This paper presents a study of the determination of surface solar irradiance from satellite using a multi-layer perceptron (MLP) compared to the state-of-the-art Copernicus Atmosphere Monitoring Service (CAMS) retrieval model. The paper is well written and well structured. The scientific question is relevant to the community and to the journal's field of application. However, the title does not accurately describe the content of the article, which deals with one particular model, namely an MLP with a hidden layer. Machine learning is a broad field that cannot be reduced to MLPs. In several places in the text, the conclusions are intended to be extended to machine learning, but they should be limited to the MLP used here.

The outcome of this work is to underline Machine Learning-specific validation issues primarily linked to generalization. These limitations are not specific to any type of ML model, we thus think that any model could be used to illustrate them.

Furthermore, more complex models are trained the same way as MLP and suffer from the same issues when it comes to generalization or out-of-domain performance. As a matter of fact, more complex models are more subject to overfitting and thus bad generalization; the conclusions obtained for MLP are therefore very likely applicable to more complex architectures (e.g. CNN, RNN, LSTM, Transformers, GAN).

We agree that this was not obvious in our introduction; we have updated the text to justify our choice better:

> “Our objective is not to introduce a new retrieval method; hence, we have deliberately opted for a simple, fully connected architecture. This choice allows our conclusions regarding generalization to extend more effectively to the realm of complex networks (convolutional, recurrent, attention-based, generative, etc.), which are generally prone to encountering greater generalization challenges (Wang et al., 2017; Ranalli and Zech, 2023)”

We also agree that investigating more complex architecture is of interest, and have thus updated the perspective section (6.3):

> “The investigation of more sophisticated neural network architectures is also of interest and would become particularly relevant when dealing with larger input datasets. Architectures such as Convolutional Neural Networks (CNNs), Recurrent Neural Networks (RNNs), or Spatio-Temoral Transformers hold promise, especially when a broader context in both time and space is required. However, it is important to recognize that such complexity may raise the risk of generalization issues, as more complex models are generally more likely to overfit.”

The sensitivity study is interesting, but raises some questions about the selection of predictors for the MLP model. For example, it is noted that not including AOD at 500nm leads to underestimation. So why not include AOD 500 as a predictor? In general, the comparison between MLP and CAMS seems biased, since CAMS seems to have access to more variables (especially thanks to the clear-sky model). For a neutral comparison, wouldn't it be better to list all the variables used by CAMS (including those hidden in the clear-sky model) and use them as predictors of the MLP model? How can MLP be expected to take into account the effect of AOD if it has no access to AOD values? The same applies to albedo.

Adding AOD 500 and albedo as predictors would indeed probably improve the performance of the model. Similarly, we could imitate CAMS's structure, and decompose the problem in the computation of a cloud index and of a clear sky (ie learn a clear sky index instead of directly the GHI). It would also be interesting to see if the normalization of MSG radiance into reflectance could help the model learn. Generally, there is a lot of feature engineering that could be done to make the model stronger. This is, however, not the purpose of this study, and because it would be a sizeable work, we believe it should be kept for future work.

That being said, we agree that a possible approach would have been to use exactly the same data as CAMS, i.e. adding AOD and albedo as input; this data, however, comes from complex numerical models (MODIS, Copernicus). In this study, we preferred to keep the ML model as simple as possible and to only use data from MSG (the other predictors are calendar-derived).

Regarding the comparison with CAMS, it is true that the two models use slightly different data. CAMS has some knowledge of the albedo and AOD while the ML model has access to more MSG pixels (even though strictly speaking, CAMS module McCloud does use neighboring pixels to determine albedo). CAMS should therefore only be seen as a reference that allows us to relativize the ML model performance in each station.

We agree that this approach may impact the results discussed in section 5.2.2 (impact of aerosols). We added a discussion to that section:

> "This result is somewhat expected, as CAMS model integrates some information about the AOD (through McClear), whereas ML model does not. Adding AOD-related predictors to the neural network may help decrease the performance gap between the two methods for clear skies."

The ML model is assigned 9 pixels, whereas for CAMS, only 1 pixel is used. Why not considering an average of the 9 pixels for CAMS? This would give an idea of the contribution of MLP compared with a simple spatial average.

We agree that some simple post-processing of CAMS – such as smoothing as you suggest – is likely to decrease its RMSE. We preferred, however, to use CAMS 'as is' because that is how it is evaluated in most studies and, arguably, used by the majority of users. In addition, using *raw* CAMS output allows us to compare to orthogonal approaches: one purely ML-based and one without any statistical processing.

