Main Concern #1: The train/validation/test split

Lines 189-190 describe that the data was split into 80% training, 20% validation, and 10% test set. Besides the effect that this sums up to 110%, and that the use of the 3 subsets should be explained more closely, my major concern is regarding the splitting strategy, which is not explained. The data splitting strategy for timeseries data needs to account for the high temporal correlation of samples, as a random splitting strategy for timeseries data yields a test set which is not independent of the training and validation set, and can thus result in overoptimistic performance score on the test set. The authors thus need to clearly describe their train/validation/test splitting strategy, and how it sufficiently takes into account the temporal correlation of the data. Since the cross-validation uses simple k-fold cross validation, which is not appropriate for timeseries data, the associated cross-validation scores are not trustworthy. Moreover, if the authors did use a split strategy that does not take into account the temporal correlation for the following model trainings, all the scores cannot be trusted either, and the experiments would need to be repeated with a more appropriate split. Literature for timeseries forecasting discusses splitting methods and pitfalls when working with timeseries data, which need to be considered also when regressing on timeseries data (see e.g. Hyndman et al., 2018).

- We agree that the original description of the data split was incomplete and that the percentages were incorrectly stated. We have corrected the text to 70% training, 20% validation, and 10% testing, and we now state explicitly how the split was performed.
- In the experiments, the full dataset was randomly shuffled prior to splitting. To assess whether temporal correlation was inflating model skill, we performed an additional chronology-preserving experiment in which the models were trained using data up to 2014 and evaluated on data from 2015–2024. We carried out this test for MAR-IA2 and MAR-IA2-ERA. The resulting skill was unchanged: MAR-IA2 remained at $R^2 = 0.97$and MAR-IA2-ERA remained at $R^2 = 0.83$. This indicates that, for these configurations, the main conclusions are not sensitive to whether the split is random or temporally separated.
- We have added this clarification to the revised manuscript and now note explicitly that the reported performance is robust to a temporally separated train/test split.

Main Concern #2: Attribution analysis and SHAP values: The authors do not sufficiently separate the use of SHAP values to explain the model predictions on the one hand and explain the physics on the other hand. While the SHAP values can serve as a “sanity check”

for the model and to explain the reason for the model's predictions, care needs to be taken to translate that into interpretations of underlying physics.

The explanation of the SHAP implementation is not clear to me. Some more clarification on the implementation and mentioning the used libraries are needed. I wanted to find out more via the code as there is some Code/Data Availability given. However, the given Zenodo records include only data, not the code.

Lines 109-112, 219, 326-327 explain the use of SHAP values for model interpretation, by quantifying the contribution of each input feature to the model's prediction. However, lines 57 and 124 then indicate that these SHAP values are used to draw physical conclusions and find causal relations; and in line 334 it is claimed that the physical interpretability is now confirmed. However, just because the SHAP values do seem (mostly) reasonable, it does not mean that they explain causal relationships. Based on the current explanation, I am not convinced that the SHAP analysis supports conclusions exceeding the explainability of the ML model itself. For example, in line 145 the issue of using correlated inputs for attribution analysis is mentioned, and in line 296 the fluctuation of the SHAP values is explained by the remaining correlation of the features. However, the discussion beforehand used the SHAP values to draw physical conclusions, not discussing how these correlations may influence the SHAP values, and thus the trends observed. Also, direct melt drivers are identified using SHAP values, although input variables were used, which are not direct melt drivers. Furthermore, the fluctuation of the SHAP values for longitude in Figure 8 is not discussed at all.

Besides my doubts on the validity of the attribution study using SHAP values, the use of an ML emulator for such an attribution study is not properly motivated. The MAR model delivers all the radiation and turbulent heat flux values to calculate the surface energy balance and explain the melt drivers, see also Wang et al. (2021), Zhang et al. (2023), and Hofer et al. (2017).

The terminology related to SHAP is currently inconsistent (e.g., "Shapley values", "SHAP values", and "Shapley coefficients"), which is rather confusing. In the ML literature, "SHAP values" is typically used for the specific method, while "Shapley values" refers to the underlying game-theoretic concept.

- We thank the reviewer for this comment. We agree that SHAP values should be interpreted primarily as a tool for model interpretation, rather than as direct evidence of causality or as a standalone diagnosis of the underlying physics. In response, we revised the manuscript throughout to clarify this distinction.

