesian hyperparameter optimisation reduces the need for manual configuration, while parallel execution enables more effective use of the available computational resources.

References

- De Gelis, I., Colin, A., and Longepe, N. (2021). Prediction of categorized sea ice concentration from sentinel-1 SAR images based on a fully convolutional network. *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing*, 14:5831–5841.
- O'Malley, T., Bursztein, E., Long, J., Chollet, F., Jin, H., Invernizzi, L., et al. (2019). KerasTuner. <https://github.com/keras-team/keras-tuner>.
- Saldo, R., Kreiner, M. B., Buus-Hinkler, J., Pedersen, L. T., Malmgren-Hansen, D., Nielsen, A. A., and Skriver, H. (2020). Ai4arctic/asip sea ice dataset - version 2. https://data.dtu.dk/articles/dataset/AI4Arctic_ASIP_Sea_Ice_Dataset_-_version_2/13011134.
- Snoek, J., Larochelle, H., and Adams, R. P. (2012). Practical bayesian optimization of machine learning algorithms. *CoRR*, abs/1206.2944.
- Stokholm, A., Wulf, T., Kucik, A. S., Saldo, R., Buus-Hinkler, J., and Hvidegaard, S. M. (2022). AI4SEAICE: Toward solving ambiguous SAR textures in convolutional neural networks for automatic sea Ice concentration charting. *IEEE Transactions on Geoscience and Remote Sensing*, 60:1–13.