Here are some detailed comments:

- l. 52: Why did the authors only use 3 bands out of the 12 available? And why not use the HRV channel which is always available for France?

  Testing the impact of other channels on a ML model performance certainly makes sense. The ability to fusion several channels is even one motivation for using machine learning. For this work, however, we only had access to a large database of past data for these three channels.

  In addition, these three channels are the ones used by heliosat2 (it uses 0.6 and 0.8) and heliosat4 (it uses all three, and, in some rare cases, 3.7um as well), so we can expect them to be the most important.

  We have added a note to section 3.2 to mention that using other bands would be of interest:

  > "These are the channels mainly used by Heliosat 2 and Heliosat 4. Other wavelengths may nonetheless be useful to a machine-learning-based model, and their impact on model performance should be explored in future work."

- l. 60: The link at the bottom of the page that describes the instrument is broken.

  Noted, thanks. I think it is fixed – at least it works for me. I will double-check once we get an online manuscript.

- l. 100: « The inverse transformation is applied to the network predictions before starting to analyze its performance ». As it is not explicit can the authors confirm that they apply the inverse transformation with the mean irradiance computed from the training set? And why do they normalize by the mean instead of removing the mean and normalizing with the standard deviation?

  We indeed use the average computed for the training. We have specified it in the text:

  > "The inverse transformation (also with the average irradiance over the training period) is applied to the network predictions."

- About the MLP structure: How the configuration (number of neurons, activations, initialisation) was chosen?

  We choose default/standard approaches for Initialization and activation. For the number of hidden layers and their size (number of hidden neurons), we tested a few configurations (64, 64x64, and 64x64x64). All had similar validation performances, so we chose the simplest one.

- l. 134: « Regularization is implemented through an early stopping procedure, which stop training if the validation error does not decrease for more than 20 epochs. ». The description of the regularization is incomplete. What is the minimum variation used to stop the training?

  It is necessary to define a minimum variation if early stopping is done with the training set. Here we use a validation set, there is, therefore, no minimum variation: the training stops if the validation error has not strictly decreased for 20 epochs.

- l. 136: « Because the last layer uses a linear activation function,... ». Why choosing a linear activation while a RELU should resolve the positiveness problem?

  It is generally recommended to use a linear activation function as the last layer for regression problems. We agree that reLu would solve the negative prediction issue, but we were worried that it may perturbate the gradient descent and cause a higher bias. Because using a linear activation function only leads to very few negative values, we chose to stick to the 'default'.

- l. 179: Are 8 years of data really needed for the training? What about the sensitivity about the size of the training set?

It is a very good point, thanks. We have added a training setup where we decrease the number of training years while keeping the number of training stations equal to 129 (the maximum). The results are discussed in section 5.3.1 "Impact of the number of training years" (see the updated manuscript).

- Table 3: The MBE values seem to be incorrect. They are not coherent with Fig. 4b

  Indeed we inverted MBE and SDE. Thank you for spotting this.

- l. 235: CAM instead of CAMS

  Indeed, thanks.

- l. 246: « Furthermore, CAMS seems to handle situations for which the clear-sky is close to one better than ML model». Is that seen from the yellow "spot" near kc=1 that is more diffuse for ML? This statement is not clear.

  That is indeed what we meant, but looking closely at the graph, it is not that clear and we may have been influenced by the results of 5.2 in this analysis. We have removed the sentence.

- Figure 3: « ML model (a and c) and for CAMS (b and d) » should be « CAMS model (a and c) and for ML (b and d) »

  Indeed, thanks.

- l. 256: Contradiction with statement line 233 while MBE was of the order of 50W.M-2

  There was a mistake in table 3 – now should be coherent.

- l. 263: « Figure 4 ». It should be Fig. 5.

  Indeed, thanks.

- l. 315: « As discussed in Section 4.2.2, we repeat the experiment 20 times for each choice of *N*, with different randomly chosen training stations at each iteration ». Is the subsampling of stations ensuring that they are well distributed in space? How do the authors achieve this?

  The subsampling is purely random and therefore there is no guarantee that the stations are well distributed in space. Training setup 3 (previously setup 2) therefore somehow overlaps with training setup 4 (previously setup 3).

  Now that we added training setup 2 (with decreasing number of training year), this further suggests that indeed the bad performance for some choice of 20 stations in training setup 3 comes from the stations' location more than the size of the training set.

  We mention this in the results section 5.3.2:

  > " Put in perspective with the results of Section 5.3.1, this suggests that the issue is not the size of the training set, but the location of the training stations."