- First, we now state explicitly in the Methods section that SHAP values are interpreted as model-based attributions of the trained emulator. They quantify how the emulator distributes predictive importance across the provided predictors, but they do not by themselves establish causal physical relationships. We have accordingly softened several statements in the Introduction, Results, Discussion, and Conclusions that previously implied stronger causal or physical claims than warranted.
- Second, we added an explicit limitation statement noting that correlations among predictors can redistribute SHAP attribution among related variables, even after removal of highly collinear inputs. We now clarify that the SHAP results are discussed as diagnostics of emulator behavior, to be used towards physical understanding, rather than as direct proof of physical melt drivers. Relatedly, we revised the discussion of geographic variables such as latitude and longitude to avoid interpreting them as direct physical drivers and instead describe them as proxies.
- Third, we clarified the SHAP implementation in the Methods section, including the library used. Specifically, SHAP values were computed using the Python shap library applied to the trained XGBoost models; in our implementation, shap.Explainer was used, which for tree-ensemble models applies the corresponding tree-specific SHAP method.
- Fourth, we standardized the terminology throughout the manuscript. We now use "SHAP values" consistently for the method outputs, while reserving "Shapley values" only for the underlying game-theoretic concept. Terms such as "Shapley coefficients" were removed to avoid confusion.
- Finally, we clarified the motivation for the emulator-based attribution analysis. Our objective is not to replace direct process-based diagnosis from MAR's native SEB output. Rather, the emulator and SHAP analysis provide a complementary and computationally efficient framework for interpreting the learned melt surrogate.

Main Concern#3: The selection of input variables

While the conclusion highlights that predictor selection was guided by physical relevance and statistical analysis, I don't see that this was done in a sufficient manner.

Line 80 mentions 2-meter air temperature to be a driver of melt energy. However, while 2-m air temp. is closely related to heat fluxes, considering the surface energy balance, it is not a driver of melt itself, as the surface energy balance is directly driven by shortwave and longwave net radiation, sensible heat flux. latent heat flux, and ground heat flux (Lenaerts

et al., 2019). The use of additional inputs, especially when trying to interpret the drivers physically, needs some more explanation.

Specifically, the motivation for using topographic variables needs more explanations. Line 83 claims that topographic variables modulate the local energy balance and atmospheric conditions. However, when using the atmospheric conditions themselves as model inputs, why use the topographic variables that contribute to those atmospheric conditions?

- We thank the reviewer for the comment. We are not sure what the reviewer means "done in a sufficient manner". We considered the drivers of melting from a Surface Energy Balance perspective and included air temperature because it is correlated with skin temperature, but, as we explain, it does not saturate at 0ºC when melting occurs. The air temperature, therefore, can provide further information to the emulator about the intensity of melting or any other information the data could provide to the model. Same for topographic variables. We wanted to explore the information content that such proxies could provide to the models' performances and also add an input that could potentially be explored in the future by others. We point out that the emulator is not aiming to create a new physical model or develop new theories about melting and SEB, but simply to replicate the model's performance while remaining consistent with energy-balance knowledge. We also decided to remove the longwave because we thought that the relatively very small improvement due to its inclusion might not be worth it in terms of potential data availability for extending the use of the emulator to colleagues. Again, the point is to have a model that compromises computational efficiency, quality of the outputs that are close to the original climate model, and applicability from others using reanalysis or other datasets.

Main Concern #4: Conflicting model scores

The abstract claims an R2 of 0.99, but none of the resulting models reach such a high score according to Table 2. Also, the formulation implies that this high score is reached for the model using ERA5 inputs, which seems misleading.

- We agree that the abstract wording was misleading. The value of $R^2 = 0.99$ referred to the optimum identified during hyperparameter tuning, not to the reported models in Table 2. We have therefore revised the abstract to report the performance of the final models consistently with Table 2, and we now state that the best-performing configuration reached $R^2 = 0.98$. We also revised the wording to avoid implying that this performance was achieved by the ERA5-based model.

The discussion, conclusion, and Table 2 do not mention if these are the scores on the test set. Also, the scores in the conclusion are different from those in the discussion and in Table 2.

- We have revised the manuscript to state explicitly that the reported quantitative performance metrics are test-set scores. In particular, we clarified this in Section 3.1 and in the caption of Table 2. We also corrected the Conclusions so that all reported R^2, MSE, and related values are fully consistent with the final test-set results shown in Table 2. Thanks for pointing this out.

Inconsistent use of metrics: in 2.5 MSE is reported, in 3.1 RMSE, and in 3.3 mean absolute difference and mean difference are used.

- To improve consistency in the presentation of model performance, we added RMSE values to the site-based evaluation in Section 3.3, while retaining the previously reported summary statistics for direct comparison with the original text. RMSE is now used alongside the existing reported quantities at Swiss Camp and K-transect S6.

Main Concern #5: Related literature

lines 41-43: The authors only mention one paper (Doury et al., 2023) for a specific ML application, which refers to a downscaling method and thus seems somewhat loosely connected to the melt emulation task, as there exists more closely related literature. Furthermore, the authors mention attribution analyses based on ML-emulation, which are not backed up by the two given references.

On the other hand, the ML approaches for downscaling climate fields (like Doury et al., 2023) should be discussed in the context of the MAR-IA2ERA models. Since these emulators rely on ERA5 data reprojected onto the MAR grid, it would be helpful to clarify that first downscaling ERA5 to the MAR configuration would ensure spatial consistency between the emulator inputs and the resulting melt predictions, and that a lot of research is being done in developing such downscaling via ML.

Lines 89-90: The given literature for the applications of XGBoost seems quite unrelated to the task in the paper too. Some additional references to work in climate/cryosphere research would be interesting, e.g. Veldhuijsen et al. (2025).

References should be double checked: Nghiem et al. (2012) was cited four times in the paper, but does not appear in the reference list. In line 291 it is used as only reference for a claim that refers to multiple studies. In contrast, Tedesco et al. 2016a and 2016b seem to be the same reference.

- We have substantially revised the Introduction to better position the manuscript within the recent literature on ML applications in cryosphere and climate science. In particular, we now discuss recent related studies by Lütjens et al. (2025), Bochow et al. (2025), and Schlager et al. (2026), and clarify how our work differs from downscaling-focused approaches and from recent neural-network-based melt emulators.
- We also clarified the role of the MAR-IA-ERA configuration. Specifically, we now note that ERA5 variables are reprojected to the MAR grid for use in the emulator, but are not dynamically downscaled to MAR-consistent fields, and we acknowledge that ongoing ML-based downscaling developments are highly relevant for improving such applications.

Further comments:

Inconsistent terminology, e.g. surface temperature and skin temperature are used interchangeably; same with inputs, variables, features, and predictands

- We have revised the manuscript to make the terminology more consistent throughout. In particular, we now use surface temperature consistently in place of mixed use of “surface temperature” and “skin temperature”; predictors for model input variables; target variable or meltwater production for the model output; and SHAP values for the attribution results. We also corrected several remaining instances where terms such as “predictands” and “features” were used inconsistently.

While the formulation “approximately normal distribution” in line 142 is odd in general, especially albedo and upward longwave radiation show distributions that are not comparable with a normal distribution at all. It would therefore be more accurate to state that the variables are not strongly concentrated within specific value ranges, rather than characterizing them as approximately normal. Furthermore, the x axis (or the data itself) seems cropped for most data, and the existence of long tails and extreme values cannot be judged.

- We agree that the phrase “approximately normal” was too strong and not accurate for several predictors. Since Figure 1 is intended primarily to compare the overall distributional shapes of the predictor variables rather than to characterize tail behavior in detail, we revised the text accordingly. The new wording avoids referring to them as approximately normal.

The hyperparameter optimization does not state the ranges that were chosen for the different parameters to be optimized, and how many different combinations were tried in total.

- We have revised Section 2.5 to state explicitly the hyperparameter ranges used in the Bayesian optimization. Specifically, the search space was defined as n_estimators = {100, 200, 500, 700, 1000, 1500, 2000, 2500}, max_depth = {2, 3, 5, 10, 15}, learning_rate = {0.05, 0.10, 0.15, 0.20}, min_child_weight = {1, 2, 3, 4}, and gamma = {0, 0.25, 0.5}. This corresponds to 1920 possible combinations in the full discrete search space. Because we used Bayesian optimization (BayesSearchCV) with n_iter = 50, 50 candidate configurations were evaluated adaptively rather than exhaustively testing all combinations.

I do not understand the claim in 365-367: “the MAR-IA emulator allows running past and future scenarios without forcing the atmospheric model in MAR with fields that are not always available for past periods and future simulations.” - Which fields are not available, which emulator product to you mean, and how do you justify the validity of extrapolating past and future data?

- We have added the following to clarify this: Moreover, the emulator offers a practical alternative when the full MAR atmospheric model cannot be run. MAR requires lateral boundary forcing over Greenland with meteorological variables at several vertical levels, and these inputs are not always available with sufficient completeness for historical reconstructions or future climate simulations. The emulator circumvents this requirement by predicting meltwater directly from a reduced set of atmospheric and surface predictors, without integrating the full atmospheric model. Its use for past or future applications is therefore promising, but should be restricted to conditions that remain within, or close to, the range represented in the training data, with further validation needed for true extrapolative cases.

Table 2 includes RMSE and bias for the 95% CI which was not explained nor mentioned or interpreted in the main text.

- We have revised the manuscript to explain that the 95% confidence intervals are reported in Table 2 and to mention them explicitly in the main text.

Figure 4 includes so many datapoints, that the distribution of errors is not visible; a density plot would be a better fit here. Furthermore, plotting test, validation and train data on top of each other is not of any use here, as the data points from train and validation set are mostly not visible anyway; and the error distribution of the subset was not discussed anyway. And again, it's unclear on which subset (train, validation, or test) the scores in the table were calculated.

- Figure 4 now only shows the test datapoints with color representing density

Figure 4 also shows the presence of negative melt predictions. In further applications, such values would likely be set to zero. It is therefore reasonable to truncate the predictions at zero and report performance metrics based on the truncated results.

- We thank the reviewer for this suggestion We have now modified the model to set to 0 any negative value. We have also created new plots and statistics that are included in the paper